



多媒体技术

回顾

- 声音
 - MIDI
 - 文件格式：头块、轨道块
 - 三维空间
 - 编程
- 视觉
 - 颜色
 - 视觉特性
 - 数字化

2.3.3 视觉媒体数字化

• BMP文件格式

- 图像文件头：提供文件的格式、大小等信息；
- 位图信息头：提供图像数据的尺寸、位平面数、压缩方式、颜色索引等信息；
- 彩色表：可选，24位色的没有；
- 位图数据：图像数据，定义位图的字节阵列。

数据段名称	对应的Windows结构体定义	大小(Byte)
bmp文件头	BITMAPFILEHEADER	14
位图信息头	BITMAPINFOHEADER	40
调色板		由颜色索引数决定
位图数据		由图像尺寸决定

2.3.3 视觉媒体数字化

- **BMP**文件格式

- 图像文件头

```
typedef struct tagBITMAPFILEHEADER
{
    UINT16    bfType;
    DWORD     bfSize;
    UINT16    bfReserved1;
    UINT16    bfReserved2;
    DWORD     bfOffBits;
} BITMAPFILEHEADER;
```

2.3.3 视觉媒体数字化

- **BMP文件格式**
 - 图像文件头

变量名	地址偏移	大小	作用
bfType	0000h	2 bytes	说明文件的类型，可取值为： <ul style="list-style-type: none">▪ BM – Windows 3.1x, 95, NT, ...▪ BA – OS/2 Bitmap Array▪ CI – OS/2 Color Icon▪ CP – OS/2 Color Pointer▪ IC – OS/2 Icon▪ PT – OS/2 Pointer
bfSize	0002h	4 bytes	说明该位图文件的大小，用字节为单位
bfReserved1	0006h	2 bytes	保留，必须设置为0
bfReserved2	0008h	2 bytes	保留，必须设置为0
bfOffBits	000Ah	4 bytes	说明从文件头开始到实际的图象数据之间的字节的偏移量。 这个参数是非常有用的，因为位图信息头和调色板的长度会根据不同情况而变化，所以我们可以用这个偏移值迅速的从文件中读取到位图数据。

2.3.3 视觉媒体数字化

- **BMP文件格式**

- 位图信息头

```
typedef struct tagBITMAPINFOHEADER
{
    DWORD    biSize;
    LONG     biWidth;
    LONG     biHeight;
    WORD     biPlanes;
    WORD     biBitCount;
    DWORD    biCompression;
    DWORD    biSizeImage;
    LONG     biXPelsPerMeter;
    LONG     biYPelsPerMeter;
    DWORD    biClrUsed;
    DWORD    biClrImportant;
} BITMAPINFOHEADER;
```

2.3.3 视觉媒体数字化

- **BMP文件格式**
 - 位图信息头

变量名	地址偏移	大小	作用
biSize	000Eh	4 bytes	BITMAPINFOHEADER结构所需要的字数。
biWidth	0012h	4 bytes	说明图像的宽度，用像素为单位
biHeight	0016h	4 bytes	说明图像的高度，以像素为单位。 注：这个值除了用于描述图像的高度之外，它还有另一个用处，就是指明该图像是倒向的位图，还是正向的位图。 如果该值是一个正数，说明图像是倒向的，如果该值是一个负数，则说明图像是正向的。 大多数的BMP文件都是倒向的位图，也就是高度值是一个正数。
biPlanes	001Ah	2 bytes	为目标设备说明颜色平面数，其值将总是被设为1。
biBitCount	001Ch	2 bytes	说明比特数/像素，其值为1、4、8、16、24或32。

2.3.3 视觉媒体数字化

- **BMP文件格式**
 - 位图信息头

biCompression	001Eh	4 bytes	说明图像数据压缩的类型。取值范围： 0 BI_RGB 不压缩（最常用） 1 BI_RLE8 8比特游程编码（RLE），只用于8位位图 2 BI_RLE4 4比特游程编码（RLE），只用于4位位图 3 BI_BITFIELDS 比特域，用于16/32位位图 4 BI_JPEG JPEG 位图含JPEG图像（仅用于打印机） 5 BI_PNG PNG 位图含PNG图像（仅用于打印机）
biSizeImage	0022h	4 bytes	说明图像的大小， 以字节为单位。当用BI_RGB格式时，可设置为0。
biXPelsPerMeter	0026h	4 bytes	说明水平分辨率，用像素/米表示，有符号整数
biYPelsPerMeter	002Ah	4 bytes	说明垂直分辨率，用像素/米表示，有符号整数
biClrUsed	002Eh	4 bytes	说明位图实际使用的彩色表中的颜色索引数 （设为0的话，则说明使用所有调色板项）
biClrImportant	0032h	4 bytes	说明对图像显示有重要影响的颜色索引的数目 如果是0，表示都重要。

2.3.3 视觉媒体数字化

- **BMP文件格式**

- 调色板：映射表，索引号与代表的颜色的对应关系。

```
typedef struct tagRGBQUAD
{
    BYTE rgbBlue;           -- 蓝色强度
    BYTE rgbGreen;          -- 绿色强度
    BYTE rgbRed;             -- 红色强度
    BYTE rgbReserved;       -- 保留
} RGBQUAD;
```

2.3.3 视觉媒体数字化

- **BMP文件格式**

- 调色板：映射表，索引号与代表的颜色的对应关系。

索引：(蓝，绿，红)

0号：(fe, fa, fd)

1号：(fd, f3, fc)

2号：(f4, f3, fc)

3号：(fc, f2, f4)

4号：(f6, f2, f2)

5号：(fb, f9, f6)

当biBitCount=1,4,8时，分别有2,16,256个表项

当biBitCount=24时，没有颜色表项

2.3.3 视觉媒体数字化

- **BMP文件格式**

- 位图数据

- 位图数据记录了位图的每一个像素值，记录顺序是在扫描行内是从左到右，扫描行之间是从下到上。
 - 当biBitCount=1时，8个像素占1个字节；
 - 当biBitCount=4时，2个像素占1个字节；
 - 当biBitCount=8时，1个像素占1个字节；
 - 当biBitCount=24时，1个像素占3个字节。
 - **24位RGB按照BGR的顺序来存储每个像素的各颜色通道的值**，一个像素的所有颜色分量值都存完后才存下一个像素，不进行交织存储。

2.3.3 视觉媒体数字化

- **BMP文件格式**

- 对齐

- 要求每行的数据的长度必须是4字节的倍数，如果不够需要进行比特填充（以0填充），这样可以达到按行的快速存取。
 - 填充后每行的字节数：

$$RowSize = 4 * \left\lceil \frac{BPP * Width}{32} \right\rceil$$

- BPP（Bits Per Pixel）为每像素的比特数，width是以像素为单位的行宽度

2.3.3 视觉媒体数字化



- 名称: Lena
- Size: 512×512
- 灰度图像

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0D 00 0E 0E ; .....

      ■
      ■
      ■
00000410h: F6 00 F7 F7 F7 00 F8 F8 F8 00 F9 F9 F9 00 FA FA ; ?齡? ? ?
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ?      .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋瀉hmv亞崢瀝?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+
      ■
      ■
      ■

```

```
00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; ....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 06 06 ; ....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 00 0E 0E ; .....
```

图像文件
头

位图信
息头

调色板

```
00000410h: F6 00 F7 F7 F7 00 F8 F8 F8 00 F9 F9 F9 00 FA FA
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C4 C5 C6 C7 C8 ; .....
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; .....
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; .....
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+
```

图像数
据


```
00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 00 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 0C 00 0D 0D 0D 00 0E 0E ; .....
```

bfType: 指定文件类型，是0x424D，即字符串"BM"，也就是说".bmp"文件的头两个字节都是"BM"。

```
00000410h: F6 00 F7 F7 FA ; ? 嶺? ? ?
00000420h: FA 00 FB FE FE ; ? ? ? ?
00000430h: FE 00 FF FF 36 ; ? .//969146<6
00000440h: 31 35 37 38 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+
```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 00 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 00 0D 0D 0D 00 0E 0E ; .....

```

bfSize: 指定文件大小，
包括这14个字节

```

00000410h: F6 00 F7 F7 FA ; ?齡? ? ?
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 00 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 00 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 00 00 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 00 00 00 0D 0D 0D 00 0E 0E ; .....

```

bfReserved1和bfReserved2:
 为保留字，必须为0，不予考虑

```

00000410h: F6 00 F7 00 00 00 00 00 00 00 00 00 00 00 00 00 ? ?
00000420h: FA 00 FB 00 00 00 00 00 00 00 00 00 00 00 00 00 ? ?
00000430h: FE 00 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 10 00 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 00 02 02 00 00 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 00 06 06 00 00 ; .....
00000050h: 06 00 07 07 07 00 00 00 00 00 00 00 00 00 00 00 ; ...
00000060h: 0A 00 0B 0B 0B 00 00 00 00 00 00 00 00 00 00 00 ; ...

```

bfOffBits: 为从文件头到实际的位图数据的偏移字节数，即文件前三个部分的长度和

```

00000410h: F6 00 F7 F7 F7 00 F8 F8 F8 00 F9 F9 F9 00 FA FA ; ???? ? ? ?
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 00 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 00 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 00 00 0D 0D 00 0E 0E ; .....

```

位图信息头：结构的长度是固定的，为40个字节

```

00000410h: F6 00 F7 F7 F7 00
00000420h: FA 00 FB FB FB 00
00000430h: FE 00 FF FF FF 00
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋瀉hmv亞崢瀉?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 00 03 00 04 04 04 00 05 05 00 06 06 ; .....
00000050h: 06 00 07 00 00 00 08 08 08 00 09 00 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 00 00 0C 0C 0C 00 00 00 0D 00 0E 0E ; .....

```

biSize: 指定这个结构的长度，为40字节

```

00000410h: F6 00 F7 F8 00 FA FA ; ? 嶢? ? ?
00000420h: FA 00 FB FC 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF 00 36 3C 36 ; ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋瀉hmv亞崢瀉?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 0C 00 0D 0D 0D 00 0E 0E ; .....

```

biWidth: 指定图像的宽度，单位是像素

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?  .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灑?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 00 0D 0D 0D 00 0E 0E ; .....

```

biHeight: 指定图像的高度，单位是像素

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?  .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮


```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 00 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 00 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 00 0E 0E ; .....

```

biPlanes: 必须是1，不予考虑

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?  .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 00 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0E 0E ; .....

```

biBitCount: 指定表示颜色时要用到的位数，常用的值为1（黑白二色图）、4（16色图）、8（256色）、24（真彩色图）。

```

00000410h: F6 00 F7 F7
00000420h: FA 00 FB FB
00000430h: FE 00 FF FF
00000440h: 31 35 37 38
00000450h: BF BD B5 B0
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 05 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 09 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0D 0E 0E ; .....

```

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?      .//969146<6
00000440h: 31 35 37 38 157801伶凉闷拿睦
00000450h: BF BD B5 B0 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

biCompression: 指定位图是否压缩，有效的值为 **BL_RGB**, **BI_RLE8**, **BL_RLE4**等，我们只讨论第一种即不压缩的情况

```
00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 00 00 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 08 00 00 00 00 00 00 00 00 00 00 00 00 ; .....
00000060h: 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ; .....
```

biSizeImage: 指定实际的位图数据所占用的字节数，通过公式计算：

biSizeImage = biWidth'字节 * biHeight
字节，上述公式中的**biWidth'字节**必须是4的整数倍（不是**biWidth**字节），表示大于或等于**biWidth**字节的最小的4的整倍数。

例如：如果**biWidth**字节=240，则
biWidth' 字节=240；如果**biWidth**字节=241，则**biWidth' 字节=244**。

```
00000410h: F6 00
00000420h: FA 00
00000430h: FE 00
00000440h: 31 35
00000450h: BF BD
00000460h: A6 AF
00000470h: AC A2
```

069146<6
闷拿睦
巨崢灏?
炒
3/*/*+

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 00 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 00 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 00 0C 00 0D 0D 0D 00 0E 0E ; .....

```

biXPelsPerMeter: 指定
目标设备的水平分辨率，
单位是像素/米

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?  .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 00 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 00 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 00 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 00 00 0E 0E ; .....

```

```

00000410h: F6 00 F7 F7 ?齡? ? ?
00000420h: FA 00 FB FB ? ? ? ?
00000430h: FE 00 FF FF ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

biYPelsPerMeter: 指定
目标设备的垂直分辨率，
单位是像素/米

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 00 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 05 06 ; .....
00000050h: 06 00 07 07 00 08 08 08 00 09 09 09 0A 0A 0A 0A ; .....
00000060h: 0A 00 0B 0B 00 0C 0C 0C 00 0D 0D 0D 0E 0E 0E 0E ; .....

```

```

00000410h: F6 00 F7 F7 ?  ?  ?
00000420h: FA 00 FB FB ?  ?  ?  ?
00000430h: FE 00 FF FF ?      .//969146<6
00000440h: 31 35 37 38 157801伶凉闷拿睦
00000450h: BF BD B5 B0 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B5 B1 B2 B3 B1 B3 B2 B1 B3 B1 A1 B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

biClrUsed: 指定本图像实际用到的颜色数，如果该项为零，则用到的颜色数为2的biBitCount次幂。

```

00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0D 00 0E 0E ; .....

```

biClrImportant: 指定本图
像中重要的颜色数，如果
该值为零，则认为所有的
颜色都是重要的

```

00000410h: F6 00 F7 F7 粉? ? ?
00000420h: FA 00 FB FB ? ? ?
00000430h: FE 00 FF FF .//969146<6
00000440h: 31 35 37 38 67801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 7B 00 0D 7B 01 00 0D 7B 7B A2 A0 ; 拷蛋瀉hmv亞崢瀝?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; O 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+

```

⋮


```
00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0D 00 0E 0E ; .....
      .
      .
00000410h: F6 00 F7 F7 F7 00 F8 F8 F8 00 F9 F9 F9 00 FA FA ; ?  ?  ?
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ?  ?  ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ?      .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 8A 8B 8C 8D 8E 8F 82 80 ; 拷蛋瀉hmv亞崢灑?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1      渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37      ^F:7I3/*/*+
```

调色板

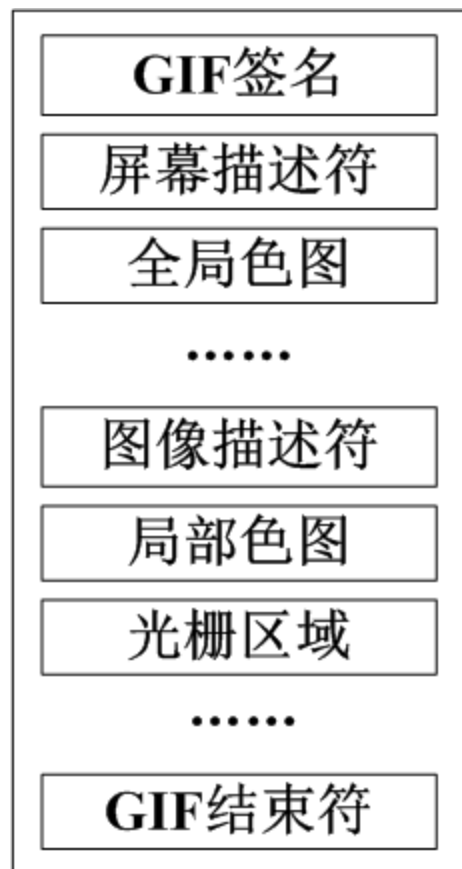
```
00000000h: 42 4D 38 04 04 00 00 00 00 00 36 04 00 00 28 00 ; BM8.....6...(.
00000010h: 00 00 00 02 00 00 00 02 00 00 01 00 08 00 00 00 ; .....
00000020h: 00 00 02 00 04 00 12 0B 00 00 12 0B 00 00 00 00 ; .....
00000030h: 00 00 00 00 00 00 00 00 00 00 01 01 01 00 02 02 ; .....
00000040h: 02 00 03 03 03 00 04 04 04 00 05 05 05 00 06 06 ; .....
00000050h: 06 00 07 07 07 00 08 08 08 00 09 09 09 00 0A 0A ; .....
00000060h: 0A 00 0B 0B 0B 00 0C 0C 0C 00 0D 0D 0D 00 0E 0E ; .....
```

图像数据

```
00000410h: F6 00 F7 F7 F7 00 F8 F8 F8 00 F9 F9 00 FA FA ; ? 嶸? ? ?
00000420h: FA 00 FB FB FB 00 FC FC FC 00 FD FD FD 00 FE FE ; ? ? ? ?
00000430h: FE 00 FF FF FF 00 2F 2F 39 36 39 31 34 36 3C 36 ; ? .//969146<6
00000440h: 31 35 37 38 4F 6C 8F B6 C1 C2 C3 C6 C4 C3 C4 C0 ; 157801伶凉闷拿睦
00000450h: BF BD B5 B0 9D 76 68 6D 76 81 86 8D 98 9E A2 A0 ; 拷蛋灣hmv亞崢灣?
00000460h: A6 AF AB B4 B3 B1 B2 B3 B1 B5 B2 B4 B3 B4 AF B1 ; 0 潮渤钹泊炒
00000470h: AC A2 7E 4E 5E 5E 46 3A 37 49 33 2F 2A 2F 2A 2B ; ~N^^F:7I3/*/*+
```

2.3.3 视觉媒体数字化

- **GIF文件格式**



重复
1
到
n
遍

6个字节: **GIF87a**

7个字节标志位

2.3.3 视觉媒体数字化

- **GIF文件格式**
 - 屏幕描述符

7 6 5 4 3 2 1 0

屏幕宽度			
屏幕高度			
m	cr	0	像素
背景			
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

1

光栅宽度，以像素为单位

2

3

光栅高度，以像素为单位

4

5

m=1 全局色图

6

cr+1 颜色分辨率的位数

像素+1 图中的位数/像素

7

通常cr = 像素

2.3.3 视觉媒体数字化

- **GIF文件格式**

- 屏幕描述符

- 一幅8位彩色**GIF**图像的文件头的16进制

4749 4638 3761 d002 bc01 f700 0006 0305

G I F 8 7 a

d002: 宽度 = $2*16^2+13*16=720$

bc01: 高度 = $1*16^2+11*16+12=444$

f7: 1,111,0,111 每个像素8位

2.3.3 视觉媒体数字化

- **GIF文件格式**
 - 全局色图

7	6	5	4	3	2	1	0
红色度	1	颜色索引0的红色值					
绿色度	2	颜色索引0的绿色值					
蓝色度	3	颜色索引0的蓝色值					
红色度	4	颜色索引1的红色值					
绿色度	5	颜色索引1的绿色值					
蓝色度	6	颜色索引1的蓝色值					
.....		剩余其他颜色					

2.3.3 视觉媒体数字化

- **GIF文件格式**
 - 图像描述符

7 6 5 4 3 2 1 0

0 0 1 0 1 1 0 0						
图像的左边						
图像的上边						
图像的宽度						
图像的高度						
m	i	0	0	0	像素	

- 1 图像分隔符字符（逗号）
- 2 图像在屏幕左边的开始处，像素计
- 3 图像在屏幕上边的开始处，像素计
- 4 图像的宽度，像素计
- 5 图像的高度，像素计
- 6
- 7
- 8
- 9
- 10

2.3.3 视觉媒体数字化

- GIF文件格式
 - 图像描述符

7 6 5 4 3 2 1 0

0 0 1 0 1 1 0 0								1
图像的左边								2
								3
								4
图像的上边								5
								6
图像的宽度								7
								8
图像的高度								9
								10
m	i	0	0	0	像素			

m=0 使用全局色图，忽略“像素”

m=1 局部色图遵循，使用“像素”

i=0 以顺序序格式化图像

i=1 以交错序格式化图像

像素+1 此图像每像素的位数

2.3.3 视觉媒体数字化

• GIF文件格式

– i=1

图像的行以4次

显示;

✓第1通道: 0、8、

16行;

✓第2通道: 4、12

、20行;

✓第3通道: 2、6、

10、14、18行;

✓第4通道: 1、3、5、

7、9、11等行。

图像行	扫描1	扫描2	扫描3	扫描4
0	1			
1				4
2			3	
3				4
4		2		
5				4
6			3	
7				4
8	1			
9				4
10			3	
11				4
12		2		
13				4
14			3	
15				4
16	1			
17				4
18			3	
19				4
20		2		

2.3.3 视觉媒体数字化

- GIF文件格式

.....	-0
.....	-1
.....	-2
.....	-3
.....	-4
.....	-5
.....	-6
.....	-7
.....	-8
.....	-9
.....	-10
.....	-11
.....	-12

2.3.3 视觉媒体数字化

- GIF文件格式



2.3.3 视觉媒体数字化

- 图像处理技术

- 图像分割算法

图像分割是将图像分成各具特性的区域，并提取出感兴趣目标的技术和过程。

灰度图像分割方法

- (1) 基于阈值的分割：通过阈值对不同物体进行分割
- (2) 基于边缘的分割：通过直接确定区域间的边界来实现分割
- (3) 基于区域的分割：把各像素划归到各个物体或区域中

2.3.3 视觉媒体数字化

- 图像处理技术

- (1) 基于阈值的分割

原理：利用图像中背景与对象之间的灰度差异

设 $f(x, y)$ 表示原图像， $g(x, y)$ 表示分割后的图像， T 为选定的灰度阈值，分割算法表示为：

$$g(x, y) = \begin{cases} 1, & f(x, y) \geq T \\ 0, & f(x, y) < T \end{cases} \quad \text{或} \quad g(x, y) = \begin{cases} 1, & f(x, y) \leq T \\ 0, & f(x, y) > T \end{cases}$$

其中：1表示物体（对象、目标）
0表示背景

2.3.3 视觉媒体数字化

- 图像处理技术
 - (1) 基于阈值的分割
- 关键： 阈值的选择



原始灰度图像



阈值=143



阈值=180

2.3.3 视觉媒体数字化

- 图像处理技术

- (1) 基于阈值的分割

关键：阈值的选择

方法1：多阈值法

设置两个灰度阈值 T_1 、 T_2 ， $T_1 > T_2$ ，这两个阈值间的灰度范围都对应于目标，即：

$$g(x, y) = \begin{cases} 1, & T_1 \leq f(x, y) \leq T_2 \\ 0, & \text{其它} \end{cases}$$

2.3.3 视觉媒体数字化

- 图像处理技术

- (1) 基于阈值的分割

关键：阈值的选择

方法2：灰度直方图阈值法

图像的灰度级范围为 $0, 1, 2, \dots, l-1$ ，设灰度级为 i 的像素个数为 n_i ，则一幅图像的总像素 N 为：

$$N = \sum_{i=0}^{l-1} n_i$$

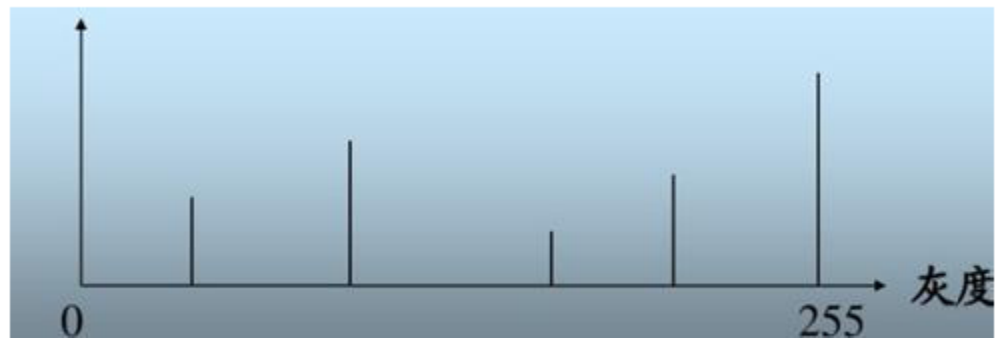
灰度级 i 出现的概率定义为： $p_i = n_i/N$

灰度直方图：灰度级的函数，它表示图像中具有某种灰度级的像素的个数，反映了图像中某种灰度出现的频率

2.3.3 视觉媒体数字化

横坐标是灰度级，纵坐标是该灰度级出现的频率或像素个数

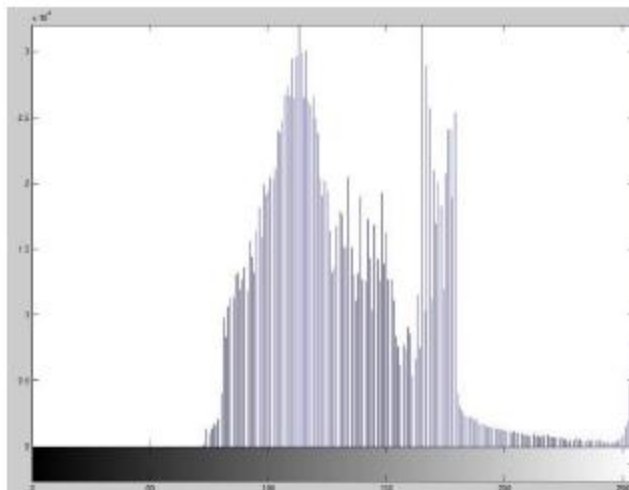
像素个数/频率



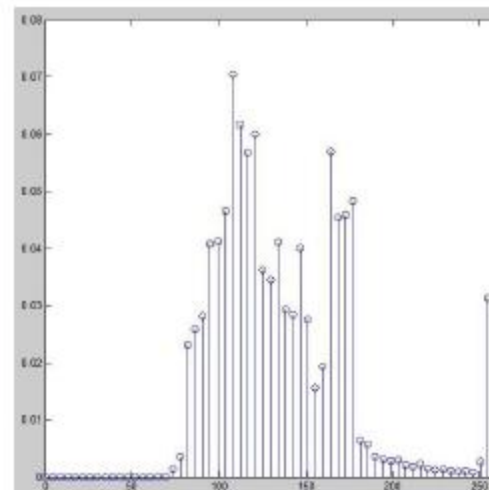
灰度直方图



原始灰度图像



灰度直方图

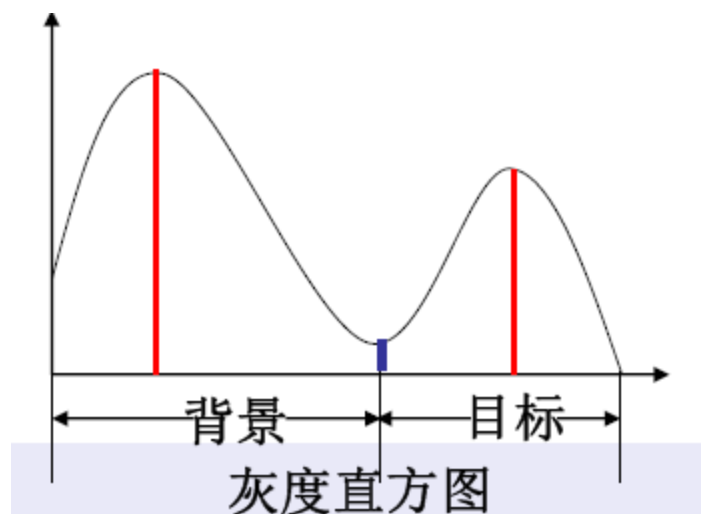


直方图归一化

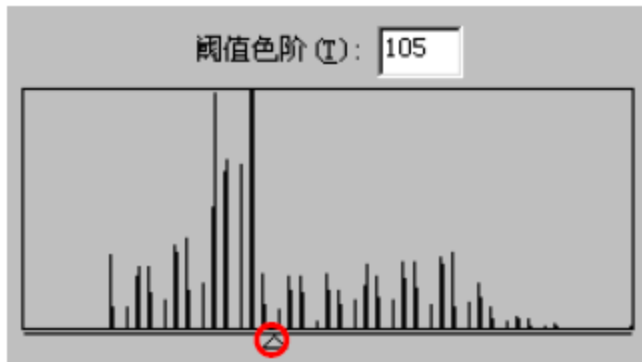
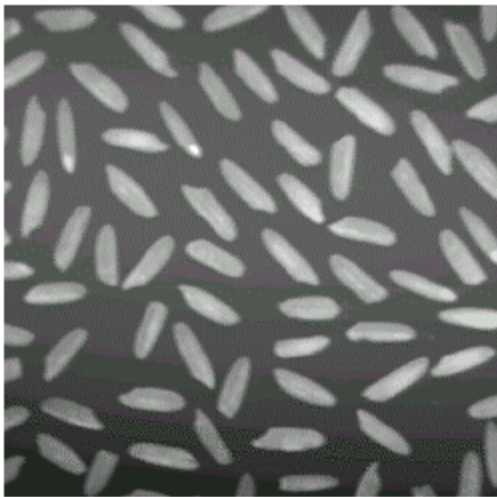
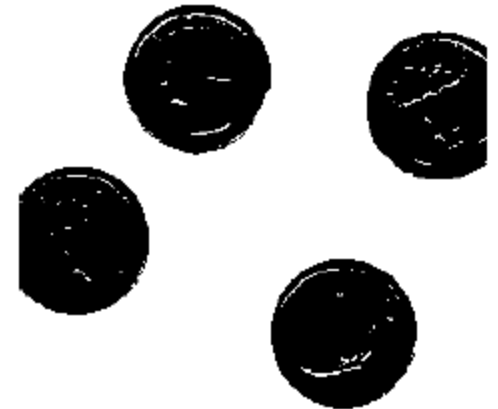
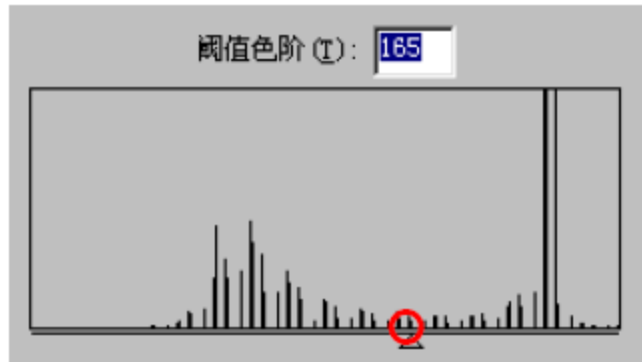
2.3.3 视觉媒体数字化

- **直方图双峰法**：如果灰度直方图呈现明显的双峰状，则选取两峰之间的**谷底**对应的灰度级作为**阈值**，谷底就是直方图的极小值。
- 为了求解极小值：将各端点相连，形成直方图的包络线 $h(z)$ ，这是一条曲线，它的极小值满足：

$$\frac{\partial h(z)}{\partial z} = 0, \quad \frac{\partial^2 h(z)}{\partial z^2} > 0$$



2.3.3 视觉媒体数字化



2.3.3 视觉媒体数字化

- 图像处理技术

- (1) 基于阈值的分割

关键：阈值的选择

方法3: OTSU

原理：类间方差最大

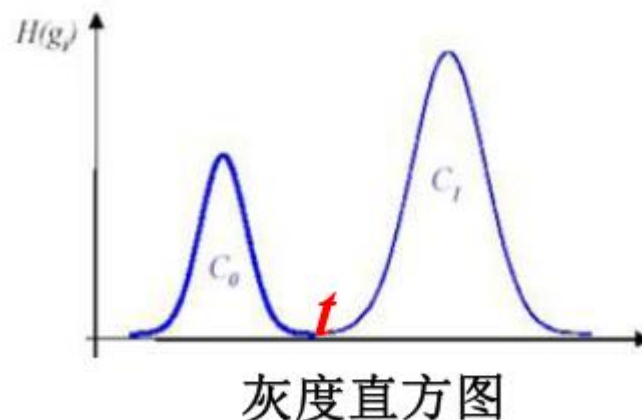
假设原始灰度图像灰度级为 $l-1$ ，灰度级为 i 的像素个数为 n_i ，则图像的全部像素数为：

$$N = \sum_{i=0}^{l-1} n_i$$

归一化直方图为：

$$p_i = n_i / N, \quad \sum_{i=0}^{l-1} p_i = 1$$

阈值 t 将图像划分为两类： $C_0=(0,1,..t)$ 和 $C_1=(t+1,t+2,...,l-1)$



2.3.3 视觉媒体数字化

C0类出现的概率:

$$\varpi_0 = P(C_0) = \sum_{i=0}^{l-1} p_i = \varpi(t)$$

C1类出现的概率:

$$\varpi_1 = P(C_1) = \sum_{i=t+1}^{l-1} p_i = 1 - \varpi(t)$$

C0类的灰度均值:

$$u_0 = \sum_{i=0}^{l-1} ip_i / \varpi_0$$

C1类的灰度均值:

$$u_1 = \sum_{i=t+1}^{l-1} ip_i / \varpi_1$$

灰度均值:

$$\mu = \mu_0 \varpi_0 + \mu_1 \varpi_1$$

类间方差:

$$\sigma^2 = \varpi_0 (\mu_0 - \mu)^2 + \varpi_1 (\mu_1 - \mu)^2$$

2.3.3 视觉媒体数字化

$$t = \operatorname{argmax} \varpi_0 \varpi_1 (\mu_1 - \mu_0)^2$$

步骤:

1. 计算灰度直方图 ;
2. 分别计算**C0**类和**C1**类的均值和概率 ϖ_0 、 ϖ_1 、 μ_0 、 μ_1 ;
3. 定义类间最大方差函数 $icv = \varpi_0 * \varpi_1 * (\mu_0 - \mu_1) * (\mu_0 - \mu_1)$;
4. 求解阈值 t ;

技巧: 优化类间最大方差函数, 计算 t

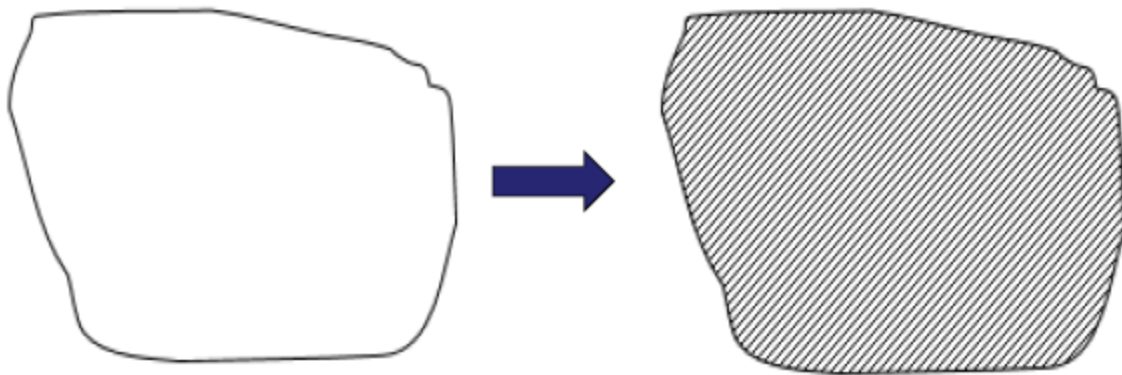
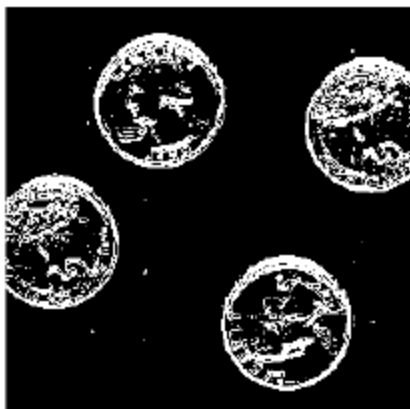
```
for (k=0;k<256;k++)  
{  
     $icv = \varpi_0 \times \varpi_1 \times (u_1 - u_0) \times (u_1 - u_0)$   
    if (icv>fmax)  
    {  
        fmax=icv;  
        thresholdValue=k;  
    }  
}
```

2.3.3 视觉媒体数字化

- 图像处理技术

- (2) 基于边缘的分割

基本原理：当物体与背景有**明显对比度**时，通过跟踪图像的**边界**，进而实现图像分割

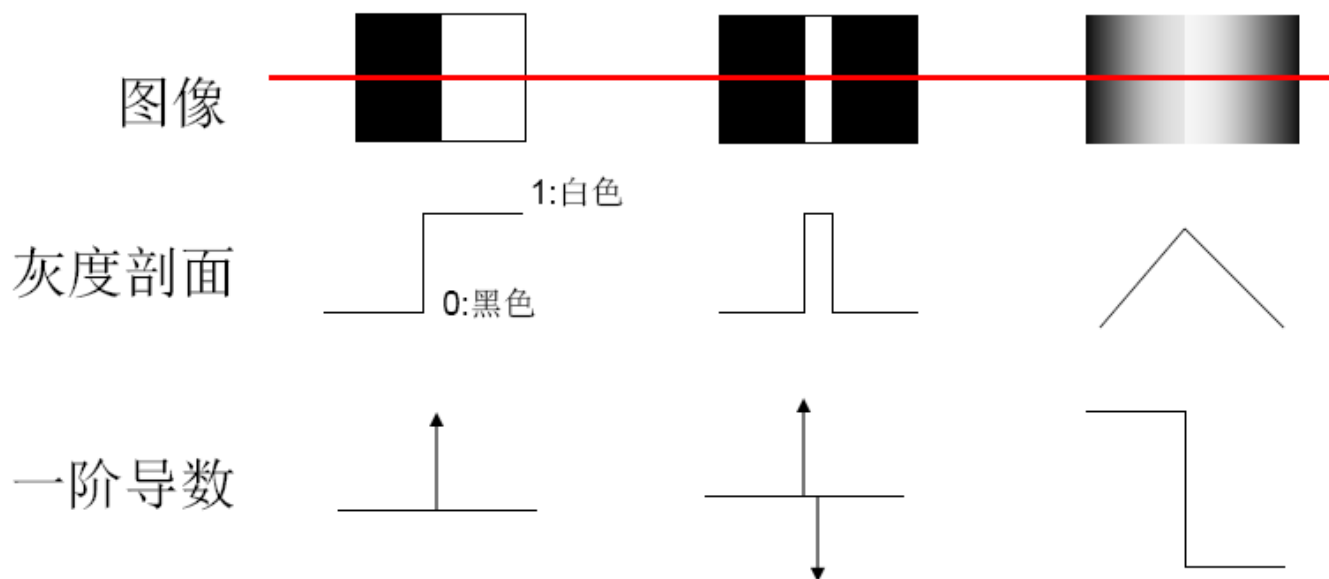


关键：如何获得图像边界？

2.3.3 视觉媒体数字化

- 图像处理技术

- (2) 基于边缘的分割



结论：一阶导数在图像由暗变明的位置处有1个向上的阶跃，而其他位置都为0，这表明可用一阶导数的幅度值来检测边缘的存在，幅度峰值一般对应边缘位置。

2.3.3 视觉媒体数字化

- 一个二元连续函数表示为 $f(x,y)$ ，它在 (x,y) 的**梯度**可表示为：

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

梯度是一个**矢量**，模和方向角为：

$$\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2}$$

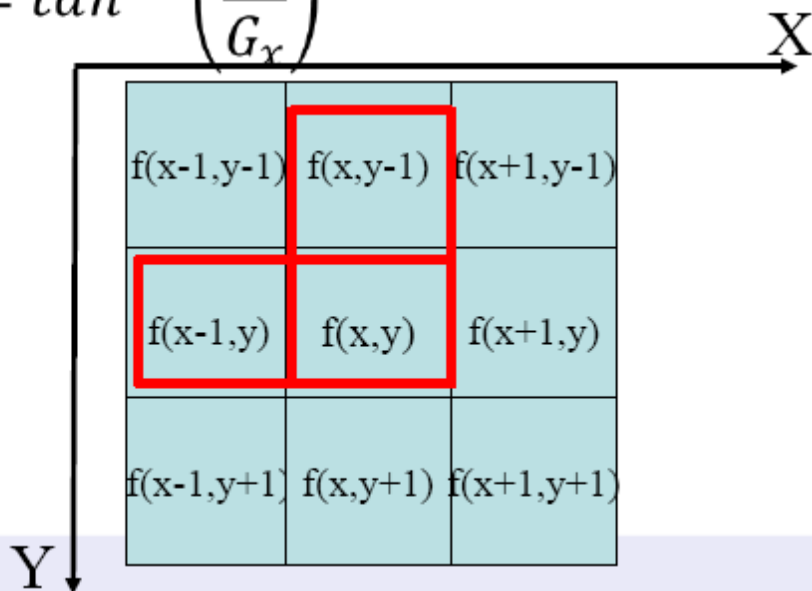
$$\alpha(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

问题：离散数字图像如何求梯度？

可用一阶差分代替求偏导：

$$\nabla_x f(x, y) = f(x, y) - f(x-1, y)$$

$$\nabla_y f(x, y) = f(x, y) - f(x, y-1)$$



2.3.3 视觉媒体数字化

一个二元连续函数表示为 $f(x,y)$ ，它在 (x,y) 的**梯度**可表示为：

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

梯度是一个**矢量**，模和方向角为：

$$\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2}$$

$$\alpha(x,y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

梯度：一阶差分求偏导；梯度的模： $\nabla f = \text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2}$



原始图像



G_x



G_y



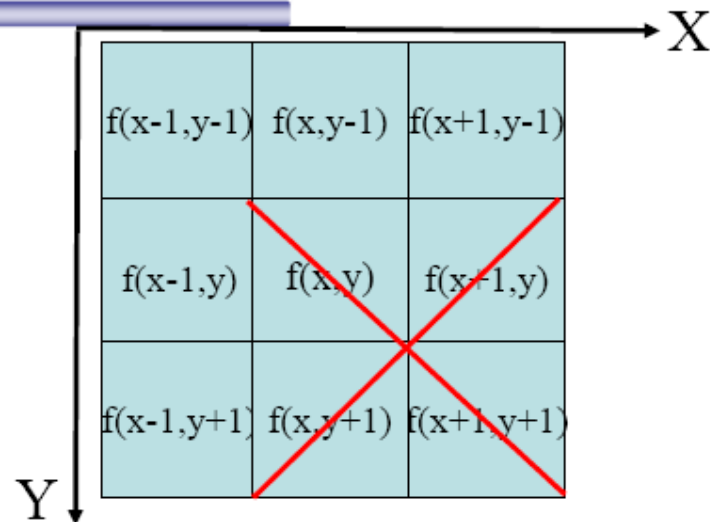
$\nabla f \approx |G_x| + |G_y|$

2.3.3 视觉媒体数字化

Roberts方法：采用对角方向相邻两像素之差表示微分，即

$$\nabla_x f(x, y) = f(x, y) - f(x + 1, y + 1)$$

$$\nabla_y f(x, y) = f(x + 1, y) - f(x, y + 1)$$



相关与卷积：信号与系统分析中基本运算**相关与卷积**，在实际图像处理中都表现为**邻域运算**。

- **邻域运算**：输出图像中每个像素是由**对应的输入像素及其一个邻域内的像素共同决定**；
- 通常邻域是远比图像尺寸小的一规则形状。如一个点的邻域定义为以该点为中心的一个圆内部或边界上点的集合。

2.3.3 视觉媒体数字化

- 相关运算

两个连续函数 $f(x)$ 和 $T(x)$ 的相关记作:

$$T(x) \circ f(x) = \int_{-\infty}^{\infty} T(x)f(x+a)da$$

- 给定图像 $f(x,y)$ 大小 $N*N$, 模板 $T(i,j)$ 大小 $m*m$ (m 为奇数)
- 相关运算: 首先使模板中心 $T((m-1)/2,(m-1)/2)$ 与 $f(x,y)$ 对应

$$g(x,y) = T \circ f(x,y) = \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} T(i,j) f\left(x+i-\frac{m-1}{2}, y+j-\frac{m-1}{2}\right)$$

2.3.3 视觉媒体数字化

• 相关运算

两个连续函数 $f(x, y)$ 和 $T(x, y)$ 的卷积运算为：

— 给定图像 $f(x, y)$

— 相关运算：首先

$$g(x, y) = T \circ f(x, y)$$

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$

$$g(x, y) = \int_{-\frac{n-1}{2}}^{\frac{n-1}{2}} \int_{-\frac{m-1}{2}}^{\frac{m-1}{2}} f(x-u, y-v) T(u, v) du dv$$

小 $m \times m$

$(m-1)/2$

$$\frac{n-1}{2}, y + \frac{m-1}{2}$$

T00	T01	T02
T10	T11	T12
T20	T21	T22

应

当 $m = 3$ 时

$$g(x, y) = T(0,0)f(x-1, y-1) + T(0,1)f(x-1, y) + T(0,2)f(x-1, y+1) + \\ T(1,0)f(x, y-1) + T(1,1)f(x, y) + T(1,2)f(x, y+1) + \\ T(2,0)f(x+1, y-1) + T(2,1)f(x+1, y) + T(2,2)f(x+1, y+1)$$

将图像中的每个像素与模板中的每个元素对应相乘，所有乘积之和作为中心像素的新值 $g(x, y)$ 。

2.3.3 视觉媒体数字化

- 卷积运算

两个连续函数 $f(x)$ 和 $T(x)$ 的卷积记作:

$$T(x) * f(x) = \int_{-\infty}^{\infty} T(x)f(x-a)da$$

- 给定图像 $f(x,y)$ 大小 $N*N$, 模板 $T(i,j)$ 大小 $m*m$ (m 为奇数)
- 卷积运算: 使模板中心 $T((m-1)/2,(m-1)/2)$ 与 $f(x,y)$ 对应

$$g(x,y) = T \circ f(x,y) = \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} T(i,j) f\left(x-i+\frac{m-1}{2}, y-j+\frac{m-1}{2}\right)$$

2.3.3 视觉媒体数字化

- 卷积运算

两个连续函数 $f(x, y)$ 和 $T(x, y)$ 的卷积运算

— 给定图像 $f(x, y)$ 的大小 $m \times n$

— 卷积运算：使 $f(x, y)$ 与 $T(x, y)$ 的大小 $m \times n$ 的卷积核 $T(x, y)$ 进行卷积运算

$$g(x, y) = T \circ f(x, y) = \sum_{i=-1}^{m-1} \sum_{j=-1}^{n-1} T(i, j) f(x-i, y-j)$$

当 $m = 3$ 时

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$

T_{00}	T_{01}	T_{02}
T_{10}	T_{11}	T_{12}
T_{20}	T_{21}	T_{22}

$$g = T(0,0)f(x+1, y+1) + T(0,1)f(x+1, y) + T(0,2)f(x+1, y-1) + \\ T(1,0)f(x, y+1) + T(1,1)f(x, y) + T(1,2)f(x, y-1) + \\ T(2,0)f(x-1, y+1) + T(2,1)f(x-1, y) + T(2,2)f(x-1, y-1)$$

模版（卷积核）旋转180度；再加权平均（类似相关运算）。

2.3.3 视觉媒体数字化

- 模板B为 3×3

$$\mathbf{B} = \begin{bmatrix} b_{00} & b_{01} & b_{02} \\ b_{10} & b_{11} & b_{12} \\ b_{20} & b_{21} & b_{22} \end{bmatrix}$$

b00	b01	b02
b10	b11	b12
b20	b21	b22

a1	a2	a3	a4	a5	a6	a7
b1	b2	b3	b4	b5	b6	b7
c1	c2	c3	c4	c5	c6	c7
d1	d2	d3	d4	d5	d6	d7
e1	e2	e3	e4	e5	e6	e7
f1	f2	f3	f4	f5	f6	f7

数字图像中c3像素的相关运算结果为：

第一步：以c3为中心选择 3×3 的邻接矩阵A；

$$\mathbf{A} = \begin{bmatrix} b_2 & b_3 & b_4 \\ c_2 & c_3 & c_4 \\ d_2 & d_3 & d_4 \end{bmatrix}$$

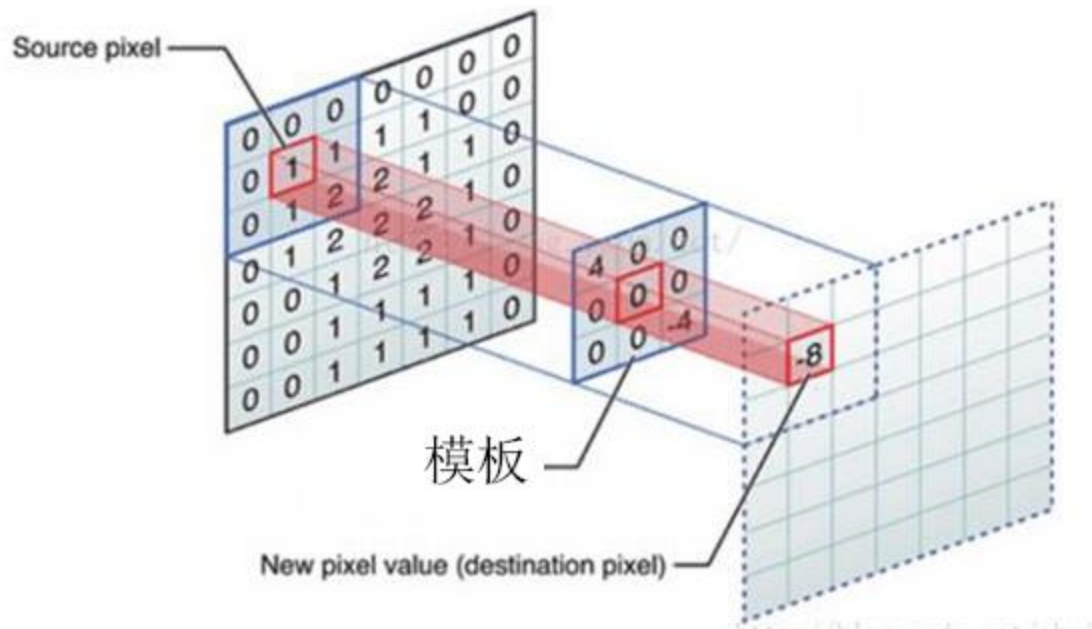
b2	b3	b4
c2	c3	c4
d2	d3	d4

注意：模板的大小
与当前像素邻域的
大小一致

第二步： $g(c3) = b_{00} * b_2 + b_{01} * b_3 + b_{02} * b_4 + b_{10} * c_2 + b_{11} * c_3$
 $+ b_{12} * c_4 + b_{20} * d_2 + b_{21} * d_3 + b_{22} * d_4$

2.3.3 视觉媒体数字化

例题：相关运算



原始图像7行*7列

问题：如何对整幅图像进行相关操作？

模板：

$$\begin{bmatrix} 4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -4 \end{bmatrix}$$

像素位置 (2, 2) 的相关运算

选择邻域3*3：

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

$$4 \times 0$$

$$0 \times 0$$

$$0 \times 0$$

$$0 \times 0$$

$$0 \times 1$$

$$0 \times 1$$

$$0 \times 0$$

$$0 \times 1$$

$$+ 2 \times (-4)$$

$$\hline -8$$

作业3

- 读入24位真彩色BMP图像filein.bmp，转换为8位（256种灰度颜色）的图像fileout.bmp（**左下角**显示学号最后一位）。

要求：

- （1）提交源代码（包括运行说明）和生成的fileout.bmp文件。
- （2）提交doc文档，回答在fileout.bmp中，bfSize、bfOffBits、biSizeImage的值（十进制）分别是多少？他们的关系是什么？

提示：

- （1）Windows下的BITMAPFILEHEADER、BITMAPINFOHEADER、RGBQUAD等类型的使用。