第17卷 第2期 华侨大学学报(自然科学版) Vol. 17 No. 2
1996年4月 Journal of Huaqiao University (Natural Science) Apr. 1996

多种格式图形接口设计与实现*

张全伙 范慧琳 吴泓昌

(华侨大学计算机科学系,泉州 362011)

摘要 介绍BMP和SPT格式图形的接口设计及程序实现.

关键词 格式图形,接口,C语言,SPT文件,BMP文件

分类号 TP 311

通用的图形软件资源很难直接在各种软件系统之间共享,特别是作为特殊图形的汉字,很难进入丰富多彩的西文图形软件,这给研制赏心悦目、图文并茂的图形软件系统带来诸多不便,通常的图形软件能够处理的图形格式又往往较少.这样,各种图形软件的通用接口设计就很有意义.正是基于这种考虑,我们在研制"图形生成与编辑软件集成化系统"中,设计了BMP和SPT格式图形的应用软件接口,并用Turbo C语言在AST 386微机上实现.格式图形的读取采用了直接写屏技术,显示速度也得到了明显提高.

1 SPT格式图形文件系统接口设计与实现

WPS桌面印刷系统中有一个SPT图文编辑器,它使用的图形文件缺省的后缀为SPT,故称SPT格式.SPT可进行中文编辑,而且有多种字体可供选择,对于企盼在图形中使用精美中文的用户,它无异是一个很好的工具.显然,要实现SPT格式图形的读取显示需对其格式和存储结构有详细的了解.SPT格式文件头有64个字节,开头的15个字节是一标志字符串"super-star file",第34个字节开始的两个字分别表示图形文件的高度和宽度.紧接在文件头之后的是位图数据.SPT是单色图形存储格式,屏幕上每一象素点只对应存储器中的一个二进制位(bit),一个二进制位可表示两种状态:1表示亮点,0表示暗点.SPT格式从图象第1扫描行左侧开始存储点阵信息,字节中高位表示行中左边象素点.例如图象的某一扫描行起始8个象素点依次是"暗暗亮亮亮暗亮亮",则其存储形式为00111011.

在SPT格式文件中,一般取行宽为8的倍数.假定一幅图象行宽w,高度h,则每行需$w/8$个字节,整幅图象需$(w/8)*h$个字节.再加上64字节的文件头,整个文件长度为$(w/8)\times h+64$个字节.节省存储空间是SPT格式的优点,而只能存储单色图象是它的不足之处.

在系统中,我们的用户界面采用图符ICON界面,菜单形式是文本菜单和图符混合使用,对系统的每一个功能用一个形象化的小图符以代替呆板的文字叙述,并借助鼠标进行各种功

* 本文1995-11-07收到;福建省自然科学基金资助项目

能的切换及绘制操作.SPT 格式图象读取显示功能是通过屏幕上弹出一个窗口,提示图形格式的选取,点中 SPT-ICON,则提示输入文件名.用户输入文件名后,屏幕上显示出 SPT 图象,此后可对屏幕上的图形进行再加工,使之更加完美.

SPT 格式图象读取可用 Turbo C 描述如下

```
void spt _ load(void)
{
  unsigned int c,m,n;
  FILE * fpl;
  void get _ name(char * );
   outtextxy (56-SUB _ X, 85-SUB _ Y,
"SPTNAME>>");
  get _ name(sptfname);
  mouse _ off ( );
  restore _ image (54-SUB)_ X,25-SUB
    _Y/ * 白色为前景色 * /
    menu _ buffer);
    mouse _ on ( );
  if (fpl = fopen (sptfname, "rb")) = =
NULL)
{
    restorecrtmode( );
    printf("%S:open error!",sptfname);
    return;/ * exit(1); * /
  }
  fseek(fpl,34,SEEK _ SET);
  fread(&spt _ width,2,1,fpl);
        / * 读入 SPT 图形的宽度 * /
  fseek(fpl,36,SEEK _ SET);
        / * 读入 SPT 图形的高度 * /
    fread(&hight,2,1,fpl);
if((spt _width * 8)! =0
        spt _ width=spt _ width/8+1;
  else spt _ width=spt _ width/8;
  if (hight > PORT _ BOTTOM-PORT _
TOP)
hight=PORT _ BOTTOM _ PORT _ TOP;
  fseek(fpl,64,SEEK _ SET);
        / * 定位到图形数据区 * /
  for(i=0;i<hight,i++)
  {
    for(j=0;j<spt _ width;j++)
    {
        c=fgetc(fpl);
        / * 蓝色为背景色 * /
        for(m=0;m<8;m++)
        {
          n=c>>(7-m)&1;
          if((j * 8+m+PORT _ LEFT<
PORT _ RLGHT) (i<PORT
    _ BOTTOM))
            {
              if(n)mem _ point(j * 8+m+
PORT _ LEFT,i+PORT _ TOP,BLUE);
    else mem _ point(j * 8+m+PORT _
LEFT,i+PORT _ TOP,WHITE);
            }
          }
    / * 改变关键字 BLUE WHITE 可以 * /
      }
    / * 选择任意两种颜色 * /
    }
    fclose(fpl);
  }
```

2 BMP 格式图形软件系统接口设计与实现

MS-WINDOWS 中有一个小型绘图程序 Paintbrush,它提供了很强的图形绘制和编辑功

能，诸如图形的放大、缩小、拼接、旋转、倾斜等一应俱全．Paintbrush 使用的主要存储格式为 BMP 格式．BMP 格式文件是属于 MS-WINDOWS 的资源文件，在 MS-WINDOWS 应用程序中可以直接使用，而且位图文件的格式与 OS/2 中 Presentation Manager 的位图格式相同，因此在实际中得到广泛的应用．增加 BMP 文件的读取显示对于增加图形软件系统的通用性及功能互补有很大帮助．

BMP 格式文件由以下四部分组成：位图文件头、位图信息头、颜色表和位图数据．它们的结构定义如下：

```
typedef struct tag Bitmap fileheader
{UNIT bfTYPE;/ * 位图文件类型必须为 BM * /
DWORD bfsize;/ * 位图文件的大小，以字节为单位 * /
UNIT bfReserved1;/ * 位图文件保留字，必须为 O * /
UNIT bfReserved2;/ * 位图文件保留字，必须为 O * /
DWORD bfoffBits;/ * 位图阵列起始位置，相对于位图文件头的偏移量 * /
} BITMAPFILEHEADER;
typedef struct tag BITMAPFILEHEADER
{ DWORD bisize;/ * bmiHeader 的长度，以字节为单位 * /
LONG biwidth;/ * 位图的宽度，以象素为单位 * /
LONG biHight;/ * 位图的高度，以象素为单位 * /
WORD biBitCount;/ * 每个象素所需位数，必须是 1(单色)，4(16 色)，8(256 色或 24(2^24
                    色点之一 * /
DWORD biCompression;/ * 位图的压缩类型，必须是 O(不压缩)，1(R1 的压缩类型)，2
                        (B1-R 所压缩类型)之一 * /
DWORD bisizeImage;/ * 位图的大小，以字节为单位 * /
LONG bixpelespermeter;/ * 位图目标设备水平分辨率，以每米象素数为单位 * /
LONG biypelespermeter;/ * 位图目标设备垂直分辨率，以每米象素数为单位 * /
DWORD biclruserd;/ * 位图实际使用的颜色表中的颜色变址数 * /
DWORD biclrImportant;/ * 位图显示过程中被认为重要的颜色变址数 * /
} BITMAPINFOHEADER;
typedef struct tagRGBQUAD
{ BYTE rgbBlue/ * 蓝色亮度值 * /
  BYTE rgbGreen / * 绿色亮度值 * /
  BYTE rgbRed / * 红色亮度值 * /
  BYTE rgbReserved;
} BTMAPRGBQUAD;
```

BMP 文件按行存储图象．存储时从左到右，从下向上扫描图象，依次记录每一象素的颜色值．假设一幅图象有 m 行 n 列，则 BMP 文件最先存放第 $m-1$ 行，然后是第 $m-2$ 行，…直到第 0 行．根据 BiBitcount 的取值只能是 1 或 4 或 8，则每个字节分别存放 8 或 4 或 1 个象素的彩色值．因此，每行象素需存储量是 $(n\times \text{BiBitcount})/8$，若 $n\times \text{BiBitcount}$ 不刚好是 8 的整

数倍,则不是的位用填充.BMP格式还规定,每行象素所占字节数必需是4的倍数,所以每行象素实际所占的字节数为[(n * BiBitcount+31)/32] * 4.

BMP格式图象读取可用Turbo C描述如下:

```
/* 设置调色板 */
void set_palette(FILE * fpl)
{
  int i;
  fseek(fpl,0x36,SEEK_SET);
  outportb(0x3c8,0);
  outportb(0x3c9,rgbquad.red>>2);
  outportb(0x3c0,rgbquad.green>>2);
  outportb(0x3c9,rgbquad.blue>>2);
}
/* 读取所存储的图形 */
void bmp_load( )
{
  char fname[80];
  int i,j,temp_x,temp_y;
  int col=SUB_X,row=SUB_Y;
/* 从点(col,row)开始显示图形 */
  FILE *fpl;
  unsigned char color,coll;
  void get_name(char *);
   outtextxy (56-SUB_X, 85-SUB_Y,
"BMPNAME>>");
  get_name(fname);
  mouse_off ( );
  restore_image (54-SUB_X,25-SUB
_Y,menu_buffer()
                mouse_on ( );
  if((fpl=fopen(fname,"rb"))==NULL)
  {
    restorecrtmode();
    prinft("%s,isn't exist! /n",fname);
    return;/*exit(1);*/
}
fseek(fpl,0,SEEK_SET);
fread(&ftype,sizeof(ftype),fpl);
if(ftype! =0x4d42)
      /* 判断文件类型 */
{
  printf("Not's BMP file! /n");
  exit(1);
}
fseek(fpl,0x0a,SEEK_SET);
fread(&offset,sizeof(offfset),1,fpl);
fseek(fpl,0x12,SEEK_SET);
fread(&width,sizeof(width),1,fpl);
          /* 读图形的宽与高 */
fread(&height,sizeof(height),1,fpl);
fseek(fpl,offset,SEEK_SET);
    /* 定位到图形数据的起始位置*/
for(i=heigh;i>=0;i-)
{
        /*从下向上显示*/
  for(j=0;j<width;j++)
  {
        /*是从下往上存放的*/
    color=fgetc(fpl);
    coll=(color&0xf0)>>4;
    /*取字节头四位*/
    switch(coll){
    /* 两个象素点用一个字结 */
        case 0xl:coll=0x4;
                 break;
        case 0x4:oll=0x1;
                  break;
        case 0x6:coll=0x3;
break;  /* 把象素码值还原*/
        case 0x7:coll=0x8;
break;  /* 因BMP图形存放*/
        case 0x8:coll=0x7;
break;  /* 的是经过转换后*/
```

```
        case 0x9:coll=0xc;
break;  /* 的值,必须经过 */
        case 0xb:coll=0xe;
break;  /* 还原处理 */
        case 0xc:coll=0x9;
                 break;
        case 0xe:coll=0xb;
                 break;

temp_x=col+j;
temp_y=row+i;
if((temp_x<=PORT_BOTTOM)
(temp_y<=PORT_RIGHT))
                 mem_point(col+j++,
row+i,coll);
    /* 显示象素点 */
coll=(color&0x0f);  /*取字节后四位 */
switch(coll)
{
      case 0xl:coll=0x4;
               break;
      case 0x4:coll=0x1;
               break;
      case 0x6:coll=0x3;
               break;
      case 0x7:coll=0x8;
               break;
      case 0x8:coll=0x7;
               break;     /* 还原 */
      case 0x9:coll=0xc;
               break;
      case 0xb:coll=0xe;
               break;
      case 0xc:coll=0x9;
               break;
      case 0xe:coll=0xb;
               break;
  temp_x=col+j;
  temp_y=col+i;
  if((temp_x<=PORT_BOTTOM)||
(temp_y<=PORT_RIGHT))
  mem_point(col+j,row+i,coll);
  }
}
fclose(fpl);
}
void save_pic(char *fname)
{
FLLE *fpl;int i;
register long j;char far *ptr;
fpl=fopen(fname,"wb");
outportb(0x3ce,4);
for(i=0;i<4;i++)
{
  outportb(0x3cecf,i);
  ptr=(char far *)0xa0000000;
  for(j=0;j<384001;j++)
  {
  putc(*ptr,fpl);
  ptr++;
  }
}
fclose(fpl);
outportb(0x3cf,0);
}
void load_pic(char *fname)
  {
    FILE *fpl;register int i;
    register long j;char far *ptr;
    fpl=fopen(fname,"rb");
    outportb(0x3c4,2);
    for(i=1,i<9;i*=2)
    {
      outportb(0x3c5,i);
      ptr=(char far *)0xa0000000;
```

```
for(j=0,j<384001;j++)
{
  *ptr=getc(fpl);
  ptr++;
}
}
fciose(fpl);
outportb(0x3c5,0xf);
}
```

参 考 文 献

1 章 伟,刘春敏. 图符用户界面的程序设计. 中国计算机用户,1994,(3):63～66
2 张晓东. TVGA 1024*768 模式下的两个基本图形功能函数. 中国计算机用户,1994,(5):40～46
3 曲 勇. 软件封面的制作. 电脑编程技巧与维护,1995,(2):54～59

Design and Implementation of Diversified Format-Graphic Interfaces

Zhang Quanhuo Fan Huilin Wu Hongchang

(Dept. of Computer Science, Huaqiao Univ., 362011, Quanzhou)

Abstract The introduction of formatted file of graphic into various application software systems of graphic is of significance for improving the generality of these system and the complementarity of their functions. The authors present here the design and the implementation of BMP and SPT format-graphic interfaces.

Keywords format-graphic, interface. C language, SPT file, BMP file