

实现若干图文特效的 VB 编程方法和示例

朱从旭 (长沙中南工业大学信息工程学院计算中心 410083)

摘要:本文探讨在 VB 中用定时器和 API 编程实现文字和图形若干动态特效的方法,并提供了程序示例。

关键词:VB(Visual Basic) 图文特效 API(Application Programming Interface)函数

在电影或电视中经常可见到一些字幕或图片的动态变化特技,在制作多媒体教学软件或广告宣传媒体等应用中,也常常需要用到一些图文的动态变化特效来增强软件的效果。VB 作为 Windows 软件的通用开发工具,具有强大的多媒体应用开发功能,尤其是利用 Windows API(应用程序编程接口)可大大扩展其功能,使 VB 在多媒体软件开发方面具有更广阔的应用前景。本文讨论运用 VB 的定时器和 API 编程实现图片和文字多种动态变化特效的方法。

一、实现的编程方法和原理

1. 程序界面设计

在 Windows 程序设计中,一个重要而基本的工作是界面设计。为便于对编程方法的理解,我们首先简介本文述及的程序界面。在 VB 工程窗体 Form1 中添加如下控件并预设有关属性:一个标签控件 Label1,其 backcolor 等于 form1 的 backcolor;bordstyle 为 0;Autosize 为 True。一个图象控件 Image1,其 Stretch 和 Visible 均为 True;Bordstyle 为 0。两个图片框控件:Picture1(AutoRedraw = True)和 Picture2(AutoRedraw = False),Visible 属性均为 False,ScaleMode 属性均为 Pixels;Bordstyle 属性均为 0。三个定时器控件 Timer1、Timer2 和 Timer3 的 Enabled 属性均为 false;给 interval 设定适当非零值。8 个命令按钮,采用控件数组从左到右依次为 Cmd(0)到 Cmd(7),界面如文后程序运行图所示。

2. 文字动态特效的实现方法

采用下列方法可实现文字动态放大或缩小的效果。基本原理是基于文字的 font 对象的 size 属性,该属性决定了文字的大小;通过 VB 的定时器控件定时改变文字 font 对象的 Size 属性,就达到动态改变文字大小的目的。本文示例程序中通过 timer1 的 timer 过程来控制标签 Label1 的标题文字大小;单击 cmd(0)可触发 timer1-timer

过程。

而通过连续改变标签中的文字字数,可产生文字的渐现渐隐效果。即从一字符串中取若干个文字作为标签 Label1 的标题,如果所截取的字数递增(直到等于字符串的全部字数),则实现文字渐现效果;而若截取的字数递减(直到字数为 0),则实现文字渐隐效果。本文示例程序通过定时器 timer2 的 timer 过程定时改变截取的字数,来达到此效果;单击 cmd(1)可触发 timer2-timer 过程。此外,定时改变标签的 forecolor 属性可产生文字的闪烁效果;定时改变标签的位置(Top 和 left 属性)可产生文字的运动效果,这些效果本文不打算详论。

3. 若干图形动态特效的实现方法

将图象控件的 stretch 属性设为 true,可实现控件中图片大小随图象控件大小的变化而变化。通过定时器定时改变 image1 控件的大小,便产生图片动态缩放的效果。产生此效果的示例程序可参见 timer3-timer 过程,单击 cmd(2)按钮可触发 timer3-timer 过程。

用 API 函数 BitBlt 复制图片的方法可实现多种图形视觉过渡效果。该函数的原型为:

```
BitBlt(hDestDC, x, y, nWidth, nHeight, hSrcDC, xSrc, ySrc, dwRop)
```

其中各参数的含义为: hDestDC 和 hSrcDC 分别表示目标设备描述表句柄和源设备描述表句柄。即要把图片从 hSrcDC 复制到 hDestDC。PictureBox 控件的 hDC 属性就是我们要用到的设备描述表句柄。x, y 表示目标中左上角的位置, nWidth, nHeight 表示目标中的宽和高;即要把图片复制到目标设备描述表中的什么位置,以及复制多宽和多高。xSrc 和 ySrc 表示源图片中的左上角位置;即表示从什么位置开始复制。DwRop 参数表示一种绘制方式,常用直接复制方式,这时就取 SRCCOPY。创建一个通用过程 PicCpy 完成多种复制方式,将图片框 Picture1 中的图片作为源图片(源图片应不可见),将其复制到目标图片框 Picture2 中。值得注意的是:如果源图

片框中未装图片,复制过程实际上就是用窗体背景色去覆盖目标图片框,这时便产生目标图片渐隐效果。每隔一时间隔复制一个图片条纹(条纹宽用 strip 表示),且复制的位置(p2)每次变化 strip。用 API 函数 Sleep 实现延时,即每复制一个条纹后应延迟一个时间 delay。参数 strip 和 delay 指定不同的值则得到不同的动态效果。此外, PicCpy 过程还可通过参数 flags 来表示多种不同的变化方式(如从左到右渐变,从上到下渐变,从中间往两边渐变等等),本文程序提供了四种变化方式作为示例,适当修改程序还可实现许多其他变化方式。对两个 API 函数 BitBlt 和 Sleep 及常量参数 SRCCOPY 要在窗体的通用声明中声明,同时声明所需的窗体级全局变量。

二、实现若干图文特效的示例程序

由于采用控件数组技术,使得所有的单击按钮事件共由一个 cmd-click(Index)过程来处理,通过参数 Index 的值可标识究竟哪一个按钮被单击。本文所有的文字和图形都有两种相反的变化方式(如放大与缩小,渐现与渐隐),由一二值变量的值决定何种效果。连续单击同一按钮产生的变化效果相反,均通过转换相应的二值变量 k(i)来实现(数组元素 k(i)都只能取 -1 或 1)。转换 k(i)采用如下程序结构:if k(i)=1 then k(i)=-1,else k(i)=1。对全局变量和 API 函数的通用声明(API 函数 Sleep、BitBlt 及其常参数 SRCCOPY 的声明从略,可从 API 浏览器中复制、粘贴):

```
Dim k(7) As Integer, num As Integer, zt As Integer, s As String
```

```
Private Sub Form-Load()
```

```
For i = 0 To 7
```

```
    k(i) = -1 '初始化所有二值变量为 -1
```

```
Next i
```

```
s = "产生文字动态特效" '字符串 s 中的文字可为其他文字
```

```
zt = 5 '初始化 zt,zt 为字体大小控制变量
```

```
num = 0 '初始化 num,num 存放被显示文字的字数
```

```
End Sub
```

```
Private Sub Cmd-Click(Index As Integer)
```

```
    If k(Index) = 1 Then
```

```
        k(Index) = -1 '转换二值变量 k(Index)
```

```
    Else
```

```
        k(Index) = 1
```

```
    End If
```

```
    Select Case Index '根据 Index 值确定用户单击了哪一个命令按钮 Cmd(Index)
```

```
        Case 0 ' Index=0,则激活 timer1 的 timer 过程
```

```
            Timer1.Enabled = True
```

```
        ' Case 1,2 与上述类似,分别激活 timer2、timer3 的 timer 过程
```

```
        Case 7 '退出应用程序
```

```
            Unload Me:End
```

```
        Case Else '调用图片复制通用过程 PicCpy
```

```
            PicCpy 20, 40, Index '前两个参数决定图象变化的快慢和步幅效果
```

```
        End Select
```

```
End Sub
```

```
Private Sub Timer1-Timer() '实现文字缩放的过程
```

```
    zt = zt + 5 * I(0) '若 k(0)=1 则使字体放大;k(0)=-1 使字体缩小
```

```
    If zt > 40 Or zt <= 5 Then '控制字体放大或缩小的范围
```

```
        Timer1.Enabled = False
```

```
    End If
```

```
    Label1.Font.Size = zt
```

```
    Label1.Left = (Form1.Width - Label1.Width) / 2 '使缩放显示的文字在窗体中居中
```

```
    Label1.Top = (Form1.Height - Label1.Height) / 2
```

```
    Label1.Caption = s
```

```
End Sub
```

```
Private Sub Timer2-Timer() '实现文字渐隐渐现的过程
```

```
    Label1.Font.Size = 40
```

```
    num = num + k(1) '若 k(1)=1 则截取字符串中字数 num 增加,l(1)=-1 时截取字数减少
```

```
    If num >= 0 And num <= Len(s) Then
```

```
        Label1.Caption = Left$(s, num)
```

```
    Else
```

```
        Timer2.Enabled = False
```

```
    End If
```

```
End Sub
```

```
Private Sub Timer3-Timer() '实现图片缩放的过程,k(2)决定图片是放大还是缩小
```

```
    Image1.Width = Image1.Width + (Form1.Width / 10) * k(2)
```

```
    Image1.Height = Image1.Height + (Form1.Height / 10) * k(2)
```

```
    If (Image1.Width <= Form1.Width / 10) Or (Image1.Width >= Form1.Width / 2) Then
```

Timer3.Enabled = False '确定图片放大或缩小到何种程度为止

End If

Image1 = LoadPicture("e:\multi\cat.bmp") '图片文件可用别的图片

End Sub

'实现图片渐现渐隐多种视觉过渡的通用过程:

Private Sub PicCpy(strip As Integer, delay As Integer, flags As Integer)

Dim p1w As Long, p1h As Long, px As Long, py As Long, px1 As Long, py1 As Long

Picture2.Visible = True '复制时,源图片可以隐藏,但目标图片应可见

If k(flags) = 1 Then '实现目标图片的渐现时,源图片框有具体图片

Picture1 = LoadPicture("e:\multi\cat.bmp") '图片文件可用别的图片

Else '实现目标图片的渐隐时,源图片框无图片

Picture1 = LoadPicture("")

End If

p1w = Picture1.ScaleWidth: p1h = Picture1.ScaleHeight

px = 0: py = 0: px1 = p1w / 2: py1 = p1h / 2

Select Case flags

Case 3 '图片从左至右变化

Do While px <= p1w

BitBlt Picture2.hDC, px, 0, strip, p1h, Picture1.

hDC, px, 0, SRCCOPY

px = px + strip

Sleep delay :DoEvents

Loop

Case 4 '图片从中间往两边变化

Do While px1 <= p1w

BitBlt Picture2.hDC, px1, 0, strip, p1h, Picture1.

hDC, px1, 0, SRCCOPY

BitBlt Picture2.hDC, p1w - px1, 0, -strip, p1h, Picture1.hDC, p1w - px1, 0, SRCCOPY

px1 = px1 + strip

Sleep delay :DoEvents

Loop

Case 5 '图片从上至下变化

Do While py <= p1h

BitBlt Picture2.hDC, 0, py, p1w, strip, Picture1.

hDC, 0, py, SRCCOPY

py = py + strip

Sleep delay :DoEvents

Loop

Case 6 '图片从中间往上下变化

Do While py1 > 0

BitBlt Picture2.hDC, 0, py1, p1w, strip, Picture1.hDC, 0, py1, SRCCOPY

BitBlt Picture2.hDC, 0, p1h - py1, p1w, -strip, Picture1.hDC, 0, p1h - py1, SRCCOPY

py1 = py1 + strip

Sleep delay :DoEvents

Loop

End Select

End Sub

以上程序均在 VB5.0 和 Windows98 环境下调试通过。将 PicCpy 过程的有关代码稍加修改或增加一些条件选择,可实现更多效果,程序中几种典型图片视觉过渡效果如图 1-4 所示(为便于比较,在制作下列界面图时,有意将程序左边源图片的 Visible 属性改为了 True)。

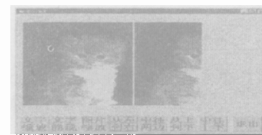


图1 图片从左至右渐现



图2 图片从上至下渐隐



图3 图片从中间往两边渐现



图4 图片从中间往上下渐隐

参考文献

- [1] 赵振修,鲁守智 编著,(1998.5), Visual Basic 5.0 视窗编程技术,北京大学出版社。
- [2] 王克己 主编,(1998.10), Visual Basic 5.0 中文版用户手册,人民邮电出版社。
- [3] 李楠,祁梅等编,(1998.10), Visual Basic 5.0 编程实例和技巧,人民邮电出版社。

(来稿时间:1999年4月)