

胎侧字体排列图软件 LTSL 数据结构

陈振艺

[上海轮胎橡胶(集团)股份有限公司 轮胎研究所,上海 200245]

摘要:介绍轮胎胎侧字体排列图辅助设计软件 LTSL 的主要数据结构,其字符的数据结构声明在排列项目数据结构中,排列项目数据结构声明在字体排列图数据结构中,使得代码层次分明,且不易出错。编程时采用预先计算部分变量的方法,避免了软件操作时的临时计算,使软件使用时不会有反应迟钝的感觉。

关键词:LTSL 程序;字体排列图;排列项目;字符定位位置;旋转角度

中图分类号:TQ330.4⁺1;TP311.52 **文献标识码:**B **文章编号:**1006-8171(2005)09-0531-04

LTSL 是用 Visual Basic 6.0 开发的有编辑、显示、输出功能和多种字体数据库的轮胎胎侧字体排列图辅助设计软件,能迅速、方便、灵活、独立地完成设计工作^[1]。本文介绍 LTSL 中的主要数据结构。

1 LTSL 的数据结构

每一个应用软件都会使用大量的变量,定义的变量类型和数量主要取决于软件的功能。LTSL 可以编辑修改每个排列项目整体特性,如字体、大小和排列位置等,因此需为每一个排列项目定义如字体名称、字符高度、字符宽度、定位位置坐标等参数。LTSL 也可以单独编辑修改排列项目中的字符特性,因此也要为每个字符定义字体名称、字符高度、字符宽度、定位基准点坐标和绕字体排列图中心点旋转角度等参数。

在编程过程中,为单个字符、排列项目和整个字体排列图各定义了一个数据结构,具体数据结构定义如下。

```
Public Type UserLetterInfo '单个字符数据结构
    FontName As String '字体名称
    Height As Double '字符高度
    Width As Double '字符宽度
    LetterGap As Double '与前一字符间距
    BasePointX As Double '定位基准点 X 坐标
```

```
BasePointY As Double '定位基准点 Y 坐标
RotateAngle As Double '绕字体排列图中心点旋转角度
ObliqueAngle As Double '字符倾斜角度
LineWidth As Double '字符笔划轮廓线条宽度
Letter As String '字符内容
Ratio As Double '字符高宽比
HatchLineGap As Double '装饰性锯齿线条间距(0 表示没有锯齿线)
HatchAngle As Double '装饰性锯齿线条角度
HatchLineWidth As Double '装饰性锯齿线条宽度
HatchDepth As Double '装饰性锯齿线凸起高度
HatchStartGap As Double '装饰性锯齿线起始位置调整值
ProfileLine As EntityGroup '字符外形轮廓线
InitialValue As Boolean '是否要对字符参数数值初始化
.....
End Type
Public Type UserItemInfo '排列项目数据结构
    Type As String '排列项目类型名称
    Content As String '排列项目字符内容
    Text As String '排列项目字符内容(已经
```

作者简介:陈振艺(1961-),男,上海人,上海轮胎橡胶(集团)股份有限公司轮胎研究所工程师,主要从事轮胎结构设计和计算机辅助设计软件开发工作。

去除空格)

FontName As String '排列项目字体名称

Height As Double '排列项目字符高度

Ratio As Double '排列项目字符高宽比

LetterGap As Double '字符间距

SpaceGap As Double '空格所代表的字符
间距

Depth As Double '字符凸起高度

CircumferentialAlign As Long '周向对齐
方式(左、中、右)

AlignAngle As Double '周向对齐角度

RadialAlign As Long '径向对齐方式(顶、
中、底)

AlignRadius As Double '径向对齐半径

FrameHeight As Double '活络块高度(0
表示无活络块)

FrameCenterLength As Double '活络块周
向中心线长度

FrameMidRadius As Double '活络块径向
中心半径

FrameIncludeAngle As Double '活络块包
含角度

FrameCenterAngle As Double '活络块中
心对齐角度

ProfileLine As EntityGroup '排列项目外
形轮廓线

AlignLine As EntityType '排列项目径向
对齐基准线

LineWidth As Double '字符笔划轮廓线条
宽度

ObliqueAngle As Double '字符倾斜角度

Length As Double '排列项目沿径向对齐
基准线的圆弧长度

IncludeAngle As Double '排列项目包含
角度

StartAngle As Double '排列项目起始角度

CenterAngle As Double '排列项目中心角度

StartPointX As Double '排列项目起始基
准点 X 坐标

StartPointY As Double '排列项目起始基
准点 Y 坐标

HatchLineGap As Double '装饰性锯齿线
线条间距(0 表示没有锯齿线)

HatchAngle As Double '装饰性锯齿线线
条角度

HatchLineWidth As Double '装饰性锯齿
线线条宽度

HatchDepth As Double '装饰性锯齿线凸
起高度

HatchStartGap As Double '装饰性锯齿线
起始位置调整值

.....

LetterNumber As Long '排列项目的字符
数量(不包括空格)

LetterInfo() As UserLetterInfo '排列项目
所属字符变量信息

End Type

Public Type UserLayoutInfo '整个排列图数据
结构

Specification As String '轮胎规格

PatternType As String '花纹类型

OutsideRadius As Double '排列范围外圈
半径

InsideRadius As Double '排列范围内圈半径

LayoutStyle As String '排列图式样

SingleLoad As Long '单胎负荷

DualLoad As Long '双胎负荷

SinglePressure As Long '单胎压力

DualPressure As Long '双胎压力

.....

TubeType As String '内胎形式

ZoomScale As Double '视图放大倍数

ZoomLeft As Double '视图左边边界

ZoomTop As Double '视图顶部边界

ZoomRotateAngle As Double '视图旋转
角度

CurItemNumber As Long '当前被选择的
排列项目编号

CurLetterNumber As Long '当前被选择
的字符编号

FileName As String '文件保存名称

PasswordString As String '密码

```

FileSavePath As String '文件保存路径
MaxUndoNumber As Long '最大可恢复
(Undo)数量
.....
ItemNumber As Long '排列项目数量
ItemInfo() As UserItemInfo '所有排列项目
变量信息
End Type

```

需要说明的是,上述数据结构中省略了部分参数;排列项目数据结构和单个字符数据结构中有一些重复变量,如字体名称、字符高度等,这部分变量是用来编辑修改排列项目整体特性以及初始化各字符变量数值的(用来计算和定位字符的是各个字符的相关变量);排列项目数据结构中包含角度、起始角度和中心角度3个变量,通常只需知道其中两个,第3个可由另外两个计算出来;为方便编辑修改时的调用,计算定位排列项目时就赋值给变量,可以节省计算时间。

2 数据结构的使用

以计算字符定位位置和旋转角度为例。字符的定位及基本数据如图1所示,其定位基准点可以为字符中心,也可以位于一角,在LTSL中字符的定位基准点位于排列项目径向对齐基准线上,当径向对齐方式采用底部对齐时,定位基准点为字符的左下角,当径向对齐方式采用顶部对齐时,定位基准点为字符的左上角。假定排列项目数据结构中的基本参数(如字符高度、字符宽度、字符间距、对齐方式、角度和半径等)已经赋值,因字符宽度和字符间距均为直线距离,需换算为弧长,因此先定义一个换算函数:

```

Public Function ChordToArc (ByVal Chord As Variant,
ByVal Radius As Variant) As Double
    ChordToArc = 2 * Arcsin(Chord/2/Radius) * Radius
End Function

```

然后为每个字符数据结构中的变量赋值(即初始化)。赋值程序如下。

```

With LayoutInfo. ItemInfo (LayoutInfo. CurItemNumber)
    For i = 1 To .LetterNumber

```

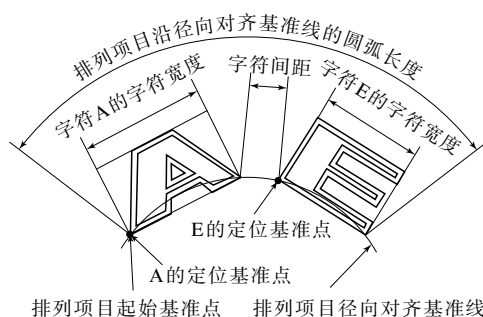


图1 字符的定位及基本数据示意

```

.LetterInfo(i). Height = . Height
.LetterInfo(i). LetterGap = . LetterGap
.....
Next i
.LetterInfo(1). LetterGap = 0 '第1个
字符的该变量的值为零
End With

下面的代码计算 Length 变量(排列项目沿径向
对齐基准线的圆弧长度)和3个角度。
With LayoutInfo. ItemInfo (LayoutInfo. CurItemNumber)
For i = 1 To .LetterNumber
    .Length = .Length + ChordToArc(. LetterInfo(i). Width, . AlignRadius) + .ChordToArc (. LetterInfo (i). LetterGap, . AlignRadius)
Next i
.IncludeAngle = .Length/. AlignRadius
If .CircumferentialAlign = 0 Then '周向左
对齐
    .StartAngle = .AlignAngle
ElseIf .CircumferentialAlign = 1 Then '周向
中心对齐
    .StartAngle = .AlignAngle + .IncludeAngle/2
Else '周向右对齐
    .StartAngle = .AlignAngle + .IncludeAngle
End If
.CenterAngle = .StartAngle - .IncludeAngle/2
End With

```

计算各字符的定位基准点和旋转角度程序的

```
代码如下(Angle1, Angle2 和 r1 为临时变量, x0  
和 y0 是字体排列图中心点坐标, hpi 代表  $\pi/2$ )。  
Angle2 = LayoutInfo. ItemInfo(LayoutInfo. Cur  
ItemNumber). StartAngle  
r1 = LayoutInfo. ItemInfo(LayoutInfo. Cur  
ItemNumber). AlignRadius  
For i = 1 To LayoutInfo. ItemInfo(LayoutInfo.  
CurItemNumber). LetterNumber  
With LayoutInfo. ItemInfo(LayoutInfo. Cur  
ItemNumber). LetterInfo(i)  
Angle2 = Angle2 - ChordToArc(. Letter-  
Gap, r1)/r1  
Angle1 = ChordToArc(. Width, r1)/r1/2  
.BasePointX = x0 + Cos(Angle2) * r1  
.BasePointY = y0 + Sin(Angle2) * r1  
.RotateAngle = Angle2 - Angle1 - hpi  
Angle2 = Angle2 - 2 * Angle1  
End With  
Next i
```

3 程序特点

从源代码中可以看出, 计算字符定位位置和旋转角度使用的是各个字符的变量参数, 而不是排列项目中的变量参数, 因此无论是初次生成排列项目, 还是编辑修改排列项目, 均可共用一个子

程序, 且能保存修改过的数据。

字符的数据结构声明在排列项目数据结构中, 同样, 排列项目的数据结构声明在字体排列图数据结构中, 这样可使代码层次分明, 数据从属关系明确, 调用时不容易出错, 且保存或读取数据文件也非常方便, 只需一条指令(Put/Get)即可。

排列项目和字符的数据结构中都有 Profile-Line(外形轮廓线)变量, 该变量是鼠标在屏幕上双击选择排列项目或字符时使用的, 在生成/修改排列项目中各字符定位位置时, 需预先计算各排列项目和字符的外形轮廓线, 一旦有鼠标双击事件, 就可确定鼠标位置是否落在某一外形轮廓线内, 若不预先算好而采用临时计算的方法, 则要增加许多编程工作量和代码, 且操作时可能会有反应迟钝的感觉。

4 结语

定义数据结构, 并按层次声明和调用, 能使参数的使用和管理有条不紊, 增强代码的通用性, 方便参数的修改和添加, 从而保证 LTSL 代码的正确性和软件顺畅使用。

参考文献:

- [1] 陈振艺. 轮胎胎侧字体排列图辅助设计软件 LTSL 简介[J]. 轮胎工业, 2005, 25(8): 467-470.

收稿日期: 2005-04-11

美国国家公路交通安全局颁布

最终轮胎气压法规

中国分类号: TQ336.1 文献标识码: D

印度《印度橡胶杂志》2005年89卷28页报道:

美国国家公路交通安全局(NHTSA)于2005年3月颁布了有关在美国销售的所有新轿车上安装轮胎气压监控装置(TPMS)的最终法规。

NHTSA说, 汽车公司必须安装当汽车上一条或几条轮胎的气压低于推荐用气压25%以上时能够测得掉压情况的装置。

对于这一法规, 橡胶加工者协会(RMA)立即表示强烈的反对。它认为, 最终法规提出的25%的标准可能导致轮胎欠压达到不能安全地承载满负荷车辆的程度。按照NHTSA的标准, 司机可能会用欠压相当严重、但尚未达到报警程度的轮

胎行驶数千英里。

RMA主席Donald Shea说, 协会支持安装及时向司机提供低压警报的轮胎气压报警器, 但是协会认为, 具体规定可能在轮胎实际上已严重欠压时给司机造成其轮胎气压仍是正常的错误安全感。

去年9月NHTSA颁布的有关这一问题的原始法规受到美国轮胎行业 and 用户安全机构的广泛批评。

据NHTSA说, 新法规将于联邦登记处发布30日以后生效。该法规要求汽车公司在所有毛重4 536 kg以下的轿车上安装TPMS, 从2006型车开始, 到2008型全部新轿车都要安装TPMS。启动期从今年9月开始。

(涂学忠摘译)