

二维图形绘制程序

Ming

2008-10-29

二维图形绘制的程序都包括 $F(x)$ 图象绘制、参数曲线绘制、曲线拟合、色彩深度三维。下来将对其逐个分析。

1. $F(x)$ 图象绘制

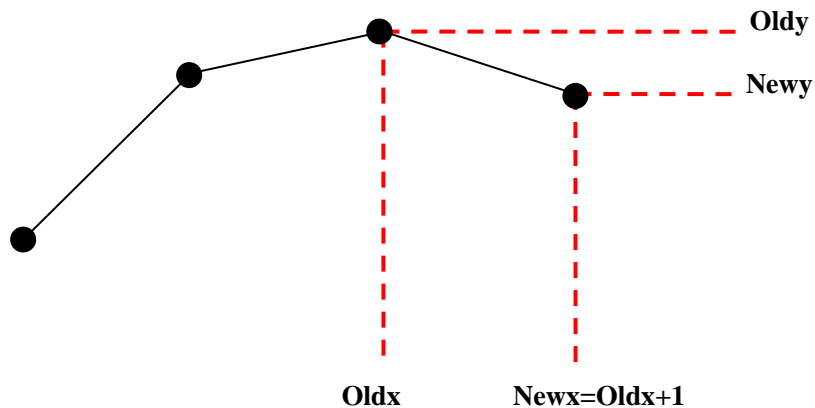
$F(x)$ 图象绘制依赖表达式计算树，在构建完相应的表达式计算树后，对其添加变量 “x” 和常量 “pi”、“e”。随后计算屏幕 x 点对应的函数 x 值，然后计算一个函数 y 值，将其转化为屏幕 y 值，这个过程如下：

```
Fx=SxtoFx(Sx)
SetVar("x",Fx)
Fy=ComputeTree(Root)
Sy=FytoSy(Fy)
```

在获取到第一个 y 值以后对逐个 x 点进行扫描（从 1 到屏幕宽），然后对每个 x 计算屏幕 y 值，与上一个点进行连线。（如果计算出的屏幕 y 值超出屏幕则不绘制该条直线）

```
g.drawline(Oldx,Oldy,Newx,Newy)
```

完成最终点扫描后即得到 $F(x)$ 图象。绘制方式如下图所示：



2. 参数曲线绘制

参数曲线绘制大致与 $F(x)$ 图象绘制类似，不同在于这里需要两个表达式， $F_x(t)$ 和 $F_y(t)$ ，注册的变量是 t 而不是 x 。绘制的时候是按照步长从 t_{min} 走到 t_{max} ，分别计算 F_x 、 F_y ，再转换成 S_x 、 S_y 来绘制直线。

3. 曲线拟合

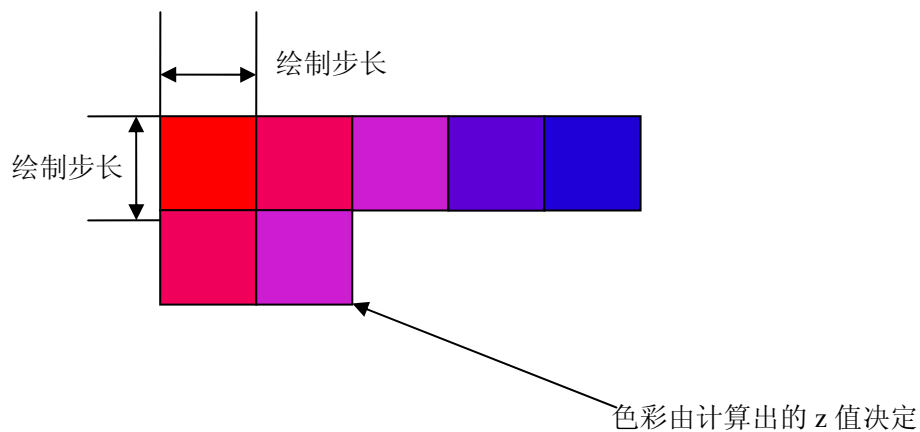
曲线拟合在拟合次数为样本点个数减一的时候即为多项式插值，采用计算均差的方法计算出均差数组，再转为多项式各次的系数。低次的均为解矛盾方程组来获得各次系数。

绘制曲线时即与 $F(x)$ 图象绘制类似，但计算不依赖表达式，而是依赖于前面计算出的系数来计算函数值。

4. 色彩深度三维

色彩深度三维是通过颜色来表示第三位的方式，把三维图像放到了二维中，计算的时候依赖表达式求值，注册有两个变量 x 、 y 。扫描的时候按绘制步长走屏幕像素。

绘制的方式如下图所示：



绘制的色彩为 $((\text{int})(255 * \text{tmp}), 0, (\text{int})(255 - 255 * \text{tmp}))$ ，其中 $\text{tmp} = (Fz - \text{MinZ}) / (\text{MaxZ} - \text{MinZ})$ 。染色的时候使用 `g.fillRect(Sx, Sy, Step, Step)` 完成方格染色绘制。