

# 浙江大学

课程名称：                    计算机动画                    

姓    名：                    肖  鸢                    

学    院：                    计算机学院                    

专    业：                    数字媒体技术                    

学    号：                    3150102459                    

指导教师：                    于金辉                    

2017  年    10 月  14 日

# 浙江大学实验报告

课程名称：\_\_\_\_ 计算机动画 \_\_\_\_ 实验类型：\_\_\_\_ 综合 \_\_\_\_

实验项目名称：\_\_\_\_ 路径曲线与运动物体控制 \_\_\_\_

学生姓名：\_\_\_\_ 肖鸢 \_\_\_\_ 专业：\_\_\_\_ 数字媒体技术 \_\_\_\_ 学号：\_\_\_\_ 3150102459 \_\_\_\_

同组学生姓名：\_\_\_\_ 无 \_\_\_\_ 指导老师：\_\_\_\_ 于金辉 \_\_\_\_

实验地点：\_\_\_\_ 机房 \_\_\_\_ 实验日期：\_\_\_\_ 2017 年 10 月 14 日 \_\_\_\_

## 一、 实验目的和要求

- (1) 设计并实现一个路径控制曲线。
- (2) 了解动画动态控制的基本原理和方法，提高相关动画编程能力。

## 二、 实验内容和原理

### 内容

- (1) 掌握 Cardinal 样条曲线的表示和算法，了解控制参数对曲线形状的影响。

( 2 ) 对照 Cardinal 样条曲线的数学表示和程序代码的对应关系。

( 3 ) 在路径曲线上放置一小汽车，使其在路径上运动起来，汽车运动速度可调

## 原理

在动画中可用样条曲线来表示运动路径以及不规则物体的形状。通用的插值表达式为：

$$P(u) = U^T M B$$

其中

$$U^T = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix}, B = \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix}$$

## 三、 实验器材

Qt Creator 5.7.0 @ Win7

## 四、 实验步骤

### (1) 界面设计

1. 界面能够适应不同的大小显示要求并且能够保持基本的位置关系不变，不出现遮挡等影响用户体验的情况
2. 从美观角度考虑进行 icon 的重新设置和按键外观的重新设置
3. 为美化界面，插入一个与主题相关的 gif 动画

### (2) 鼠标事件处理

这里使得在每次点击的时候，都能直接生成一个点，并且与前一个点相连，使得对用户来说点击的先后关系比较明确并且方便在后期进行插值的时候进行观察。

### (3) Cardinal 曲线插值

1. 根据输入的 tension 初始化 Cardinal 曲线对应的矩阵
2. Cardinal 曲线插值的时候先需要对边界条件进行处理，起始点表达式如下

$$M = \tau \begin{bmatrix} -1 & 2/\tau - 1 & -2/\tau + 1 & 1 \\ 2 & -3/\tau + 1 & 3/\tau + 1 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 1/\tau & 0 & 0 \end{bmatrix}$$

末点表达式类似。

3. 利用 Cardinal 插值表达式逐段进行插值，输入的 grain 值决定了插值点的数量
4. 将上一步骤插值所得点直线连接。并且根据控制显示点数量的 density 数值将部分插值点标明以方便后续观察。

### (4) 更多曲线插值

1. 在这里希望能够对于更多的插值拟合方式有一个理解，因此增加了 Blended Parabolas、三阶 B 样条插值和 Bezier 三种不同的插值方式。
2. Blended Parabolas 对应的 M 矩阵为

$$M = \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix}$$

处理方式和 Cardinal 类似

3. 三阶 B 样条插值对应的 M 矩阵为

$$M = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

在这里不像 Cardinal 需要进行端点补齐的操作

4. Bezier 曲线对应的 M 矩阵为

$$M = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}$$

Bezier 和前几种方式相比，点的含义有所不同所以具体呈现效果也有比较大的不同

## (5) 界面清除

清除已有的点、线、插值的记录，准备下一次绘制。

## (6) 运动插值

1. 根据插值点计算出每个插值点对应的旋转角度
2. 根据所接收的运动速度大小以及运动情况（匀速、加速、减速）

确定小车在每个位置对应的运动关系

其中加速所使用的表达式为：

$$v = \sin(PI * u/2)$$

减速为：

$$v = \cos(PI * u/2)$$

## **(7) 小车运动**

1. 对于旋转角度进行处理，增加水平翻转后的图像，使得人物的运动能不发生首尾颠倒。
2. 为了表达的生动性，动态呈现小车（人物）的运动（简单来说就是呈现 gif）。
3. 增加对于垂直运动所在的临界状态的处理，使得小车（人物）不发生剧烈的抖动，使得观看更流畅。

## **(8) 弧长参数化处理**

使用 Simpson 公式进行重新的处理，使得插值点按距离均匀分布。让动画的运动速度不会受到采样的频率影响而突变。

## **(9) 计时**

在插值和参数化的时候，进行时间的统计，使得后期对于运行速度有一个直观的判断。

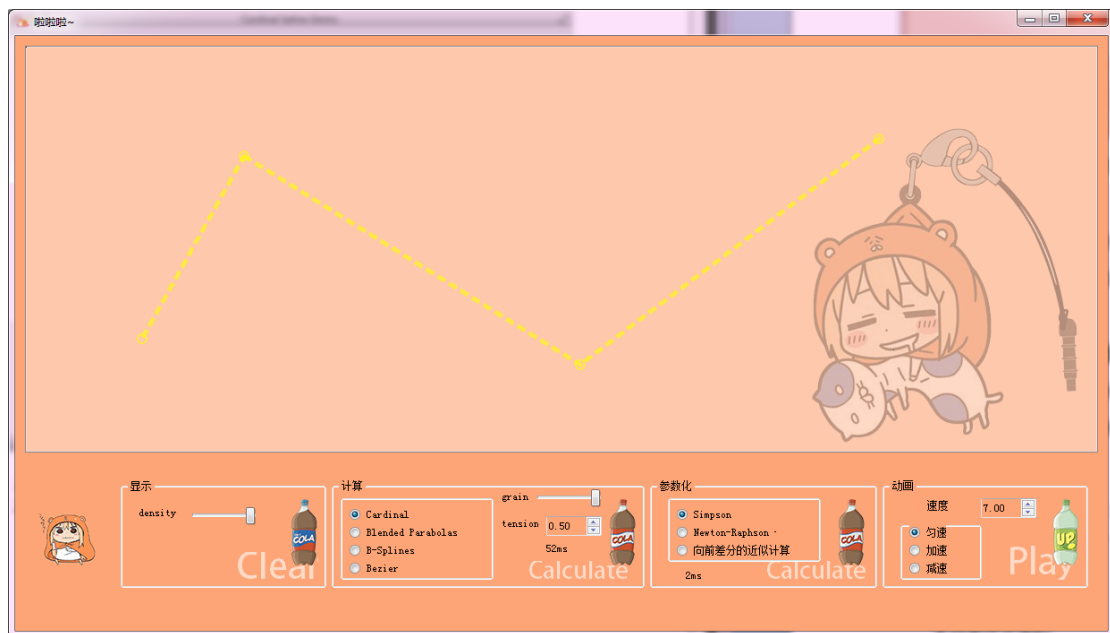
# **五、 实验结果分析**

## **(1) 界面设计**



图中左下角人物形象会自动变化。

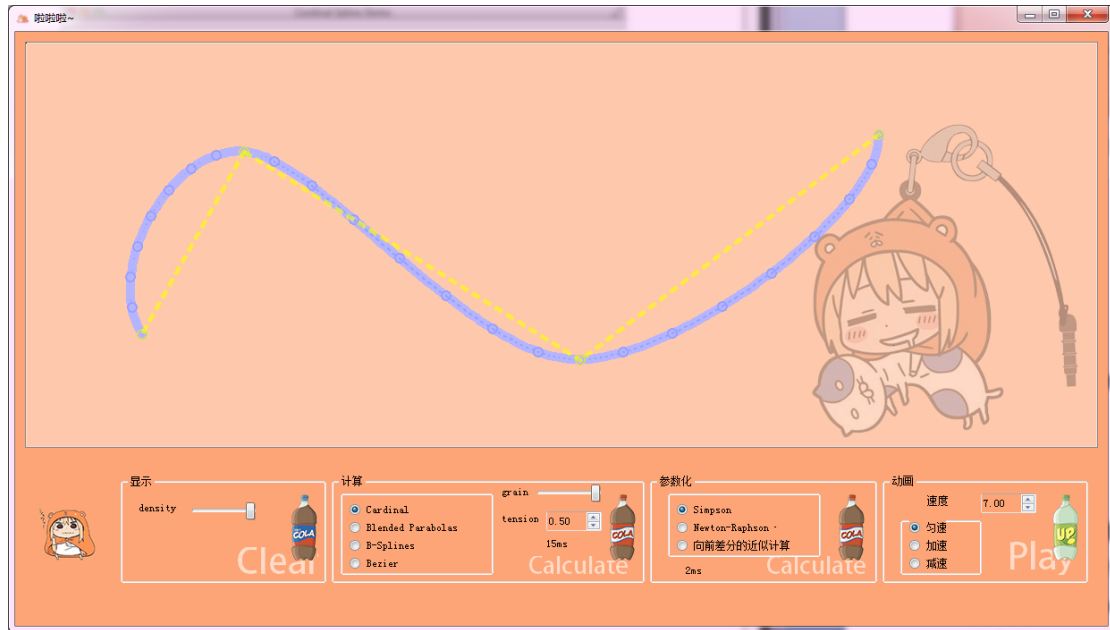
## (2) 鼠标事件处理



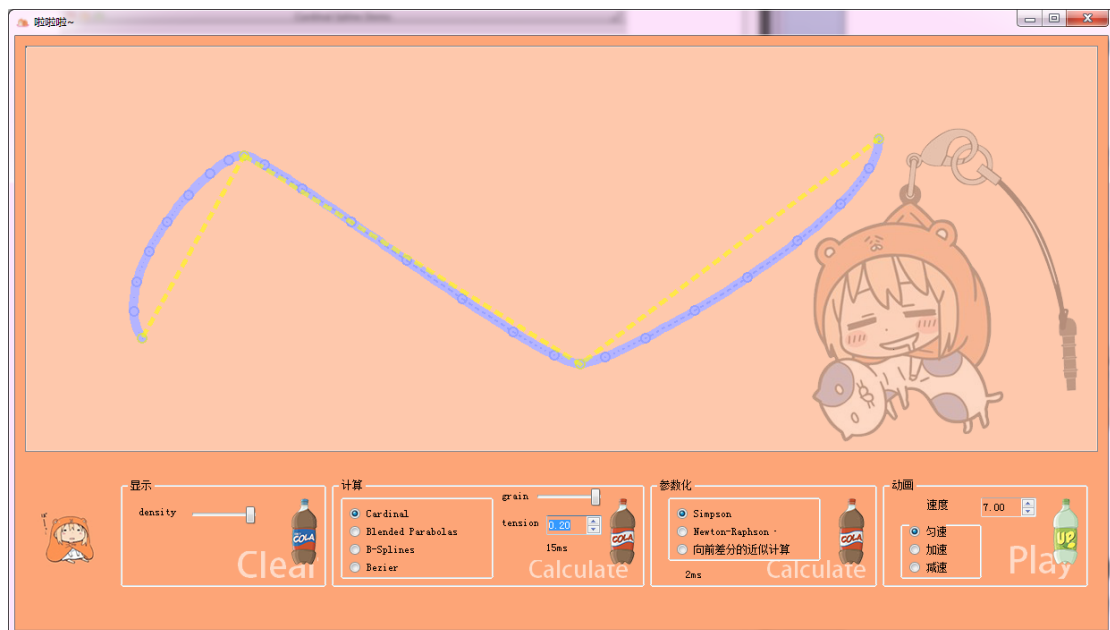
## (3) Cardinal 曲线插值

1. tension 不同

tension = 0.5

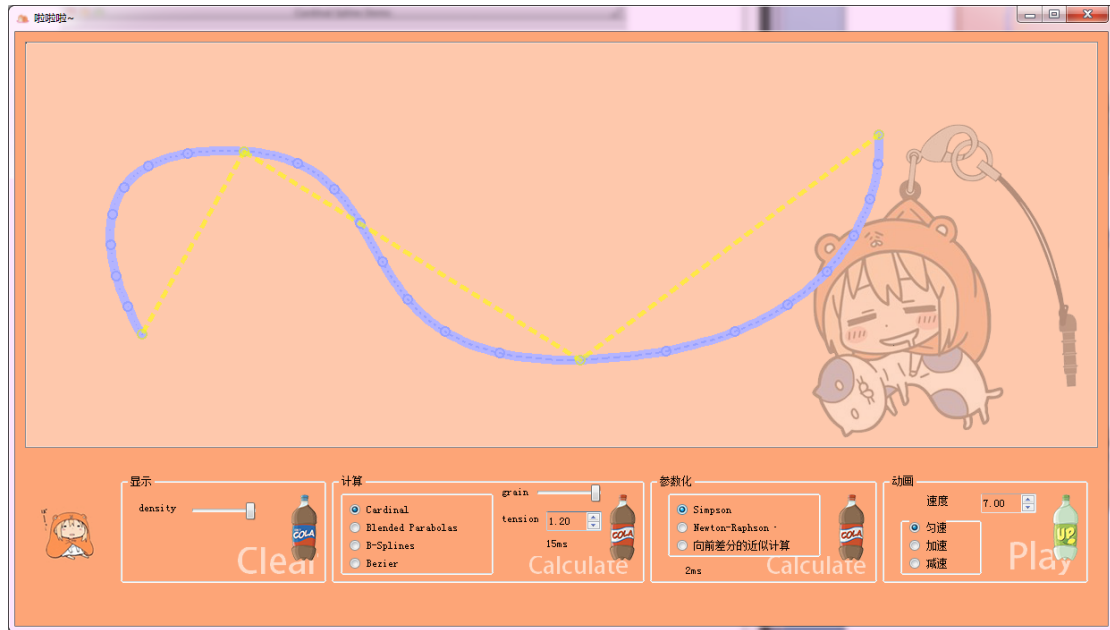


tension = 0.2



tension = 1.2



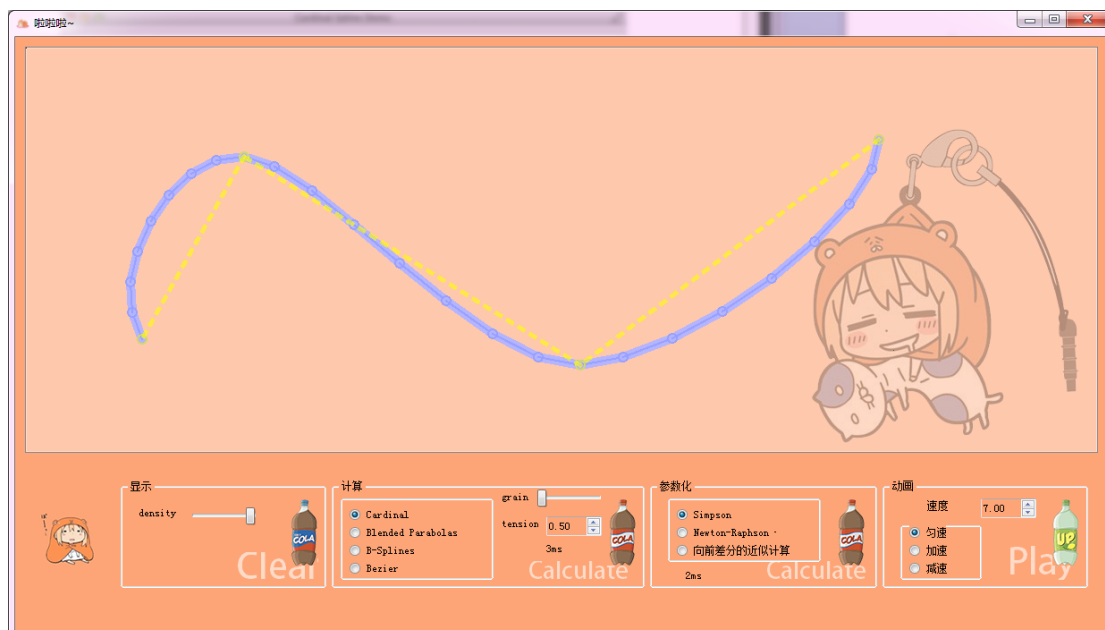


可以看到 tension 控制的是关键点附近的切线变换，tension 值越大切线变化越小。

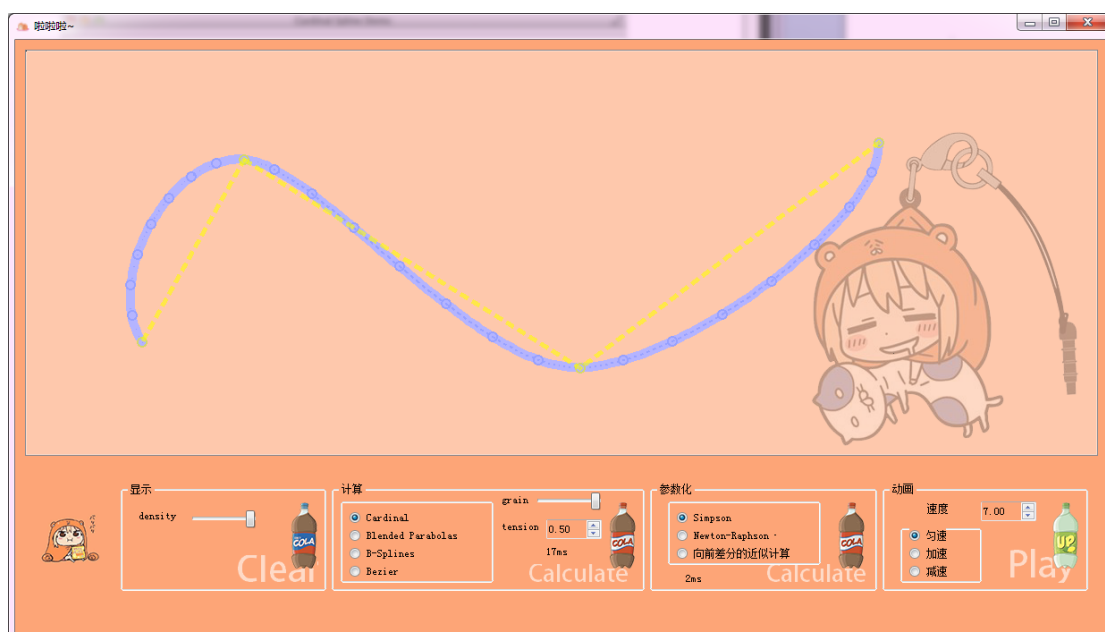
对于 tension 的含义我猜测更多取决于取关键点的风格——如果取关键点是根据，在速度平稳的区域取样更多的话可以取较大的 tension 值，相反可以取较小的 tension 值。

## 2. grain 不同

grain = 8



grain = 64

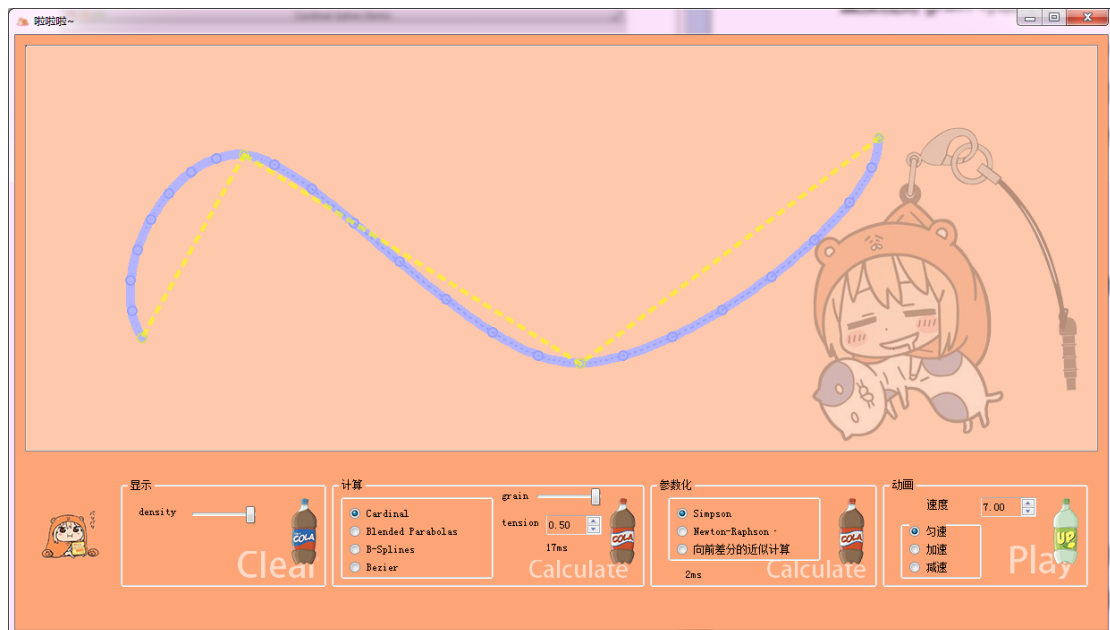


(这里上面显示的圆圈数量是有 density 控制的,从图上如果想直接观测 grain 可以观察细的紫线的分段。)

可以看到,对于大的 grain 值,计算速度更慢,但是曲线更加平滑。在 grain 值为 8 的时候还可能从图上看出明显的直线连接插值点的痕迹,而在 grain 为 64 的时候已经很难看出来了。

### 3. density 不同

density = 8



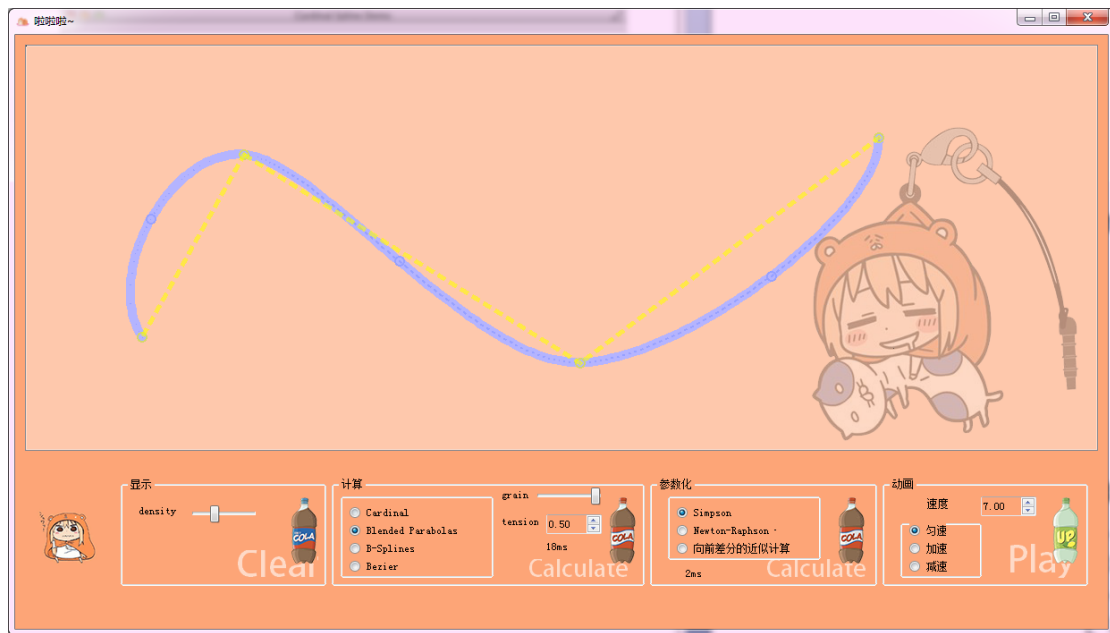
density = 1



density 控制的是对应用户界面显示的插值点数量

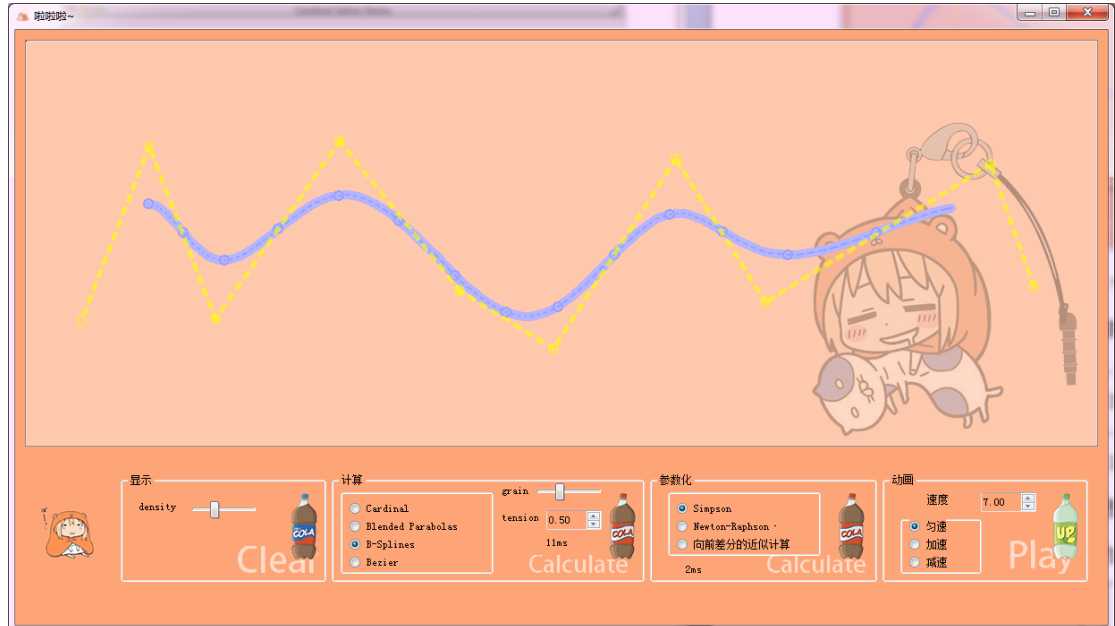
### (4) 更多曲线插值

## 1. Blended Parabolas

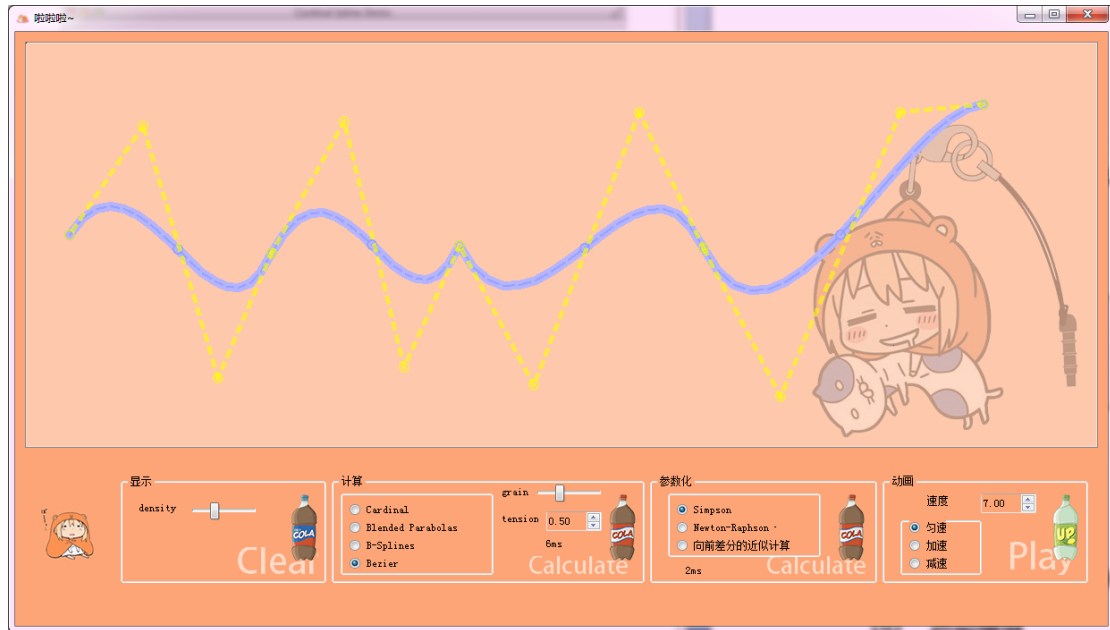


其公式形式和 Cardinal 插值， $\tau = 0.5$  的情况相似。

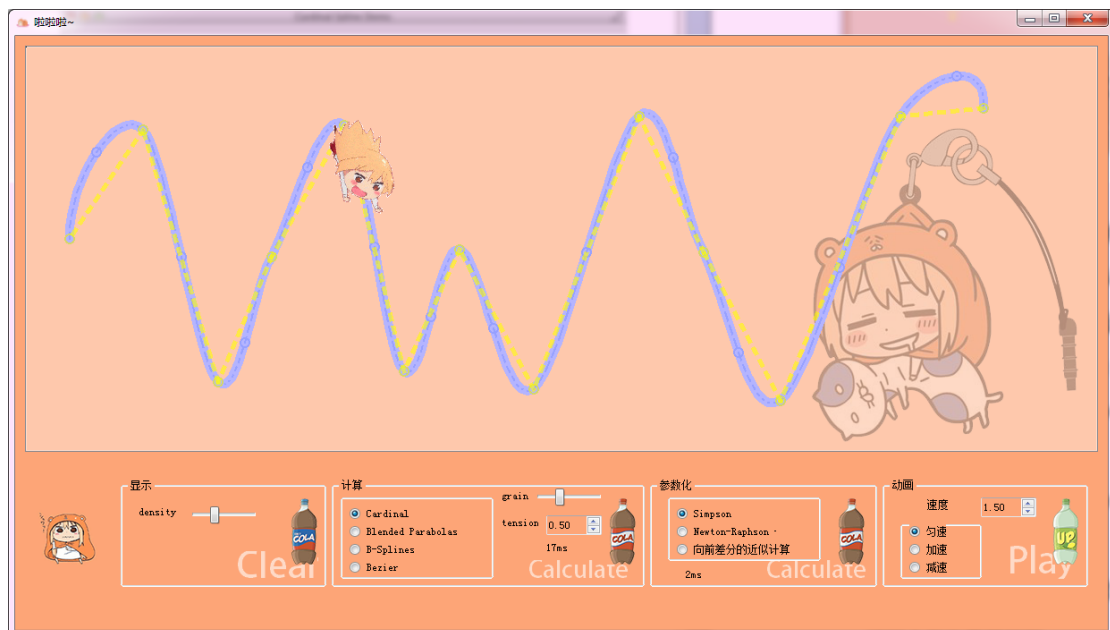
## 2. 三阶 B 样条插值



## 3. Bezier 曲线



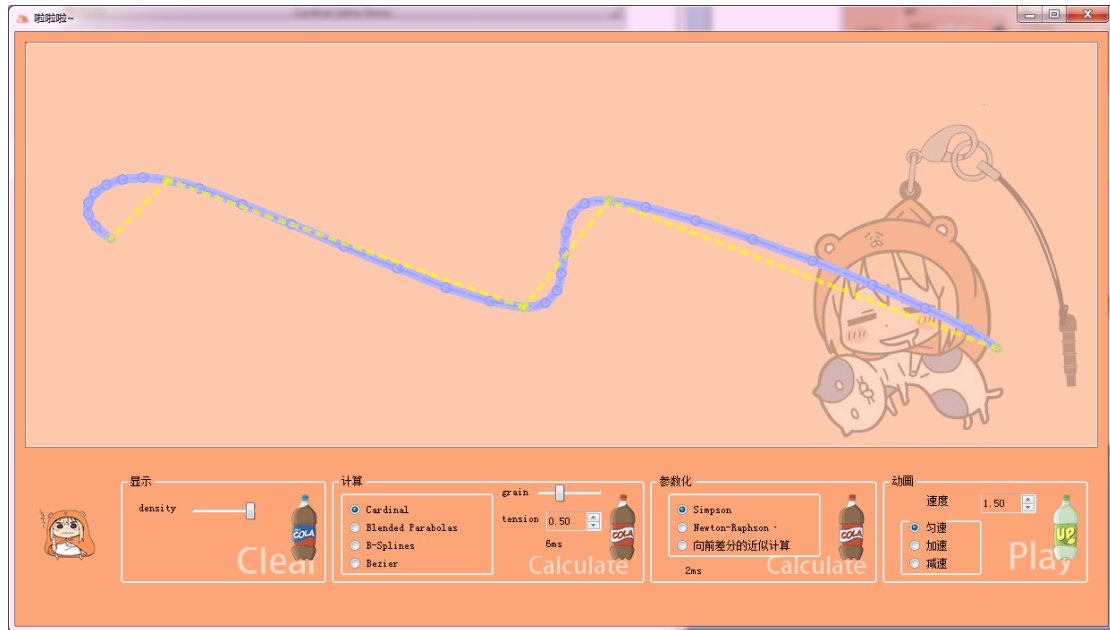
## (5) 运动



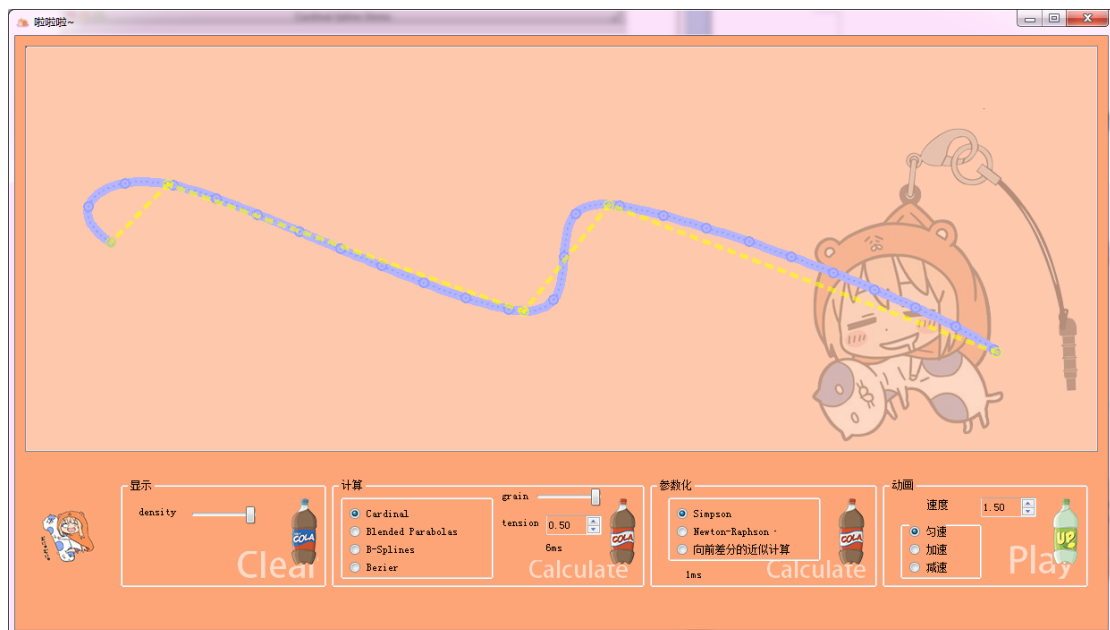
运动具体情况参见录屏。

## (6) 弧长参数化处理

未参数化



## 参数化



## 六、 实验感悟与问题

最后在实现参数化的时候跨段的处理比较粗糙，然后由于时间问题最后参数化的部分没有细调，并且下面两种类型没有实现。

在这个期间遇到了很多在具体实现之前没有办法理解的事情，比如使用动画在纵向的运动过程中会有很明显的抖动，并且向左向右的

运动需要的调整不单单是角度还有水平方向的翻转。

在确定使用 Blended Parabolas 曲线的时候觉得这种曲线的设计思路很神奇，在后来才发现他和 Hermite 曲线的格式是一致的。这应该是由于三次曲线的一次光滑性的限制。在思考这个实验的过程中也对各种插值算法有了进一步的思考。

本来想尝试对三维模型进行运动插值，后来由于时间和工程量的限制没有实现。

总的来说在这次实验中对 QT 的种种设置有了进一步的熟悉，并能更好的理解运动插值算法，并且有了对自己想法的初步实现。