



(12)发明专利申请

(10)申请公布号 CN 109857322 A

(43)申请公布日 2019.06.07

(21)申请号 201910068904.0

(22)申请日 2019.01.24

(71)申请人 广东第二师范学院

地址 510303 广东省广州市海珠区新港中路351号

(72)发明人 梁嘉慧 卢燕萍 陈怡航 张淑莲
谢敏峰 朱雄泳 陈忆群 黄润

(74)专利代理机构 广州粤高专利商标代理有限公司 44102

代理人 林丽明

(51)Int.Cl.

G06F 3/0488(2013.01)

G06T 11/20(2006.01)

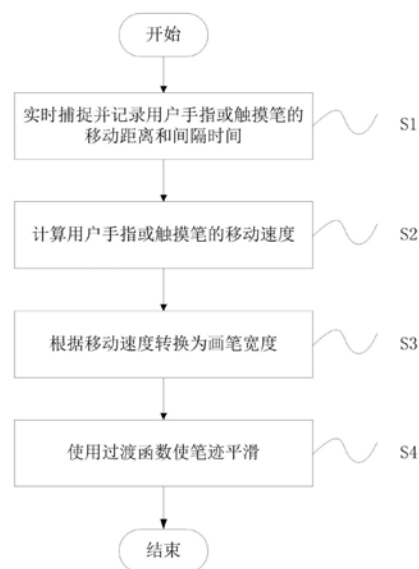
权利要求书2页 说明书5页 附图5页

(54)发明名称

一种基于安卓的画笔宽度控制方法和装置

(57)摘要

本发明公开了一种基于安卓的画笔宽度实时控制方法,包括以下步骤:S1:实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间;S2:计算用户手指或触摸笔的移动速度;S3:根据移动速度转换为画笔宽度;S4:使用过渡函数使笔迹平滑。本发明通过线性插值建立起速度和画笔宽度的转换模型,实现了毛笔的笔锋的变化。通过使用安卓自带的绘图API可以快速实现所需功能,更好适配移动端设备,增加代码可读性,通过线性插值建立起来的数学模型,在速度-时间模型基础上,优化了速度与画笔宽度转化的算法,在不增加代码复杂度的情况以及计算量的前提下更好模拟出毛笔粗细不一的笔迹的变化。



1. 一种基于安卓的画笔宽度实时控制方法,其特征在于,包括以下步骤:

S1:实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间;

S2:利用移动距离和间隔时间计算用户手指或触摸笔的移动速度;

S3:根据移动速度转换为画笔宽度;

S4:使用过渡函数使笔迹平滑。

2. 根据权利要求1所述的一种基于安卓的画笔宽度实时计算方法,其特征在于,所述步骤S1中实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间包括如下步骤:

S1.1:初始化当前绘制点坐标 (x_0, y_0) 、前一绘制点坐标 (x, y) ,均初始化为原点 $(0, 0)$,判断当前绘制点是否为初始值 $(0, 0)$;

S1.2:判断当前绘制点是否为初始值 $(0, 0)$,即判断是否为初始绘制点;若判断为初始绘制点,执行步骤S1.3;若判断为不是初始绘制点,执行步骤S1.4;

S1.3:实时获取当前绘制点坐标 (x_0, y_0) ,并将其赋值给前一绘制点坐标 (x, y) ;利用Android的Date对象获取当前时刻 t_0 ,并将其赋值给前一时刻 t ,返回步骤

S1.2;

S1.4:实时获取当前绘制点坐标 (x_0, y_0) ,利用安卓的Date对象获取当前时刻 t_0 ,完成步骤S2、S3、S4后,并将 (x_0, y_0) 赋值给前一绘制点坐标 (x, y) , t_0 赋值给前一时刻 t ,返回步骤S1.2。

3. 根据权利要求2所述的基于安卓的画笔宽度实时计算方法,其特征在于,所述步骤S2计算用户手指或触摸笔的移动速度,包括以下步骤:

S2.1:根据距离公式 $S = \sqrt{(x_0 - x)^2 + (y_0 - y)^2}$ 计算移动距离 S ;

S2.2:计算移动距离为 S 时的间隔时间 T :

$T = t_0 - t$

S2.3:计算移动速度 V :

$V = S/T$ 。

4. 根据权利要求3所述的基于安卓的画笔宽度计算方法,其特征在于,所述步骤S3根据移动速度转换为画笔宽度,包括以下步骤:

S3.1:建立以移动速度 V 为 x 轴,以画笔宽度 W 为 y 轴的坐标系,令 (V_{\min}, W_{\min}) 表示初态, (V_{\max}, W_{\max}) 表示终态,其中, V_{\min} 为用户手指或触摸笔的最小移动速度,移动速度小于等于 V_{\min} 时,画笔宽度为最小画笔宽度 W_{\min} 并保持不变; V_{\max} 为用户手指或触摸笔的最大移动速度,移动速度大于等于 V_{\max} 时,画笔宽度为最大画笔宽度 W_{\max} 并保持不变;

S3.2:初始化当前画笔宽度 W_x 、前一画笔宽度 W_b ,并将其初始化为同一的初始值,根据线性插值公式,推导得出移动速度与画笔宽度的公式,所述线性插值公式为:

$$y = y_a + (y_b - y_a) \frac{x - x_a}{x_b - x_a}$$

该式表示过点 (x_a, y_a) 和点 (x_b, y_b) 的直线,由该式得到:

$$W_x = W_{\min} + W_{\max} \bullet \frac{V - V_{\min}}{V_{\max}}。$$

5. 根据权利要求4所述的基于安卓的画笔宽度计算方法,其特征在于,所述步骤S4使用

过渡函数使笔迹平滑,包括以下步骤:

S4.1:判断画笔是否为初次设置,即判断前一画笔宽度是否等于初始值;

S4.2:若前一画笔宽度等于初始值,直接把画笔宽度设置为当前画笔宽度 W_X ;

若前一画笔宽度不等于初始值,利用过渡公式设置画笔宽度,所述过渡公式为:

$$W_c = W_X \cdot \frac{1}{3} + W_B \cdot \frac{2}{3}$$

式中, W_c 为平滑后的画笔宽度。

6.根据权利要求5所述的基于安卓的画笔宽度计算方法,其特征在于,步骤S3.1中 $V_{\min}=0.1$, $V_{\max}=5$, $W_{\min}=10$, $W_{\max}=70$,步骤S3.2中初始值为-1。

7.一种基于安卓的画笔宽度计算装置,其特征在于,包括:

记录模块,用于实现实时捕捉记录用户手指或触摸笔移动距离和间隔时间;

计算模块,用于计算用户手指或触摸笔的移动速度;

转换模块,根据移动速度转换为画笔宽度;

过渡模块,使用过渡函数使笔迹平滑。

一种基于安卓的画笔宽度控制方法和装置

技术领域

[0001] 本发明涉及安卓绘图领域,更具体地,涉及一种于安卓的画笔宽度控制方法和装置。

背景技术

[0002] 控制安卓(Android)平台画笔粗细并使笔画更平滑已经运用到许多商业产品中,比如:各种输入法、各种字帖类软件等,同时Android自带的绘图API也可以快速实现绘制功能。然而在实际应用中,大多数书写软件写出的字迹往往粗细不明显并且显的生硬不够平滑圆润,canvas实现的绘制功能甚至无法体现笔迹的粗细变化,而使用一种相对简单廉价的方式可以解决这个限制,就是利用基于插值数学模型和用户手指移动速度实时计算绘制画笔宽度的方法。然而传统的改变画笔大小的算法主要有两种:1.基于物理模拟的毛笔生成算法,其原理是模拟毛笔参数写出字体,虽然也可以较好的模拟书法的效果,但存在计算量大,实时性差,且需要人为设置和调整复杂的参数;2.基于机器学习的样本采集生成算法,这类算法的基本思路利用机器学习大量数据样本中搜索与用户笔迹相似的样本字体,再绘制画板上,但也存在计算量大且构建神经网络及训练样本集的操作都较复杂,时间需求大。而利用移动速度实时计算绘制画笔宽度的方法可避免复杂的计算量以及人为设置和调整复杂的参数,捕获得到用户手指移动速度,并利用线性插值推导出相应公式优化转换数学模型,从而实现较好的绘制效果。

发明内容

[0003] 本发明旨在至少在一定程度上解决上述技术问题。

[0004] 本发明的首要目的是提供一种基于安卓的画笔宽度控制方法。

[0005] 本发明的进一步目的是提供一种基于安卓的画笔宽度控制装置。

[0006] 为解决上述技术问题,本发明的技术方案如下:

[0007] 一种基于安卓的画笔宽度实时控制方法,包括以下步骤:

[0008] S1:实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间;

[0009] S2:计算用户手指或触摸笔的移动速度;

[0010] S3:根据移动速度转换为画笔宽度。

[0011] S4:使用过渡函数使笔迹平滑;

[0012] 上述方法通过实时获取手指移动距离以及移动的时间间隔计算出移动速度,通过数学分析的方法,构建出速度-笔画转换公式,获得实时笔画宽度,通过过渡公式,设置参数,调整前后笔画之间的宽度差距问题,使之更加平滑自然,有效降低计算复杂度与代码量,经过实时计算改变画笔宽度大小,模仿出毛笔粗细不一的笔迹。

[0013] 优选地,步骤S1中实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间包括如下步骤:

[0014] S1.1:初始化当前绘制点坐标 (x_0, y_0) 、前一绘制点坐标 (x, y) ,均初始化为原点 $(0,$

0),判断当前绘制点是否为初始值(0,0);

[0015] S1.2:判断当前绘制点是否为初始值(0,0),即判断是否为初始绘制点;若判断为初始绘制点,执行步骤S1.3;若判断为不是初始绘制点,执行步骤S1.4;

[0016] S1.3:实时获取当前绘制点坐标 (x_0, y_0) ,并将其赋值给前一绘制点坐标 (x, y) ;利用Android的Date对象获取当前时刻 t_0 ,并将其赋值给前一时刻 t ,返回步骤S1.2;

[0017] S1.4:实时获取当前绘制点坐标 (x_0, y_0) ,利用安卓的Date对象获取当前时刻 t_0 ,完成步骤S2、S3、S4后,并将 (x_0, y_0) 赋值给前一绘制点坐标 (x, y) , t_0 赋值给前一时刻 t ,返回步骤S1.2。

[0018] 优选地,步骤S2计算用户手指或触摸笔的移动速度,包括以下步骤:

[0019] S2.1:根据距离公式 $S = \sqrt{(x_0 - x)^2 + (y_0 - y)^2}$ 计算移动距离S;

[0020] S2.2:计算移动距离为S时的间隔时间T:

[0021] $T = t_0 - t$

[0022] S2.3:计算移动速度V:

[0023] $V = S/T$ 。

[0024] 优选地,步骤S3根据移动速度转换为画笔宽度,包括以下步骤:

[0025] S3.1:建立以移动速度V为x轴,以画笔宽度W为y轴的坐标系,令 (V_{\min}, W_{\min}) 表示初态, (V_{\max}, W_{\max}) 表示终态,其中, V_{\min} 为用户手指或触摸笔的最小移动速度,移动速度小于等于 V_{\min} 时,画笔宽度为最小画笔宽度 W_{\min} 并保持不变; V_{\max} 为用户手指或触摸笔的最大移动速度,移动速度大于等于 V_{\max} 时,画笔宽度为最大画笔宽度 W_{\max} 并保持不变;

[0026] S3.2:初始化当前画笔宽度 W_X 、前一画笔宽度 W_B ,并将其初始化为同一的初始值,根据线性插值公式,推导得出移动速度与画笔宽度的公式,所述线性插值公式为:

[0027]
$$y = y_a + (y_b - y_a) \frac{x - x_a}{x_b - x_a}$$

[0028] 该式表示过点 (x_a, y_a) 和点 (x_b, y_b) 的直线,由该式得到:

[0029]
$$W_X = W_{\min} + W_{\max} \cdot \frac{V - V_{\min}}{V_{\max}}。$$

[0030] 通过线性插值建立起速度和画笔宽度的转换模型,实现毛笔笔锋的变化,在速度-时间模型基础上,优化了速度与画笔宽度转化的算法,在不增加代码复杂度的情况以及计算量的前提下更好模拟毛笔粗细不一的笔迹的变化。

[0031] 优选地,步骤S4使用过渡函数使笔迹平滑,包括以下步骤:

[0032] S4.1:判断画笔是否为初次设置,即判断前一画笔宽度是否等于初始值;

[0033] S4.2:若前一画笔宽度等于初始值,直接把画笔宽度设置为当前画笔宽度 W_X ;

[0034] 若前一画笔宽度不等于初始值,利用过渡公式设置画笔宽度,所述过渡公式为:

[0035]
$$W_c = W_X \cdot \frac{1}{3} + W_B \cdot \frac{2}{3}$$

[0036] 式中, W_c 为平滑后的画笔宽度。

[0037] 通过过渡公式的设计,使画笔宽度的变化更加平滑、自然。

[0038] 优选地,步骤S3.1中 $V_{\min} = 0.1$, $V_{\max} = 5$, $W_{\min} = 10$, $W_{\max} = 70$,步骤S3.2中初始值为-

1。

[0039] 一种基于安卓的画笔宽度计算装置,包括:

[0040] 记录模块,用于实现实时捕捉记录用户手指或触摸笔移动距离和间隔时间;

[0041] 计算模块,用于计算用户手指或触摸笔的移动速度;

[0042] 转换模块,根据移动速度转换为画笔宽度;

[0043] 过渡模块,使用过渡函数使笔迹平滑。

[0044] 与现有技术相比,本发明技术方案的有益效果是:

[0045] 本发明提出了一种在安卓平台运行的,基于线性插值计算和canvas类实时计算画笔宽度的方法。通过线性插值建立起速度和画笔宽度的转换模型,实现了毛笔的笔锋的变化。通过使用安卓自带的绘图API可以快速实现所需功能,更好地适配移动端设备,增加代码可读性,同时通过线性插值建立起来的数学模型,在其速度-时间模型基础上,优化了速度与画笔宽度转化的算法,在不增加代码复杂度的情况以及计算量的前提下更好模拟出毛笔粗细不一的笔迹的变化。本发明在较好兼容了移动端的同时,很好实现了书写过程的仿真。

附图说明

[0046] 图1为一种基于安卓的画笔宽度控制方法流程图;

[0047] 图2为实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间的流程图;

[0048] 图3为根据移动速度转换为画笔宽度和使用过渡函数使笔迹平滑的流程图;

[0049] 图4为效果展示图;

[0050] 图5为一种基于安卓的画笔宽度控制装置示意图。

具体实施方式

[0051] 附图仅用于示例性说明,不能理解为对本专利的限制;

[0052] 为了更好说明本实施例,附图某些部件会有省略、放大或缩小,并不代表实际产品的尺寸;

[0053] 对于本领域技术人员来说,附图中某些公知结构及其说明可能省略是可以理解的。

[0054] 下面结合附图和实施例对本发明的技术方案做进一步的说明。

[0055] 实施例1

[0056] 本实施例提供的一种基于安卓的画笔宽度控制方法,如图1,包括以下步骤:

[0057] S1:实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间;

[0058] S2:计算用户手指或触摸笔的移动速度;

[0059] S3:根据移动速度转换为画笔宽度;

[0060] S4:使用过渡函数使笔迹平滑。

[0061] 步骤S1中实时捕捉并记录用户手指或触摸笔的移动距离和间隔时间,如图2,包括如下步骤:

[0062] S1.1:初始化当前绘制点坐标 (x_0, y_0) 、前一绘制点坐标 (x, y) ,均初始化为原点 $(0, 0)$,判断当前绘制点是否为初始值 $(0, 0)$;

[0063] S1.2:判断当前绘制点是否为初始值(0,0),即判断是否为初始绘制点;若判断为初始绘制点,执行步骤S1.3;若判断为不是初始绘制点,执行步骤S1.4;

[0064] S1.3:实时获取当前绘制点坐标 (x_0, y_0) ,并将其赋值给前一绘制点坐标 (x, y) ;利用安卓的Date对象获取当前时刻 t_0 ,并将其赋值给前一时刻 t ,返回步骤S1.2;

[0065] S1.4:实时获取当前绘制点坐标 (x_0, y_0) ,利用Android的Date对象获取当前时刻 t_0 ,完成步骤S2、S3、S4后,并将 (x_0, y_0) 赋值给前一绘制点坐标 (x, y) , t_0 赋值给前一时刻 t ,返回步骤S1.2。

[0066] 步骤S2计算用户手指或触摸笔的移动速度,如图3,包括以下步骤:

[0067] S2.1:根据距离公式 $S = \sqrt{(x_0 - x)^2 + (y_0 - y)^2}$ 计算移动距离 S ;

[0068] S2.2:计算移动距离为 S 时的间隔时间 T :

[0069] $T = t_0 - t$

[0070] S2.3:计算移动速度 V :

[0071] $V = S/T$ 。

[0072] 步骤S3根据移动速度转换为画笔宽度,如图3,包括以下步骤:

[0073] S3.1:建立以移动速度 V 为 x 轴,以画笔宽度 W 为 y 轴的坐标系,令 (V_{\min}, W_{\min}) 表示初态, (V_{\max}, W_{\max}) 表示终态,其中, V_{\min} 为用户手指或触摸笔的最小移动速度,移动速度小于等于 V_{\min} 时,画笔宽度为最小画笔宽度 W_{\min} 并保持不变; V_{\max} 为用户手指或触摸笔的最大移动速度,移动速度大于等于 V_{\max} 时,画笔宽度为最大画笔宽度 W_{\max} 并保持不变;

[0074] S3.2:初始化当前画笔宽度 W_x 、前一画笔宽度 W_B ,并将其初始化为同一的初始值,根据线性插值公式,推导得出移动速度与画笔宽度的公式,所述线性插值公式为:

$$[0075] \quad y = y_a + (y_b - y_a) \frac{x - x_a}{x_b - x_a}$$

[0076] 该式表示过点 (x_a, y_a) 和点 (x_b, y_b) 的直线,令 $V_{\min} = 0.1, V_{\max} = 5, W_{\min} = 10, W_{\max} = 70$,由该式得到:

$$[0077] \quad W_x = 10 + 70 \cdot \frac{V - 0.1}{5}。$$

[0078] 步骤S4使用过渡函数使笔迹平滑,包括以下步骤:

[0079] S4.1:判断画笔是否为初次设置,即判断前一画笔宽度是否等于初始值;

[0080] S4.2:若前一画笔宽度等于初始值,直接把画笔宽度设置为当前画笔宽度 W_x ;

[0081] 若前一画笔宽度不等于初始值,利用过渡公式设置画笔宽度,所述过渡公式为:

$$[0082] \quad W_c = W_x \cdot \frac{1}{3} + W_B \cdot \frac{2}{3}$$

[0083] 式中, W_c 为平滑后的画笔宽度。

[0084] 在具体实施过程中,经本实施例实施后的实施效果图如图4所示,字体显示的效果接近毛笔笔迹,且更为自然平滑。

[0085] 实施例2

[0086] 本实施例提供一种基于安卓的画笔宽度计算装置,如图5,包括:

[0087] 记录模块,用于实现实时捕捉记录用户手指或触摸笔移动距离和间隔时间;

[0088] 计算模块,用于计算用户手指或触摸笔的移动速度;

[0089] 转换模块,根据移动速度转换为画笔宽度;

[0090] 过渡模块,使用过渡函数使笔迹平滑。

[0091] 相同或相似的标号对应相同或相似的部件;

[0092] 附图中描述位置关系的用语仅用于示例性说明,不能理解为对本专利的限制;

[0093] 显然,本发明的上述实施例仅仅是为清楚地说明本发明所作的举例,而并非是对本发明的实施方式的限定。对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式的变化或变动。这里无需也无法对所有的实施方式予以穷举。凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明权利要求的保护范围之内。

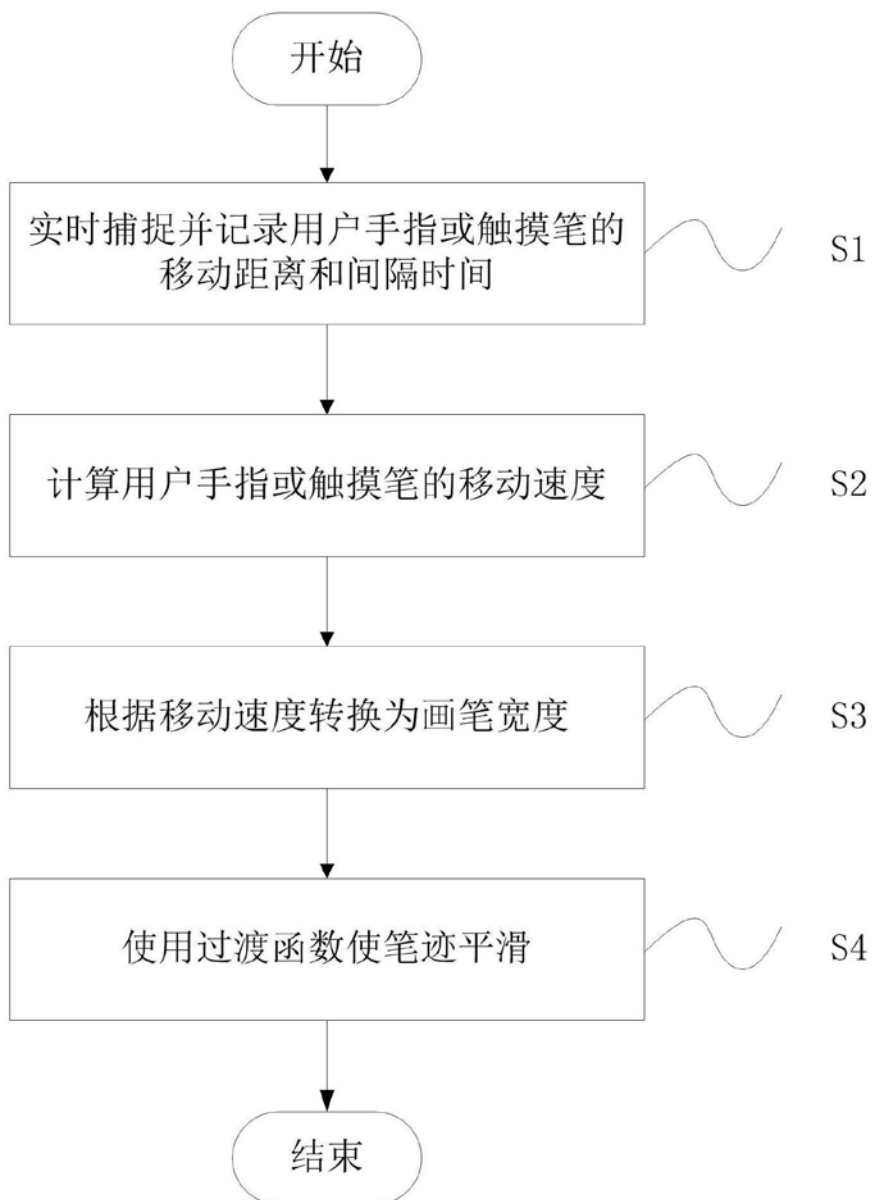


图1

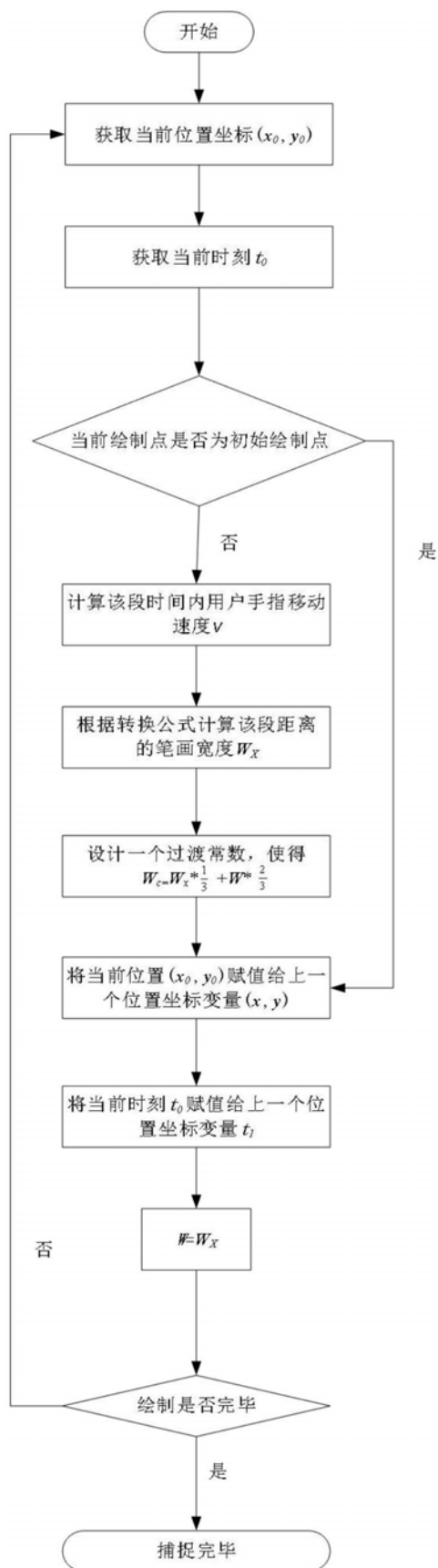


图2

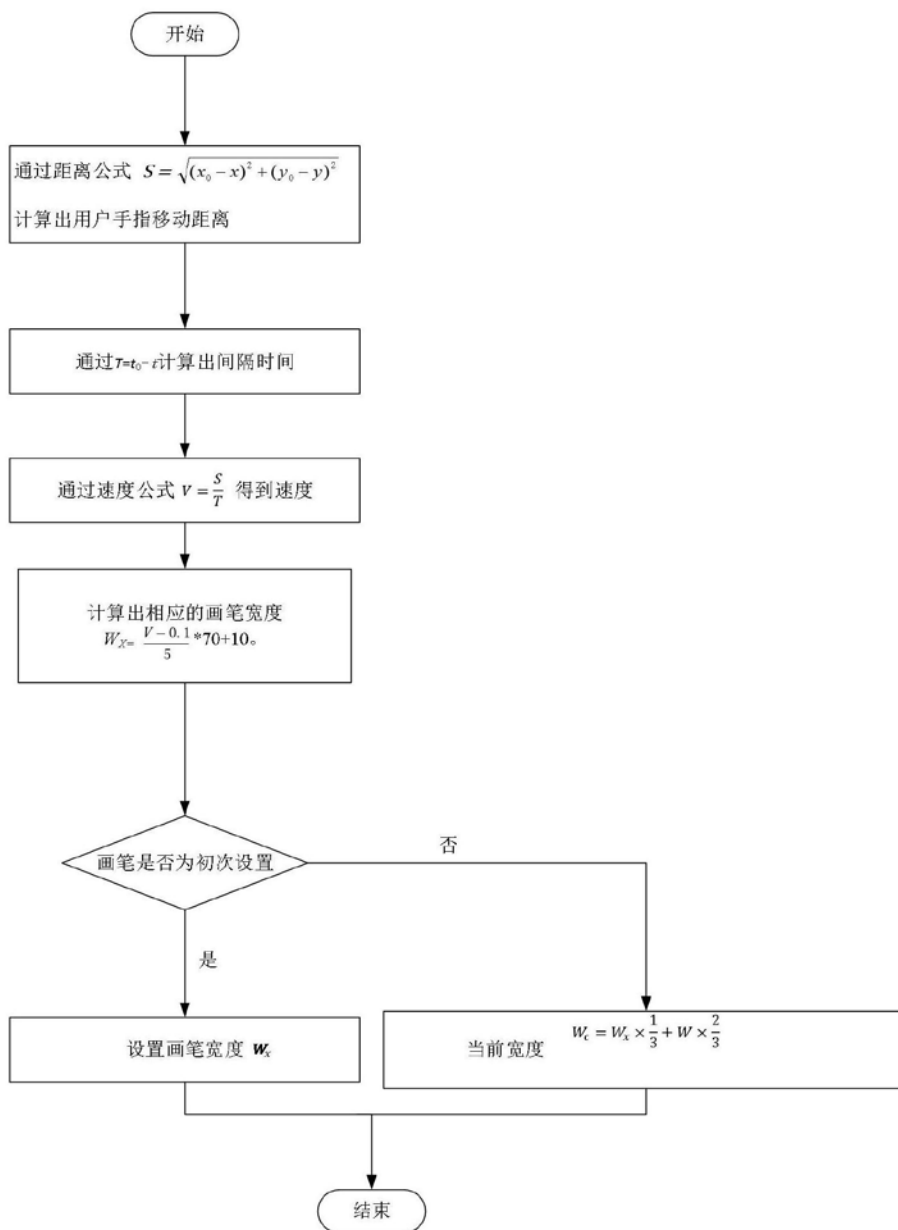


图3

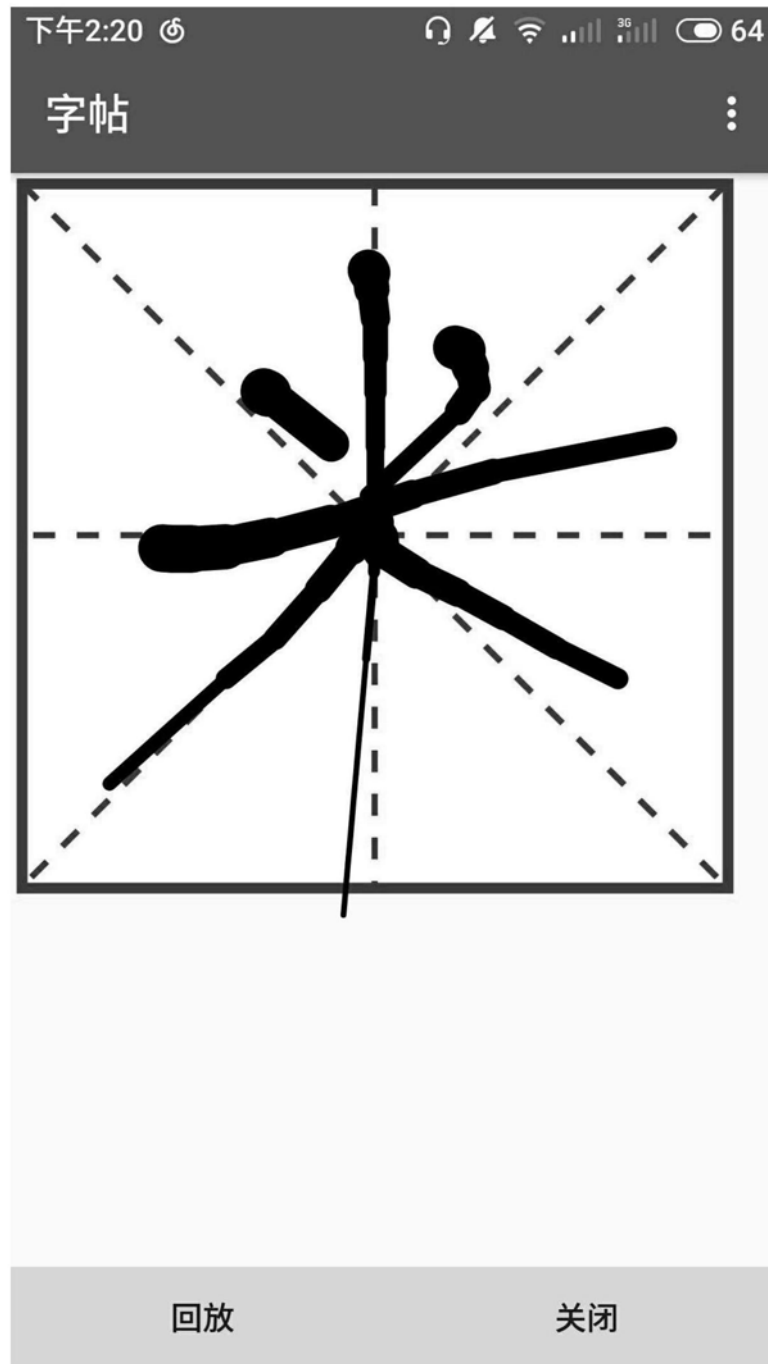


图4

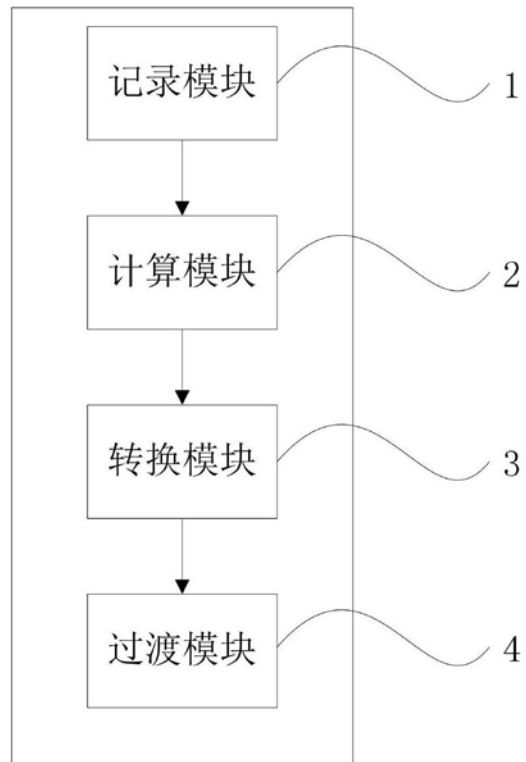


图5