



用 Scratch 玩手势控制(下)

谢作如 浙江省温州中学

毫无疑问,Leap Motion是一款很有趣的体感控制器。借助于斯蒂芬·豪威尔“Scratch 2.0 Plug-in for Leap Motion”插件,中小學生用Scratch2.0,就能编写出很酷的手势控制方面的应用。在上期的文章中,我借助“用手‘抓’住小猫”和“深海捕鱼”两个范例,初步介绍了手势控制方面的编程。在这期的文章中,我们继续研究稍微复杂的手势识别功能的实现。

需要提示的是,Scratch2.0的文件能够保存其原先使用的第三方插件信息。你只要打开原来保存的手势控制程序,就不需要再次去载入“LeapMotion.json”脚本文件了。这样一来,我们编写需要第三方插件的Scratch程序就比较方便了。

● 用手势实现放大和缩小功能

通过手势控制图片或者网页的放大和缩小,是手机触控屏上的常见应用。这一功能虽然常见,但在普通的触控屏上实现起来并不容易,因为这涉及多点触控。Leap Motion可以识别双手,刚好可以让我们体验这一好玩的功能。

以下是一个通过手势控制小鱼大小的程序,具体功能为:双手握拳,然后改变双手之间的距离,屏幕上的小鱼也就根据距离的变化,自行放大或者缩小,



图1

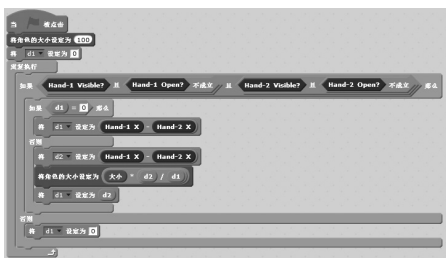


图2

其界面如图1所示。

程序功能分析:

①首先我们需要判断双手是否握拳,用“Hand-1 Open?”和“Hand-2 Open?”不成立就可以判断。原理很好理解,但代码拼接起来有点长,这也是图形化编程语言的最大不足。

②而后要判断双手之间有没有改变距离。我使用d1和d2两个变量,d1记录的是上一次的距离,d2记录的是当前的距离。然后将d2除以d1的值与角色的原大小相乘,并设定角色大小。这样就能够用动态的双手距离控制角色大小了。

需要注意的是,如果双手不处于“握拳”状态时,d1的值需要初始化为0。不然,这会导致你无法连续做缩放角色的操作。完整的代码如图2所示。

● 和电脑玩“石头剪刀布”

“石头剪刀布”是大家最熟悉的猜拳游戏了,和电脑玩“石头剪刀布”游戏的程序也多如牛毛。但是,那些程序无非是用鼠标键盘选择一个按钮,然后和电脑出的随机数进行比较,无趣得很。你甚至不知道电脑是不是耍了什么手段,毕竟电脑永远是在你选择后才暗中“出拳”。你见过有谁真正用划拳的形式和电脑玩“石头剪刀布”吗?

我们可以借助Leap Motion,做一个玩家和电脑可以“公平”玩猜拳的手势游戏。游戏设定为:当倒计时结束,玩家和电脑同时“出拳”,然后电脑识别出玩家的手势,根据自己的出拳结果,判断输赢。

游戏的核心功能并不复杂。首先要让电脑能够判断玩家出的拳(手势)是石头还是剪刀或者布。这三个手势中,比较麻烦的是“剪刀”,这需要进行多次测试,看Leap Motion的识别结果是否准确。比如我就用下页图3的代码,分别判断当玩家出“剪刀”的时候,五个手指的状态分别是处于什么情况,记录并分析。



图3

也许是插件的BUG或者是Leap Motion的识别存在某些不足,“剪刀”这一手势,插件反馈的往往不是“食指和中指”可见,反而是“大拇指和食指”可见。经过多次测试,我发现如下表所示的规律:

手势	特点	识别代码
石头	握紧拳头,没有可见的手指。	Hand-1 Open? Finger-1 Hand-1 Visible? Finger-2 Hand-1 Visible? Finger-3 Hand-1 Visible? Finger-4 Hand-1 Visible?
剪刀	握紧拳头,只有两个指头可见	Hand-1 Open? Finger-1 Hand-1 Visible? Finger-2 Hand-1 Visible? Finger-3 Hand-1 Visible? Finger-4 Hand-1 Visible?
布	手掌处于张开状态	Hand-1 Open? Hand-2 Open?

可能有人会问,对于“石头”的判断,为什么不直接用“hand-1 Open’不成立”呢?其实,对于Leap Motion来说,“剪刀”也是“hand-1 Open’不成立”的。既然我们仅仅是为了区分三种手势,也可以更简单点,将三种情况合并起来判断,如图4。

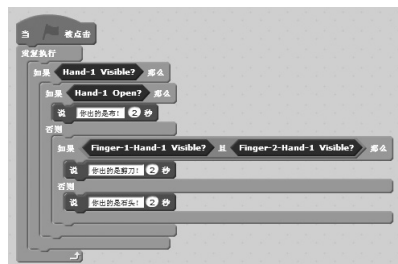


图4

需要注意的是,玩家出拳前最好要把手势做标准,然后展示在Leap Motion的上方,不要太靠近。尤其不能等手到了固定的位置再做出手势,这样容易导致Leap Motion误判。在实际的测试中,Leap Motion还是能让我满意

的,识别的准确率能达到90%。

好了,既然三种手势都能正确识别,这个程序的核心功能就实现了。我们可以优化一下程序,把其他功能完善起来。限于篇幅,具体的代码就不再赘述了。具体界面如图5—图9所示。



图5



图6



图7

看到“后出”提示了吗?在这个游戏中,我们再也不用担心电脑作弊了,因为它甚至会比你先出。当然,如果你不及时出拳,程序会判定你“后出”的。总之,谁

也别想要赖。



图8



图9

借助这两个范例,我们初步实现了一些有趣的手势识别功能,但是也发现“Scratch 2.0 Plug-in for Leap Motion”插件存在很多不足。如果你具有一定的编程能力,还可以下载插件源码,增加相应的功能,让Scratch2.0的手势控制更加强大。我们希望有更多的高手,在他人的开源代码基础上,增加新的功能后再开源出去。如果你用Leap Motion识别出更多的手势,请别忘了分享你的代码。e

如果对相关内容感兴趣,请关注主持人博客。

