

基于 Leap Motion 的空中输入法项目报告

黄冰鉴 李玮祺 李志鹏 张泽阳

2019 年 1 月 20 日

1 项目设计

人类向计算机输入信息的方式一直都是通过物理形式的键盘，然而随着计算机的发展与进步，这一方式越来越不自然，其效率也逐渐成为问题。一方面体现在计算机越来越小型化，例如智能手机，由于 Fat Finger 等问题，在智能手机上进行传统 26 键键盘的输入是较为困难的。另一方面则是当社会进入 AR/VR 时代时，继续使用传统的键盘输入方式无疑会极大地破坏沉浸感，而且其输入效率也逐渐在与语音输入或其他方式的竞争中处于下风。最重要的问题在于，使用键盘输入可能对于当前社会的大部分人都觉得十分稀松平常，可以进行相当速度的信息输入，然而这都是建立在我们已经习惯了键盘输入的基础之上的。不妨回想一下自己在刚刚接触计算机和键盘时的场景，任何一个人想要通过键盘进行快速输入都需要大量的练习和持续的使用。这样巨大的学习成本以及上述其他问题使我们考虑利用现有或未来可以预见的技术，设计一种新型的输入方式。

我们设计的输入法学习成本低，输入快速准确，并且十分自然的融入到人机交互的过程中，适应 AR/VR 的发展潮流。在我们的项目中，基于现有的手势传感器 Leap Motion，设计出一套全新的手势控制输入法。具体设计为利用人类的双手进行 26 个英文字母的选择。由于 26 个字母较多，单手不能准确的快速选择，因此我们希望左手通过水平方向的移动来选择希望输入的字母的大致区间，然后通过右手的水平移动来精确选择这个区间内的字母。具体界面如下图

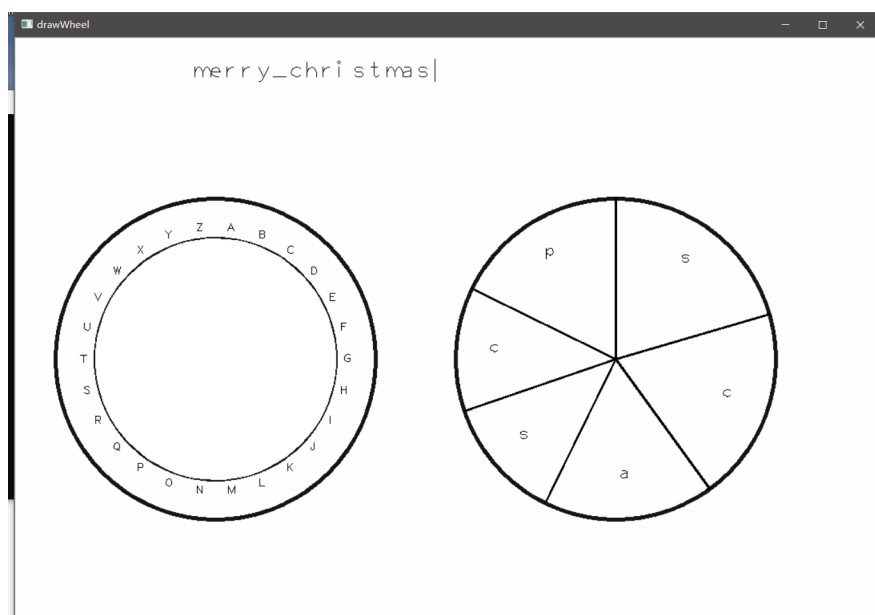


图 1: 实现效果

为了加快输入速度，我们加入了对单个字母和单词的自动联想。当用户输入了一个字母后，右手的选择范围会出现最可能出现的字母或者单词，用户可以直接通过右手的选择进行输入，不必再次移动左手。这样可以大幅加快输入速度。此外，我们还设计了三种人性化的特殊手势，用于文本输入中的常见操作。第一个是进行换行操作，我们设计的手势为双手翻掌并向上抬起，好似把之前的文本向上抬走，空出新的空间进行新的输入。第二个是删除操作，设计手势为单手握拳，直观理解为抓住当前字母并移出当前文本。第三个是空格，英文输入中经常要用到空格来对单词做分割，因此我们设计的对应手势即为单手向下拍，就像在键盘上敲下了空格键一样。

2 项目实施

我们把项目分成了四个部分：Leap Motion 的 API 数据获取以及动作识别；输入预测以及候选词选取；图形化 UI 用户界面。接下来我们将分别进行介绍。

2.1 Leap Motion API 数据获取

Leap Motion 的 API 乍一看极其复杂，既要配环境，又要查看复杂的使用文档，但实际上上手时却发现特别简单。在配好环境之后，Leap Motion 库中自带的样例程序就可以编译成功了。实际使用时，我们发现其中一个名为 `callbacksample` 的项目和我们的需求非常相似。仔细查看之后，我们发现这个样例采用的是回调的机制，即每当设备采集到一组新的

数据时，它会自动激活代码中的 `onFrame()` 函数，并且频率周期非常稳定（10ms/100Hz）。因此我们决定基于这个程序，完成动作识别。

2.2 动作识别

具体而言，每次设备调用 `onFrame()` 函数时，我们就将新的左右手数据存入一个长度为 300 的循环数组（相当于时间跨度为 3s 的数据），并将旧的数据删去。通过对时间数据的分析，我们实现了以下识别任务：

- **输入中心的动态确认：**我们采集了手掌中心的坐标，如果连续 500ms 手掌中心坐标的变化非常小的话，就将手掌坐标更新为这个点。
- **左右手的圆盘输入：**根据我们的设计，在左右手附近应该又一个虚拟的圆环作为输入。因此，如果在采集的数据中发现手的位置偏离圆心达到一定程度，我们就视为一次输入。具体的输入和角度有关，因此我们还将得到的笛卡尔坐标系下的距离向量转化为了角度，方便下一步处理。
- **空格的输入：**可以看到，在第二个识别任务中我们只使用了水平面上的坐标变化，因为我们将竖直方向上的变化用于了空格的输入。当用户用力向下拍时，他在竖直方向上会和中心产生较大偏差，我们将这个偏差视为空格的输入。
- **换行的输入：**换行同样设计为一个竖直方向上的动作，不过额外使用了手掌朝向的数据。当用户将手掌反过来手心向上，并向上抬时，我们视为换行操作，正好是和换行相反的一个动作。
- **删除的操作：**删除操作不涉及到手掌中心的变化，当用户的手掌状态从平展变为握拳时，视为一次删除单个字符的操作。

需要特别注意的是，在动作识别的实现中，我们使用的基本上是非常基础的手部数据，例如手掌中心，手掌朝向，手掌是否握拳等等。在项目之初，我们设计的手势非常的复杂，以删除字符为例，需要先握拳再移出圆环。在实际实现的过程中，我们发现，当手移出一定距离并且处于移动状态时，Leap Motion 对握拳的识别非常不稳定，导致握拳操作经常检测失败。因此后来我们将删除操作改为了静态操作。另外，因为 Leap Motion 对细节（例如关节位置，弯曲程度）的检测误差比较大，所以我们完全没有使用这些数据。

2.3 词库

词库在以下三种情况提供候选词选取的功能，即在已知当前词前缀的情况下：

1. 得到概率最高的几个补全单词
2. 得到概率最高的几个期望字母
3. 在上述两种情况缺省下，给出词频最高的字母

三种情况的候选数量为使用参数，在实验中选取合适的量，按照优先级依次补齐，最终总数量为 6 个。在总框架中，使用栈维护输入文本和当前词前缀，调用词库获取候选词和对应概率。在实际实现中，采用 nlohmann json 保存预处理数据以及查表。

2.4 UI 设计

UI 的设计主要按照实验设计部分描述的进行设计。其目的是给用户视觉上的信息反馈，形成人机交互的完整回路，进行有效的信息交互。具体实现为，左手和右手分别绘制一个选择轮盘。左手轮盘上显示出所有 26 个字母，并且在用户进行左手选择后给出反馈，显示出选择的区域。右手轮盘上则显示出左手选择的字母区间或者联想出来的字母或单词，方便用户进行连续的输入。此外还在轮盘上方显示出已输入的文本，使用户可以进行检查和验证。实现效果在实验设计部分以效果图形式给出。

3 用户实验

在完成项目后，为了验证和说明项目的特点，我们设计并进行了用户实验。

3.1 实验目的

我们的项目设计目标主要为学习成本低并且可以快速准确输入，因此我们的实验目的就是验证我们实现的输入法的确学习成本低，并且可以实现快速准确输入。

3.2 实验设计

为了达到实验目的，我们的实验设计为在被试学会基本操作后，进行五次两分钟的输入，研究被试在实验过程中的学习曲线和最终达到输入速度。为了控制其他变量，我们提供了十几句简单的英文句子，长度在 8-12 个词，要求被试在实验前记忆要输入的内容，避免在实验时间内因为需要回忆文本造成时间浪费。

被试一共有八人，均为 20-21 岁大学生，5 男 3 女，均没有接触过类似的输入方式或手势控制。在实验过程中，要求被试阅读实验须知并学会基本操作，进行两分钟计时的输

入，统计有效输入数量，重复五次，得到学习曲线。每个 Trial 之间有一到两分钟的休息，在此期间对被试进行采访和交流。

采访问题如下

- 您觉得我们的输入法有什么好用/不好用的地方
- 您在使用回车等特殊手势是否感到舒适或流畅
- 您觉得对您输入时最大的阻碍是什么

3.3 实验结果

在实验结束后，统计被试的输入，得到整体的学习曲线和十分钟练习后达到的输入速度。横坐标为 trial 的编号，纵坐标为每个 trial 内的有效输入单词数量。

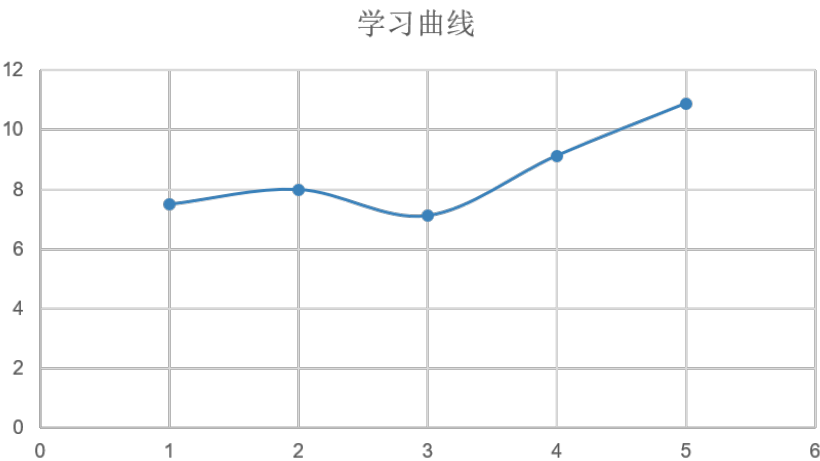


图 2: 学习曲线

可以看到被试在短暂快速的学习后就达到了较好的输入速度，且没有感受到疲劳。学习曲线符合学习的一般规律，中间的下降主要原因是被试在不太熟练的情况下进行加速反而造成速度变慢。从学习曲线中可以看到在经历了十分钟的学习后达到了接近 6wpm 的输入速度，并且个别被试达到了接近 10wpm 的速度。

此外在采访中我们收集到了一些被试的反馈和建议，并且根据这些建议进行了一些修改，比如对删除手势的修改。

- 不能快速直接选择字母所在区域
- 删除操作不够便捷

- 需要花费大量时间在寻找字母上，尤其是左轮
- 将水平面的操作改为纵平面

从实验结果上看，我们的项目达到了我们的设计目标，普通人可以在较短时间内较为熟练的掌握输入方式，并且进行一定速度的输入。通过实验我们相信在更多的训练和学习后，可以达到接近键盘输入的速度，但是所需的学习时间一定比键盘的学习时间短，同时我们的输入法还在 AR/VR 中拥有更好的应用前景。

4 项目总结

在这次人机交互课大作业中，我们完成了一个与其它组截然不同的空中输入法。在基本保证了输入速度的前提下，我们将输入法的受众从“会盲打”的人群拓展到了“认识英文字母”的人群，实现了“量”的飞跃。

通过这个项目 and 这个学期的学习，我们对于人机交互有了更深刻的认识，同时也在该项目中加入并实现了一些我们自己的想法，也有一些自己对于人机交互的思考。我们站在一个希望使人机交互更加直观和自然的思考角度，希望在拥有更多计算机的社会中，能使计算机和人类和谐共处，能使计算机更好的为人类提供更舒适更自然的服务。我们希望计算机可以适应人类，而不是人类要改变自己的习惯去适应机器，因为人机交互中，最本质最重要的要素还是人。

这是我们组的四位同学第一次完整地完成了项目研究。从最初的输入法设计，到之后的模块划分、模块实现以及整合，再到最后的用户实验和展示验收，我们每个人都收获满满。在这个过程中，我们有关于输入法设计的激烈讨论，也有因为设备精度而不得不做的妥协；有整合项目时寻找未知 bug 的痛苦，也有用户实验时看着自己系统被人使用的快乐。最重要的是，我们学以致用，将课堂上学到的理论知识以及小实验中学到的经验应用到了这次大作业中，顺利地完成了这个项目。

在此特别感谢史元春老师，易鑫老师以及古裔正助教在课堂学习和大作业完成过程中给予我们的帮助！