摘 要

LeapMotion体感控制器是一款通过手势来控制软件或硬件的一种设备，目前

**ABSTRACT**

目 录

1. 研究背景
2. LeapMotion基本原理
3. LeapMotion手势控制的建立
4. LeapMotion与Unity3d的结合实例
5. LeapMotion手势控制的优点与局限性
6. 总结与展望

第一章 研究背景

1.1 虚拟现实

虚拟现实（VR）是一种计算机生成的场景，通过感官和感知来模拟体验。身临其境的环境可以与现实世界类似，也可以是幻想的，创造出在普通物理实验中不可能的体验。增强现实系统也可被视为VR一种形式，其通过将实时摄像头馈送到耳机或者通过智能手机或者平板设备的虚拟信息分层，是用户能够观看三维图像。

目前的虚拟现实技术通常使用虚拟现实耳机或多投影环境，有时结合物理环境或道具来生成模拟用户在虚拟或虚拟环境中的物理存在的逼真图像，声音或者其他感觉。使用虚拟现实设备的人能够观察人造世界，在其中一栋并且与虚拟特征或物品进行交互行为。这种效果通常由头戴式VR设备产生，头戴式显示器在眼镜前方有一个小屏幕，但也可以通过特别设计的带有多个大屏幕的房间创建。

包括通过游戏控制器或其他设备向用户传输振动和其他感觉的VR系统被称为触觉系统。这种触觉信息通常被称为医疗，视频游戏和军事训练应用中的力反馈。

1.2 LeapMotion体感控制器

LeapMotion控制器是一块小巧的USB外围设备，设计用于面朝上的方式放置在桌面上。它也可以安装到虚拟现实的头戴式设备上。使用两个单色红外摄像机和三个红外LED，该设备可以观察一个大约距设备距离大约1米的半球区域。LED产生图案的IR光，相机每秒产生几乎200帧的反射数据。然后通过USB电缆将其发送到主机，由LeapMotion软件通过公司“复杂数学”用以公司尚未公开的方式对其进行分析，以某种方式通过比较两台相机产生的2D数据帧来合成3D位置数据。在2013年的一项研究中，控制器的整体平均精确度显示为0.7毫米。

LeapMotion最初将数千个单元分发给有兴趣为该设备创建应用程序的开发人员。LeapMotion控制器与2013年7月首次发货。2016年2月，LeapMotion发布了其核心测试软件的主要测试版本更新。这款名为Orion的软件专为虚拟现实中的手部追踪而设计。

1.3 Unity

Unity是一款多用途游戏引擎，支持2D和3D图形，拖放功能以及使用C#编写脚本。支持另外两种编程语言：Boo，在Unity5发布时启用，而JavaScript在2017年8月发布的Unity后开始弃用。

该引擎针对以下图形API：Windows和Xbox One上的Direct3d；Linux, macOS和Windows上的OpenGL；Android和iOS上的OpenGL ES; Web上的WebGL；以及视频游戏控制台中的转悠API。此外，Unity支持iOS和macOS上的地基API METAL, Android, Linux和Windows上的Vulkan以及Windows和Xbox One上的Direct3D 12.

在2D游戏中，Unity允许引入sprites和先进的2D游戏渲染器。对于3D游戏，Unity允许为游戏引擎支持的没饿平台指定纹理压缩， mipmap和分辨率设置，并为凹凸贴图，反射映射，视差映射，屏幕空间环境遮挡（SSAO）， 动态阴影贴图，渲染到纹理和全屏幕后处理效果的阴影提供支持。Unity还为开发者体统服务，包括：Unity广告，Unity分析，Unity认证，Unity云构建，Unity Everyplay，Unity IAP，Unity多人游戏，Unity性能报告和Unity写作。

Unity支持使用Cg(Microsoft的高级着色语言修改版本)创建自定义顶点，片段（或像素），向前计算着色器和Unity自己的表面着色器。

第二章 LeapMotion基本原理

2.1 LeapMotion的传感器结构

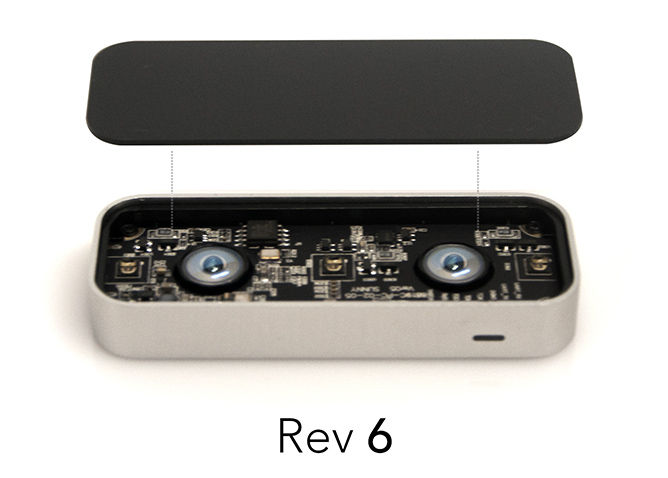
从API的角度来看，LeapMotion传感器的结构如下：

图2-1-1

大体上，Leap传感器根据内置的两个摄像头从不同角度捕捉的画面，重建出手掌在是世界三维空间的运动信息。检测的范围大体在传感器上方25毫米到600毫米之间，检测的空间大体是一个倒四棱锥体。

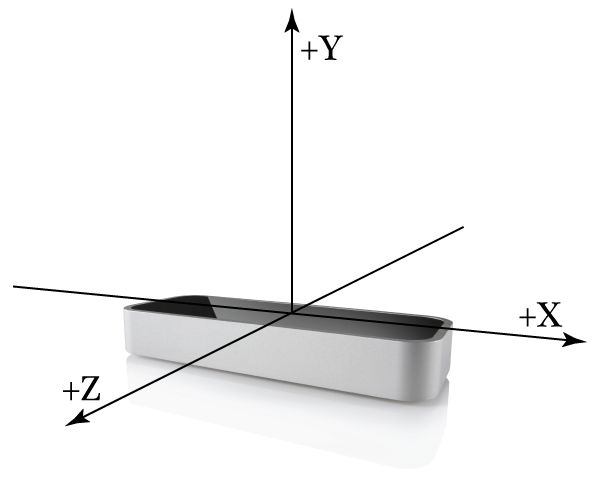
首先，Leap Motion传感器会建立一个直角坐标系坐标的原点是传感器的中心，坐标的X轴平行于传感器，指向屏幕右方，Y轴指向上方，Z轴指向背离屏幕的方向，单位为真实世界的毫米。如图所示：

图2-1-2

在使用过程中，Leap Motion传感器会定期的发送关于手的运动信息，每份这样的信息成为帧（frame）.每一个这样的帧包含检测到的：

·所有手掌的列表及信息；

·所有手指的列表及信息；

点手持工具（细的，笔直的，比手指场的东西，例如一支笔）的列表及信息

·所有可指向对象（Pointable Object），即所有手指和工具的列表及信息；

2.2 Leap Motion的数据收集

Leap传感器会给所有这些分配一个唯一标识（ID），在手掌、手指、工具保持在视野范围内时是不会改变的。根据这些ID，可以通过Frame::hand()，Frame::finger()，等函数来查询每个运动对象的信息。

Leap可以根据每帧和前帧检测到的数据，生成运动信息。例如，若检测到两只手，并且两只手都朝一个方向移动，就认为是平移，若像握着球一样转动，则记为旋转。若两只手靠近或分开，则记为缩放。所产生的数据包含：

·旋转的轴向向量

·旋转的角度（顺时针为正）

·描述旋转的矩阵

·缩放因子

·平移向量

对于每只手，可以检测到如下信息：

·手掌中心的位置（三维向量，相对于传感器坐标原点，毫米为单位）

·手掌移动的速度（毫米每秒）

·手掌的法向量（垂直于手掌平面，从手心向外）

·根据手掌弯曲的弧度确定虚拟球体的中心；

·根据手掌弯曲的弧度确定的虚拟球体的半径

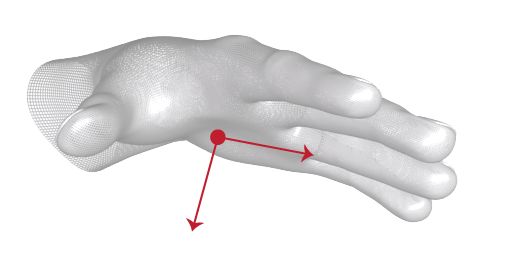
其中，手掌的法向量和方向如下图所示：

图2-2-1

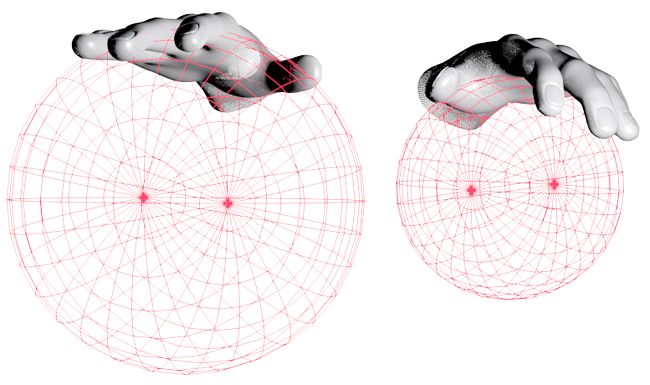
[手掌球]的圆心和半径

图2-2-2

对于每个手掌，可以检测出平移，旋转（如转动手腕带动手掌转动）、缩放（如手指分开聚合）的信息。检测的数据如全局变换一样，包括：

·旋转的轴向向量

·旋转的角度

·描述旋转的矩阵

·缩放因子

·平移向量

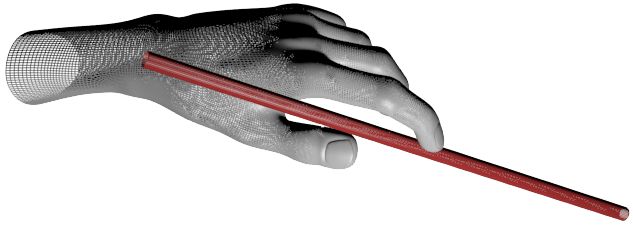
Leap除了可以检测手指外，也可以检测手持的工具。像上文所说的，就是细的，笔直的，比手指长的物件

图2-2-3

对于手指和工具，会统一地成为可指向对象（Pointable Object），每个Pointable Object包含了这些信息：

·长度

·宽度

·方向

·指尖位置

·指尖速度

方向和指尖位置如下图

图2-2-4

根据全局的信息、运动变换，手掌、手指和工具的信息和变换，开发者就可以靠这些来制作游戏和程序了。