1. 环境配置
   1. 安装Orion版

下载链接

<https://developer.leapmotion.com/get-started>

核心功能基于v2版SDK开发(需引用LeapCSharp.NET4.0.dll)

下载链接

<https://developer.leapmotion.com/sdk/v2>

* 1. 需要Unity5.X环境支持
  2. 部分Leap Motion v2 SDK中无Matrix.cs文件，已附于代码中，可依据自己的实际情况增减

1. 方法
   1. 主窗口(Form.cs, Program.cs)
   2. 监听线程与主窗口同步(FormMethod.cs)
   3. 功能实现(Compare\_Normalized.cs)\*
   4. 从接口提需要数据(Hand\_Data.cs)\*
2. 详细说明
   1. 主窗口
      1. Program.cs
         1. Program 类

说明：主线程

* + - * 1. 方法

Main()

功能：运行主窗口

输入：无

输出：无

返回：无

* + - * 1. 属性

FormHandCompare fhc

说明：后台线程与前台UI交互用的实例

* + 1. Form.cs
       1. MainForm 类

说明：主窗口，作为UI，显示Leap Motion识别到的手的3维模型和计算结果，用户能通过街面上的按钮启停后台执行的计算结果的线程

* + - * 1. 方法

MainForm()

功能：主窗口构造程序，

输入：无

输出：无

返回：无

button1\_Click(object sender, EventArgs e)

功能：按钮点击事件，控制监听线程启停，首次启动时向容器装入Unity程序

输入：事件对象，事件参数

输出：无

返回：无

MainForm\_Load(object sender, EventArgs e)

功能：主界面载入时，置ListView的列标

输入：事件对象，事件参数

输出：无

返回：无

MainForm\_FormClosed(object sender, FormClosedEventArgs e)

功能：关闭主界面事件发生后，结束全部线程，关闭Unity程序

输入：对象实例，事件

输出：无

返回：无

* + - * 1. 属性

bool Listening\_Flag

说明：监听线程启停标志

Process p

说明：Unity 进程

bool pflag

说明：Unity 进程存活标志

* 1. 监听线程与主窗口同步
     1. FormMethod.cs
        1. FormHandCompare 类

说明：Leapmotion监听线程与主窗口的控件进行数据交互，UI与后台线程的中间层，当后台线程处理完LeapMotion发来的帧数据后，根据标记位判断是否需要更新UI以及更新哪些控件(异步)

* + - * 1. 方法

FormHandCompare()

功能：类默认构造函数，实例化属性ctrl，ll，Answer\_Fingers[]

输入：无

输出：无

返回：无

StartListen()

功能：加入OnFrame委托，装入并启动(覆盖)监听线程

输入：无

输出：无

返回：无

StopListen()

功能：撤出OnFrame委托

输入：无

输出：无

返回：无

SetForm(string FN,string CCom, string CEcu,string CDot,string CAn)

功能：载入窗体中需要修改的控件Name

输入：窗体名，一般距离结果控件名，欧几里得距离比较控件名，点积比较控件名，提示信息控件名

输出：无

返回：无

ListeningThread()

功能：监听线程执行操作，将ll中的属性装入写到窗体的缓存

输入：无

输出：无

返回：无

WriteToForm()

功能：通过委托，异步的向窗体写数据

输入：无

输出：无

返回：无

Change\_Text\_Method(TextBox tb, string text)

功能：向窗体TextBox写数据（用于多线程3.2.1.1.2.2 说明：中）

输入：TextBox 实例，写入数据

输出：无

返回：无

Change\_Text\_Method(Label tb, string text)

功能：向窗体Label写数据（用于多线程中）

输入：Label实例，写入数据

输出：无

返回：无

Change\_Text\_Method(ListView tb, string[] text)

功能：向窗体ListView写数据（用于多线程中）

输入：Label实例，写入数据数组

输出：无

返回：无

* + - * 1. 属性

Controller ctrl

说明：Leap.Controller 用于连接Leap Motion

LeapListener ll

说明：获取并处理Leap Motion数据

Thread ThreadLeapListener

说明：后台运行的监听线程

string Answer

说明：窗体提示栏缓存

int Answer\_Euclid

说明：窗体欧几里得距离和差缓存

int Answer\_Compare

说明：窗体一般距离和差缓存

int Answer\_Dot\_Product

说明：窗体点积和差缓存

string[] Answer\_Fingers

说明：每只手指一般距离和差缓存

bool changeflag

说明：更新帧标记

bool runflag

说明：后台线程运行标记

delegate void Chang\_Text(TextBox tb, string text)

说明：TextBox写入委托

delegate void Chang\_Lab(Label tb, string text)

说明：Label写入委托

delegate void Chang\_List(ListView tb, string[] text)

说明：ListView写入委托

string FormName

说明：窗体名

string Ctrl\_Compare

说明：一般距离结果控件名

string Ctrl\_Euclid

说明：欧几里得距离比较控件名

string Ctrl\_DotMult

说明：点积比较控件名

string Ctrl\_Answer

说明：提示信息控件名

* + - 1. LeapListener 类

说明：获取并处理Leap Motion传来的帧信息，首先判断Leap Motion手池(Hands Pool)中有几只手，当且仅当有2之手时，对Leap Motion的自然手1进行位置和姿态的归一，若手2与手1为同侧手则只需进行位置，姿态，尺寸的归一，否则需要先做镜面反转。完成归一化后，进行区别度计算，保存结果待FormHandCompare调用。

* + - * 1. 方法

LeapListener()

功能：默认构造函数，全部属性赋初值

输入：无

输出：无

返回：无

OnFrame(object sender, FrameEventArgs args)

功能：对从Leap Motion中获取的帧进行操作

输入：事件对象，事件参数

输出：无

返回：无

Compare\_Method()

功能：执行3种差值计算，

输入：无

输出：无

返回：无

clear()

功能：清空结果

输入：无

输出：无

返回：无

Fin(int id)

功能：查询手指单词

输入：手指编号(0大拇指，1食指，2中指，3无名指，4小拇指)

输出：无

返回：对应英语单词

* + - * 1. 属性

HandKeyData HKD0

说明：手1

HandKeyData HKD1

说明：手2

string answer

说明：提示信息

int Answer\_Euclid

说明：欧几里得距离和差

int Answer\_Compare

说明：一般距离和差

int Answer\_Dot\_Product

说明：点积和差

string[] Answer\_Fingers

说明：每组手指指尖距离差

bool changeflag

说明：更新标记

* 1. 功能实现
     1. Compare\_Normalized.cs
        1. Compare 类

说明：存放比较算法的类

* + - * 1. 方法

Bone\_Compare(BoneKeyData bkd0, BoneKeyData bkd1)

功能：计算一对骨之间两端的距离之和

输入：骨1，骨2

输出：无

返回：距离和(Double)

Finger\_Compare(FingerKeyData fkd0, FingerKeyData fkd1)

功能：计算一对指之间的距离之和，所有对骨的距离之和（调用Bone\_Compare）

输入：指1，指2

输出：无

返回：距离和(Double)

Hand\_Compare(HandKeyData hkd0, HandKeyData hkd1)

功能：计算一对手之间的距离之和，所有对指的距离之和（调用Finger\_Compare）

输入：手1，手2

输出：无

返回：距离和(Double)

Hand\_Euclid(HandKeyData hkd0, HandKeyData hkd1)

功能：计算一对手之间的欧几里得距离和差(每一对指尖-掌心距离差绝对值之和)

输入：手1，手2

输出：无

返回：距离和差(Double)

Hand\_Dot\_Product(HandKeyData hkd0, HandKeyData hkd1)

功能：计算一对手之间的点积和差(每一对掌心-指尖向量中较大的自积后减两者点积)

输入：手1，手2

输出：无

返回：距离和差(Double)

* + - 1. Normalized 类

说明：对手进行大小归一化，归一参数取中指长度比

* + - 1. 函数
         1. Hand\_Normalized(ref HandKeyData hkd0, ref HandKeyData hkd1)

功能：把手2按手1尺寸归一化

输入：手1，手2

输出：手1，归一后手2

返回：无

* + - * 1. Hand\_Normalized(ref HandKeyData hkd, float changing\_parameters)

功能：把手按归一参数归一化

输入：手，归一参数

输出：归一后指

返回：无

* + - * 1. Finger\_Normalized(ref FingerKeyData fkd, float changing\_parameters)

功能：把指按归一参数归一化

输入：指，归一参数

输出：归一后指

返回：无

* + - * 1. Bone\_Normalized(ref BoneKeyData bkd, float changing\_parameters)

功能：把骨按归一参数归一化

输入：骨，归一参数

输出：归一后骨

返回：无

* 1. 从接口提需要数据
     1. Hand\_Data.cs
        1. HandKeyData 类

说明：自定的手数据类型，存放手的基本信息和坐标变换方法，其中通过计算归一翻转矩阵实现手位置和姿态的归一化—— y-掌心法线方向，z-手指向方向，x-前2者所在平面的法线，掌心移动到坐标系原点

* + - * 1. 方法

HandKeyData()

功能：实例化

输入：无

输出：无

返回：无

Load(Hand hand)

功能：将Leap.Hand内的信息复制到HandKeyData

输入：手(Leap.Hand)

输出：无

返回：无

MirrorXRI()

功能：求手的X镜面归一翻转矩阵

输入：无

输出：归一矩阵

返回：无

Transform(Matrix M)

功能：根据翻转矩阵翻转手

输入：翻转矩阵

输出：无

返回：无

GetRIMatrix()

功能：求手的归一翻转矩阵

输入：无

输出：翻转矩阵

返回：无

* + - * 1. 属性

FingerKeyData[] Finger\_

说明：存放手指信息

int Finger\_Count

说明：手指数量

Vector PalmPosition

说明：掌心位置

Vector normal

说明：掌心法线

Vector direction

说明：指向

* + - 1. FingerKeyData 类

说明：存放手指基本信息和坐标归一化方法

* + - * 1. 方法

FingerKeyData()

功能：实例化

输入：无

输出：无

返回：无

Load(Finger finger)

功能：将Leap.Finger内的信息复制到FingerKeyData

输入：指(Leap.Finger)

输出：无

返回：无

Transform(Matrix M)

功能：根据翻转矩阵翻转指

输入：翻转矩阵

输出：无

返回：无

* + - * 1. 属性

BoneKeyData[] Bone\_

说明：骨信息

float Length

说明：指长度

Vector TipPosition

说明：指尖位置

FingerType type

说明：类型

* + - 1. BoneKeyData

说明：

* + - * 1. 方法

Load(Bone bone)

功能：将Leap.Bone内的信息复制到BoneKeyData

输入：指(Leap.Finger)

输出：无

返回：无

Transform(Matrix M)

功能：根据翻转矩阵翻转骨

输入：翻转矩阵

输出：无

返回：无

* + - * 1. 属性

Vector PrevJoint

说明：前节点

Vector NextJoint

说明：后节点

Vector Direction

说明：方向