实验1 光学动作捕捉系统OptiTrack的使用

**一、实验目的**

1. 熟悉动捕系统Optitrack的实现原理

2. 掌握Optitrack的使用

**二、实验设备**

1. 动捕系统Optitrack

——运动捕捉摄像机

——运动捕捉软件Motive

2. 标定工具

3. Ubuntu系统电脑

**三、实验原理**

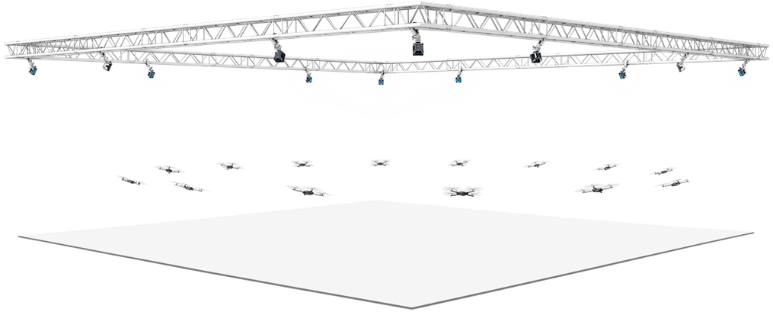


图1-1 动捕系统下的无人机飞行

室内环境下，由于GPS信号的接收限制，因此缺乏高精度的飞行定位方法。将三维红外运动捕捉系统应用于无人机室内飞行定位，这种方法精度高且稳定可靠，受到国内外的多旋翼无人机的研究机构或专业公司的特别关注。

该方案需要给多旋翼无人机上布置被动标记点（近红外光线下反光）或主动标记点（主动发射近红外光线），红外运动捕捉系统通过识别标记点来认知无人机，实时捕获无人机飞行姿态和位置信息。

控制计算机通过WIFI、蓝牙或无线数传等数据传输模块输出飞行控制指令给无人机，无人机接收到飞行指令后自主飞行。运动捕捉系统实时捕捉到无人机的六自由度信息，并传回控制计算机。控制计算机根据预设的飞行轨迹和实际的飞行轨迹的差别，修正飞行参数，并重新发送飞行控制指令给无人机，从而控制无人机高精度的、平稳的、无偏差的飞行。

相比于其他飞行定位的方案，三维红外运动捕捉系统有以下优势：

1.室内定位：室内环境下没有GPS信号，无人机无法定位；

2.高速：摄像机的拍摄速度在100FPS以上，能够提供冗余的定位信息；

3.高精度：位置坐标精确到0.1mm，远高于其他的定位方案；

4.鲁棒性：跟踪固定在无人机上的刚体，跟踪数据几乎不会丢失；

**四、实验步骤**

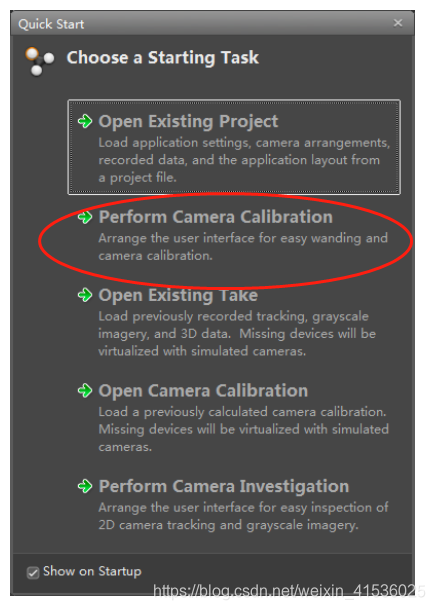
**1. 动捕系统Optitrack的标定**

光学运动捕捉系统首先要进行标定才能够使用。在标定过程中，系统通过捕获的图像计算每个相机的位置、角度、以及畸变。通过标定，Motive构建3D捕获空间。具体来说，通过观察多个同步相机的2D图像，关联到通过三角测量得到的每个相机已知的标定Marker的位置。注意，如果标定之后相机有任何变化，需要重新进行标定。即便外部设置没有改变，随着时间的变化、温度其他环境的波动，标定精度会下降，建议每天标定一次系统。

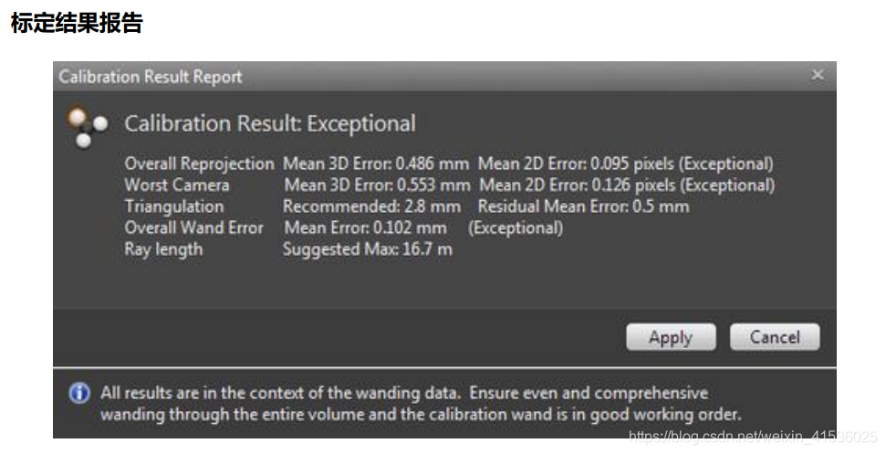
 

图1-2 标定杆 图1-3坐标轴直角标定仪

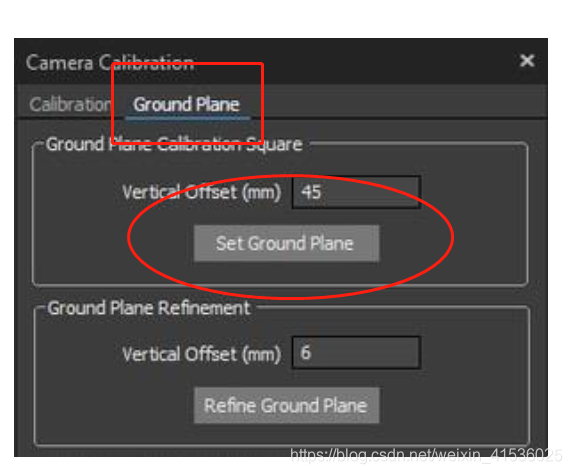
1. 准备、 优化捕获设置



1. 在相机预览窗口（Camera Preview）中，点击清除现有的Mask区域；
2. 打开标定窗口，使用Mask Visible去除场地中不能移除的反光物体；
3. 采集标定采样点（Wanding）；
4. 计算、检查标定结果；



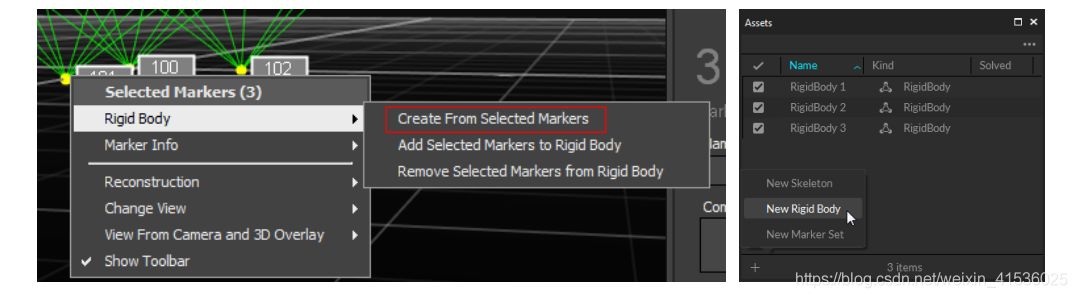
1. 放入L型标定杆，设置地面。



**2. Motive参数设置**

1. 创建刚体（实验二需要）

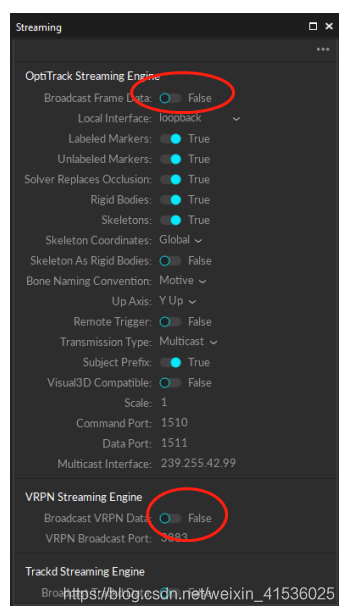
将所要创建模型的刚体贴好标记点，放置在标定好的场地内，在Motive的3D视图中框选住代表刚体模型的标记点。然后点击右键选择Rigid Body下的Create From Selected Markers；或者使用热键：Ctrl + T；或者在Asset窗口点击，选择New Rigid Body。通过以上三种方式都可以对选中Marker点完成刚体创建。



1. 数据流输出

Motive可以通过多种方式，将跟踪的数据实时传输到其他程序：Autodesk MotionBuilder、 Visual3D、 Unreal Engine4、 3dsMax、 Maya(VCS)、 VRPN、 Trackd。 Natnet SDK支持用户创建自定义客户端用来接收数据，数据流选项不需要单独的许可证。常见的运动捕捉应用依赖于实时跟踪，OptiTrack系统旨在提供低延时的数据，Streaming窗口中配置选定的服务器通过网络广播数据。可通过View选项下Data Streaming Pane

红色部分都要打开：



**3. Ubuntu计算机获取信息**

操作环境：Ubuntu 20.04 + ROS : neotic

目的：通过Optitrack获得刚体（crazyflie2.0）的姿态信息，并发送给ROS。

1. #创建文件夹获取代码包
2. mkdir -p ~/catkin\_ws/src
3. cd ~/catkin\_ws/src
4. git clone https://github.com/clearpathrobotics/vrpn\_client\_ros.git
5. sudo apt-get install ros-kinetic-vrpn
7. #编译
8. cd ~/catkin\_ws
9. catkin\_make
11. #PING一下网络是否连通
12. ping 192.168.1.2
14. #启动ROS节点
15. roslaunch vrpn\_client\_ros sample.launch  server：=192.168.1.2

**五、实操任务**

独立完成实验步骤1至3，在上位机电脑获取无人机位置与姿态信息并显示在终端上。