ESE 519, Fall 2015
Real-Time and Embedded Systems
Using the VR wall setup and OptiTrack Motive Tutorial

InVIRT-Virtual Reality Experience

JITESH GUPTA
PRITHVI GANESH KARAPA

Introduction

This tutorial will act as a quick start guide to help you understand the VR wall set up in xLab and how to use it with OptiTrack Motive. Moreover, it will also assist you in integrating tracking data collected from OptiTrack with your preferred VR application development tool, for instance, Unity3D, VRUI, MiddleVR etc.
In this tutorial, we will particularly be focusing on Unity3D.

Getting Started

In order to get started, you should have the following components:-

- Grayscale IR cameras (already setup)
- IR emitters (already setup)
- Standard Calibration Wand and Square (available in Sig Lab, CGGT)
- Retro-reflective markers (easily available online)
- Powerful Windows machine running OptiTrack Motive and Unity3D

Understanding the motion capture system

To start with, the first step should be completely understanding how the marker based motion tracking system works. Though you can search the web to get in-depth understanding of the working, here is a brief overview of the basics:-

Motion capture is the process of recording a live motion event and translating it into actionable data that allows for a 3d recreation of the performance. In other words, transforming a live performance into a digital performance. There are many types of motion capture systems, for example, electromagnetic tracking systems, acoustic tracking systems, optical tracking, mechanical tracking systems etc. and as their name suggests, they work on completely different principles.

A marker based motion tracking system (optical tracking) uses light to calculate a target's orientation along with position. The signal emitter typically includes a group of infrared LEDs and the sensors consist of nothing but only cameras which can understand the infrared light that has been emitted. Rigid bodies, consisting of retro-reflective markers reflect the infrared radiations back to the IR emitting source with minimum scattering, which is why the camera sensor is attached very close to the IR emitters. We suggest you to read this case study in order to build a very good knowledge about optical motion tracking systems.

Calibration of the motion capture system

In this section, we will discuss how to calibrate our optical motion capture system. Calibration is a very crucial step for to ensure accurate motion tracking. This phase helps in calibrating the IR camera sensors so that they can localize efficiently.

A brief introduction to how calibration works in OptiTrack Motive before we jump into more details. As each of the cameras see the retroreflective markers, they capture 2D images which are then processed by motive to generate a 3D point using triangulation i.e. It uses the images from two or more cameras to identify the location of the marker in 3D space using the idea of calculating the angles from the camera to the object.

Calibration Wand

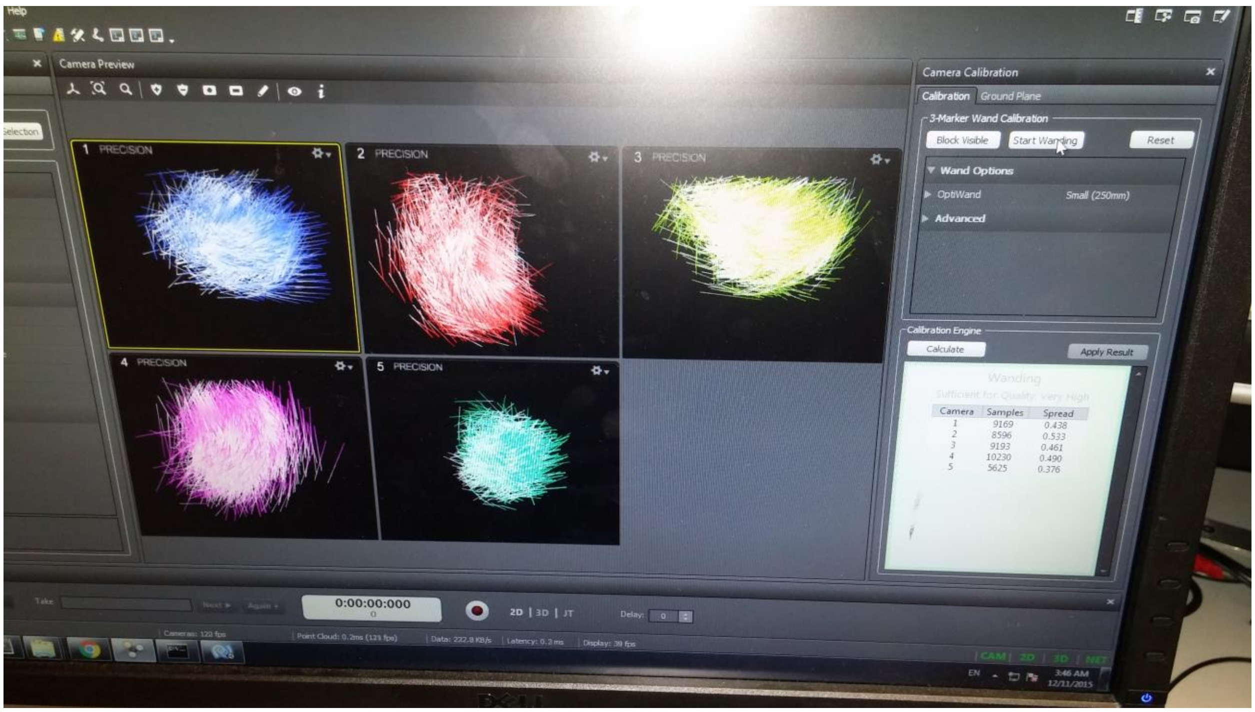
In order to start calibration, you need to have a calibration wand. Calibration wands are devices that have a set of 3 markers placed very precisely on a straight line at certain distances from each other. The farthest distance between the markers is used to identify the calibration wand. For ex, in a 250 mm calibration wand, the first and the last markers are placed precisely at a distance of 250 mm or 25cm.

Tracking Region

Once you get hold of a calibration wand, make sure that all the cameras are detecting the calibration wand. If you don’t see this happening, try to change the angle of the cameras. Make sure that all the cameras capture the calibration wand in the region you wish to be tracked. Note that the tracking region varies based on the distance between the cameras and also the angle and height at which the cameras are placed. More detail on this aspect is covered in a later section.

Collecting Samples

Once all the cameras are detecting the calibration wand in the tracking region, start the calibration process in motive by selecting the “start wanding” option under calibration tab. Once you hit this button, you should see samples being collected in all the cameras, these are identified as straight lines having a different color for each camera. The calibration pane shows a count of how many samples have been collected in each camera. The image below shows the calibration results that we achieved:-



Localization

Continue the calibration process until each camera has sufficient samples. Typically, you should look to achieve around 1500 samples in each camera for good calibration. Once the cameras receive sufficient number of samples you will see the calibration pane turn green in color. When you are satisfied with the number of samples that you collected, click on “read to apply” to and save the calibration for later use. Once you hit ready to apply motive runs a localization algorithm to localize the 2D point coordinate system to 3D.

Setting the ground plane

The ground plane needs to be set using the calibration square. The calibration square consist of three markers places in an 'L' shape the edges denoting each of the x, y and z axes. Setting the ground plane aligns the tracking surface's coordinates with that of OptiTrack Motive's. Once the ground plane is set, go ahead and save it for later use.
Now that entire calibration phase is complete, use the volume accuracy tool to measure the accuracy at which the calibration wand is being tracked.

You can visit the OptiTrack official website for detailed explanation of the calibration and rigid body creation process.

Creating rigid bodies

Careful creation of rigid bodies is a very important for a motion capture system. To create a rigid body, you need to have a minimum of three markers. Make sure that the positions of the markers are fixed and are not varying, only then the Motive will be able to track it in every position and orientation.

Note that the markers should be placed at different distances if you want to distinguish between two rigid bodies.

Selecting your application development tool

There are many toolkits available to develop virtual reality applications such as VRUI, MiddleVR, Unity3D, WorldViz etc. We started with VRUI and were able to synchronize OptiTrack with it but we faced a lot of difficulties in compiling and writing basic programs because of a lack of examples, community support and proper API documentation.

We then shifted to Unity3D because it has very large community support, easy to advanced level tutorials, detailed explanation of the API and a very efficient physics engine. We suggest you to use Unity3D for application development but if you are interested in using VRUI, we can share the conversation we had with Oliver Kreylos, who developed the VRUI toolkit.

Unity3D

As said above, you can find easily find beginner to advanced level application development tutorials in Unity3D. Moreover, there is a huge community support for Unity, therefore, you will be able to solve all your problems by searching through different forums. The main important component to create a virtual reality application is to integrate the tracking system data with your application development platform which is explained in the next section.

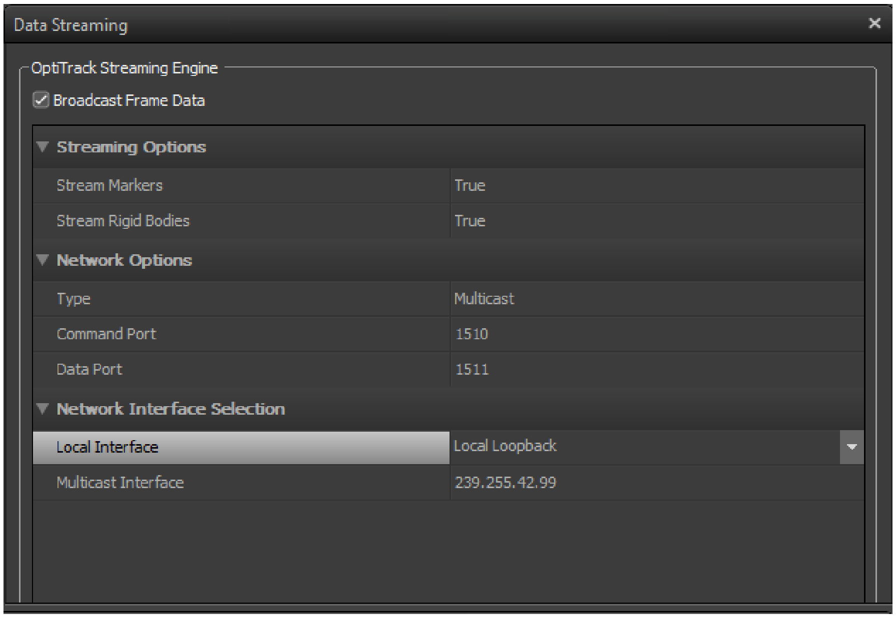
Data Integration

We used basic socket programming to bundle the tracking data from OptiTrack into IP packets and sent them on local loopback to Unity because we ran both the tools on the same Windows machine. We stream the data from OptiTrack using OptiTrack NatNet (Natural Network) SDK which is used for data streaming on Local loopback and can easily be downloaded from OptiTrack's software products. You can use also look into Virtual Reality Peripheral Network (VRPN) interface if you want. Following are the steps to enable motion tracking data streaming:-

Data Streaming from OptiTrack

- **Open the "Data Streaming" pane in the OptiTrack menu.** Under OptiTrack Streaming Engine, set local interface to **Local Loopback** if you want the data from OptiTrack to be rerouted to the same machine on which OptiTrack is running i.e. if you have your game engine running on the same machine. If you have your game engine running on a different machine you will have to forward the data by providing that machine's IP address.

- **Select "Broadcast Frame Data" in the data streaming pane**. After doing this, cross verify with the image below so that you have correctly configured the data streaming part. If you configuration does not match, then change the settings as shown in the image below:-



- If your preferred game engine is Unity, we have a readily available script which forward the data from OptiTrack NatNet SDK to Unity, All you have to do is run it and it will start streaming the data of all the rigid bodies that you have created in OptiTrack. This script is named **UnitySample** and you can find it under the following directory in the github repo:-

  MotiveToUnity\MotiveToUnity_v2.1\NatNet_SDK_2.7_UnitySampleModifiedToEnableMultipleRigidbodies\Samples\bin\Unity Sample

- Run this script and you should see data frames being sent on the command window.

Controlling game object in Unity

Once you have started streaming data from OptiTrack, you also need something running on the Unity side to receive and use that data. We wrote C# scripts to unpack this data and get position (x,y,z) values of the rigid bodies. You can find these scripts under the OptiTrack directory in the github repo. This section will describe you how to use these scripts to actually control a game object in Unity. We have prepared a video describing this process and we assume that you have gone through the basics of Unity. In this video, we demonstrate how to control the bird in the Flappy Bird game using a rigid body.

References

- http://www.vicon.com/what-is-motion-capture
- https://www.ptgrey.com/case-study/id/10385
- http://www.vrs.org.uk/virtual-reality-gear/tracking.html
- http://www.optitrack.com/support/software/motive-tracker.html