Tracking objects using OptiTrack system and ROS

Made by Or Hanoch for MSS at the University of the Witwatersrand

June 26, 2018

Contents

0 Introduction

This guide is intended to help someone new to Optitrack to get started with initial setup, tracking objects using Optitrack Motive software, and transfering tracking data to a Linux machine running ROS.

This guide was tested using Windows 7 with Optitrack Motive Tracker 2.0.1 and Ubuntu 16.04 with ROS Kinetic, using a NetGear Nighthawk router to connect the two.

1 Pre-Setup

1.1 General

In order for the Optitrack system to function properly you will need to make sure that:

1. You have reduced the amount of infra-red light source coming into your tracking area.
   (a) IR Light sources:
      i. Sunlight
      ii. IR lights could be emitted from sources such as incandescent, halogen, and high-pressure sodium lights or any other IR based devices.
2. Avoid reflective flooring. The IR lights from the cameras could be reflected by them and interfere with tracking. If this is inevitable, consider covering the floor with surface mats to prevent the reflections.
3. Avoid flexible or deformable flooring; such flooring can negatively impact your system's calibration.

More detailed information can be found in the Optitrack documentation at:
https://v20.wiki.optitrack.com/index.php?title=Prepare_Setup_Area

1.2 Required Hardware and Software

1. 1st PC running Windows 7
2. 2nd PC running Ubuntu 16.04 (or any other Linux distribution you have ROS working on)
3. NetGear Nighthawk router (or any other router)
4. Optitrack Motive Tracker hardware license dongle (USB stick)
5. At least 2 Optitrack Flex13 cameras mounted (on tripods or on a wall or in any other configuration.
6. An Optitrack Optihub2
7. Long USB cables provided by Optitrack

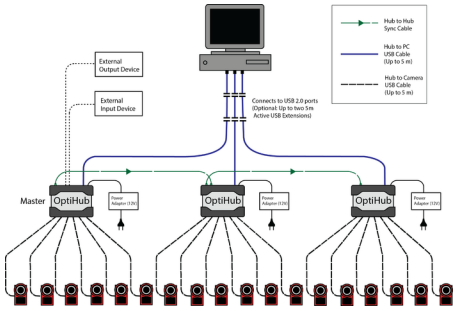


Figure 1: Flex13-OptiHub2-PC Diagram

1.3 Connecting Everything

1. Connect both computers to the NetGear router using Ethernet cables.
2. Connect the Flex13 cameras to the OptiHub2 using the long USB cables.
3. Connect the OptiHub2 to the Windows PC using a long USB cable.
4. Connect the Motive hardware license dongle to the Windows PC.
5. More information on setting up the area for the Optitrack system can be found in the Optitrack Documentations
   (a) Useful Docs:
   https://v20.wiki.optitrack.com/index.php?title=Camera_Placement
   https://v20.wiki.optitrack.com/index.php?title=Cabling_and_Wiring

1.4 Software Setup

1. **Operating Systems:**
   (a) As mentioned one PC should have Windows 7 formatted on it, and another should have Ubuntu 16.04
      i. Make sure you install all the drivers on the Windows 7 PC.
2. **Optitrack Motive (on Windows PC):**
   (a) Download Optitrack Motive Tracker on the Windows PC from the Optitrack website:
   http://www.optitrack.com/downloads/
   (b) Install Optitrack Motive Tracker.
      i. Follow the installation wizard (next all the way through).
   (c) Ask the administrator (Shunmuga Pillay as of the writing of this guide) for the license key file and copy it to:
   C:\ProgramData\OptiTrack\License
   (d) You should now be able to open and use Optitrack Motive Tracker.

i. If you have the cameras connected already then you should see their drivers being installed automatically and afterwards they should appear in Motive.

3. **ROS (on Ubuntu PC):**
   (a) Follow the ROS tutorials and install ROS Kinetic.
      i. ROS Tutorials can be found at:
      http://wiki.ros.org/ROS/Tutorials
4. **PuTTY (on Windows PC):**
   (a) In order to not need to continuously switch between computers using PuTTY to ssh from the Windows machine to the Ubuntu machine is very helpful.
      i. PuTTY can be downloaded for Windows from:
      https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
   (b) Otherwise a hardware KVM can also be very convenient.

2 Network Setup

1. If using a single drone you want to define your router as a bridge for the drone's wifi router.
2. If using multiple drones you want to define the router as its own wireless and LAN network and connect the drone and PCs to it.
3. Instruction on how to do this with the NetGear Nighthawk can be found in a separate guide.

3 Installing and Launching vrpn_client_ros (on Ubuntu Machine)

IMPORTANT NOTE:
vrpn_client_ros should **not** be confused with ros_vrpn_client! ros_vrpn_client is a different, and older, package intended for ROS Hydro!
vrpn_client_ros found at: http://wiki.ros.org/vrpn_client_ros

3.1 Short Option (untested)

Installing from Ubuntu repositories directly:
(This has been recommended to me after I finished installing the long way - it should work, but I can not vouch for it.)

Run the command:

```
sudo apt-get update
sudo apt-get install ros-kinetic-vrpn-client-ros
```

3.2 Long Option (tested)

Install from source:
(This option is longer, but is the way I did it, and I can vouch for it.)

1. Go to catkin src folder:

   ```
   cd ~/catkin_ws/src
   ```

2. Clone the repository:

   ```
   git clone https://github.com/ros-drivers/vrpn_client_ros.git
   ```

3. Update Ubuntu repositories before installing dependencies:

   ```
   sudo apt-get update
   ```

4. Go to cloned Github repository and install dependencies using:

   ```
   cd <directory of cloned github repository>
   rosdep install --from-paths .
   ```

5. Go to catkin home directory (may differ if you made it differently when installing ROS):

   ```
   cd ~/catkin_ws
   ```

6. "make" catkin:

   ```
   catkin_make
   ```

7. Source the setup.bash file to get ROS to auto-complete for the vrpn_client_ros package.

   ```
   source ~/catkin/devel/setup.bash
   ```

3.3 Launching vrpn_client_ros (on Ubuntu PC)

1. Make sure the network is working between the Ubuntu PC and the Windows PC (can be done using "ping")
2. To run vrpn_client_ros run the command:

   ```
   roslaunch vrpn_client_ros sample.launch server:=<ip_of_windows_machine>
   ```

   i.e. for windows machine IP 192.168.1.102

   ```
   roslaunch vrpn_client_ros sample.launch server:=192.168.1.102
   ```

   (a) Note: You can find the IP on the Windows PC by going to the command line (winkey+R -> type "cmd") and entering

   ```
   ipconfig
   ```

3. You should see a "connection established" line at the end
4. Check if the node is running with:

   ```
   rosnode list
   ```

   (a) You should see vrpn_client_ros there
5. If you already set up motive with the streaming configurations and you have a rigid body marked in your world you can see if the data is passed to ROS using:

   ```
   rostopic echo /vrpn_client_node/<name_of_rigid_body_in_motive>/pose
   ```

   i.e. for rigid body named “ardrone1”

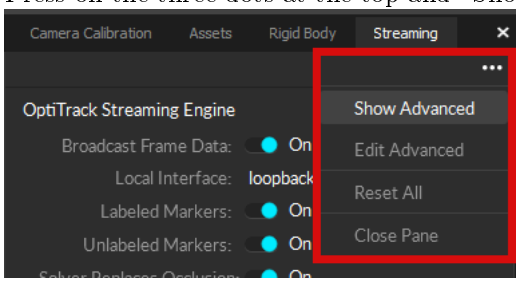
   ```
   rostopic echo /vrpn_client_node/ardrone1/pose
   ```

4 Configuring Motive (on Windows PC)

4.1 Streaming Data

This is mostly based on instructions from http://wiki.ros.org/mocap_optitrack (even though they reference an old version of motive) but I will explain what I did anyways.

1. In motive go to the streaming pane (using the button )
2. Press on the three dots at the top and "Show Advance”

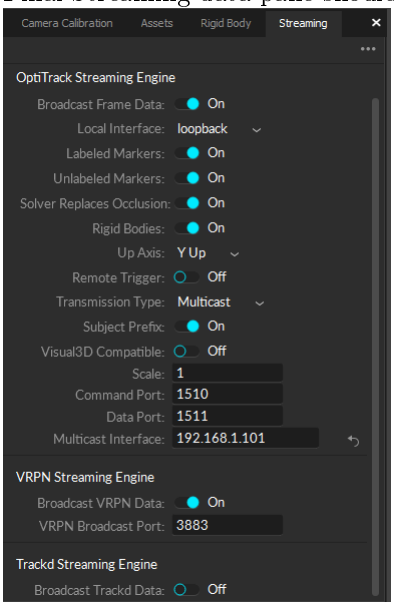


3. Not sure if this is relevant, but "local interface" is on "loopback"
4. Make sure "Rigid Bodies" is on
5. Set "Transmission Type" to "Multicast"
6. Set "Multicast Interface" to your ROS machines IP address
   (a) You can get the IP of your Ubuntu PC using the command

   ```
   ifconfig
   ```

7. Set "Broadcast VRPN Data" to on
8. “VRPN Broadcast Port" should stay 3883

Final Streaming data pane should look similar to:

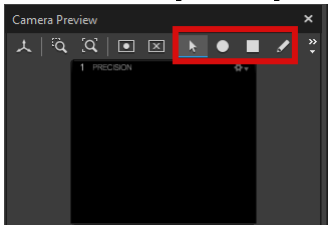


4.2 Calibrate Cameras

Visual example of calibration is needed to understand how to calibrate the cameras properly. The instructions and example in the Optitrack documentation video will be better than anything I can explain here: https://www.youtube.com/watch?time_continue=92&v=cNZaFEghTBU

That being said, in a nutshell:

1. Take all infra-red reflective markers out of your tracking area
2. Go to the calibration pane (using the button )
3. In the camera preview pane mark all of the infrared reflections using the buttons at the top:



4. In the calibration pane, under the “Wanding” section, click “Start Wanding”:

5. Using your Opti-wand walk in a grid like manner while moving the wand in circles (see example in aforementioned video)

6. When you finished wanding and the camera preview seems full enough click "Calculate" in the calibration pane under the "Calibration" section:

7. If all went well the pop-up window should tell you that the result is "Exceptional" - press Apply.

   (a) If for some reason you got a "Poor" result, or anything else thats not "Exceptional" try calibrating again until you get an "Exceptional" result.

4.3 Creating Rigid Body

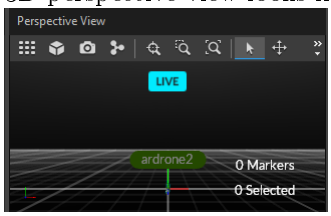
I followed the instructions in the Optitrack documentation:
https://v20.wiki.optitrack.com/index.php?title=Rigid_Body_Tracking

But here is that explanation re-written:

1. Assuming the cameras are connected properly you should see the dots from your markers appear on the screen.

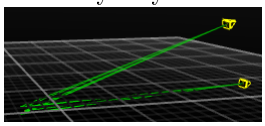
2. Change view to 3D perspective view pane if you are only in camera preview by pressing:

   3D perspective view looks like this:



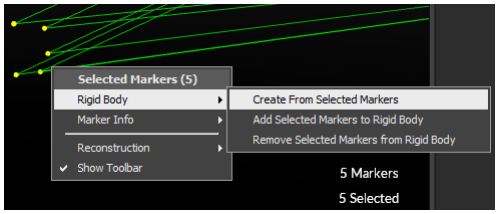
3. Select all associated rigid body markers in the 3D view pane by clicking and dragging around them.

   (a) These points are sometimes hard to see in 3D perspective view. Drag and make a large area, you will see many "rays" from the camera go to a point - your rigid body should be there, zoom in to see it.

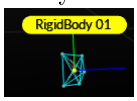


4. Create a Rigid Body asset from the selected markers:

   (a) From Perspective View: While the markers are selected, right-click on the perspective view to access the context menu. Under the Rigid Body section, click Create From Selected Markers.



5. Once the rigid body is created, the markers will be colored (labeled) and interconnected to each other. The newly created rigid body will be listed under the Assets pane.



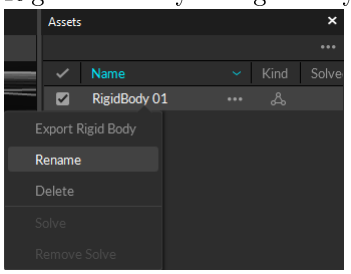
   (a) If the center point (aka pivot point) of your rigid body is not in the correct location try deleting the rigid body and creating it again.

      i. If this still doesn't work you can try looking at the "Adjusting Rigid Body Pivot Point" section in the above link (I haven't needed to try it yet)

6. Change rigid body name:

   (a) Go to the "Assets" pane (using the button )

   (b) Right click on your rigid body and "Rename"



   (c) Set the name to the same name you expect to be using in your future code/ROS

5 Check That It Is Working (on Ubuntu PC)

1. Run vrpn_client_ros run the command:

```
roslaunch vrpn_client_ros sample.launch server:=<ip_of_windows_machine>
```

   (a) Note: You can find the IP on the Windows PC by going to the command line (winkey+R -> type "cmd") and entering

```
ipconfig
```

2. Make sure that Motive Tracker is open in your Windows PC and that the rigid body is defined and streaming data is configured (from section 4)

3. Have a look at the data being streamed to the ROS topic of the rigid body:

   ```
   rostopic echo /vrpn_client_node/<name_of_rigid_body_in_motive>/pose
   ```

   i.e. for rigid body named "ardrone1"

   ```
   rostopic echo /vrpn_client_node/ardrone1/pose
   ```

   (a) If everything is OK you should now see XYZ positions as well as XYZW orientation:

```
header:
  seq: 676
  stamp:
    secs: 1523861569
    nsecs: 160093826
  frame_id: world
pose:
  position:
    x: 0.915969848633
    y: -0.303546786308
    z: -0.347537845373
  orientation:
    x: -4.48726932518e-05
    y: -3.48467146978e-05
    z: -0.000156470516231
    w: -1.0
```

6 Finalizing Calibration

This step will make the 3D axis in Motive correspond to those you are using in the real world.

6.1 Auto Calibration

Motive has an option to automatically set the axis corresponding to the calibration square (comes part of the Optitrack kit, see image). The best explanation for this can be found in the official Optitrack docs.

I will mention the steps here as well:

1. Put markers on the calibration square as follows:
   (a) 1 marker in the center
   (b) 1 marker on each axis - for the z axis use farther peg, for x axis use closer peg. (You can ignore notation for Z axis, you can use it as your X axis)
2. Put the calibration square where you want the origin of the axis to be.
3. Lift the feet of the square by screwing the legs in and out if your floor is uneven until the square is level.
4. In Motive:
   (a) Select the 3 markers

(b) In the calibration pane under the "Ground Pane" look at "Ground Pane Calibration Square"

i. Set "Vertical offset" to the distance (in mm) from the ground to the center of your marker. Without lifting the feet this was 45mm for me.

ii. Click "Set Ground Pane"

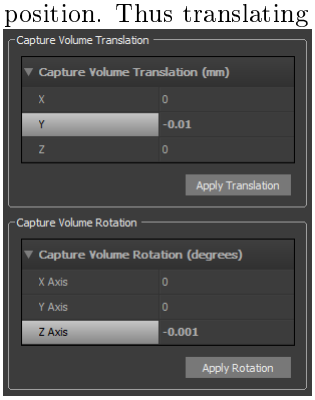
iii. You should see the orientation of the 3D axis in Motive change appropriately.

(c) If your floor isn't level you might also want to refine your ground plane. As I have not done that myself I will not explain it here, but you can find the official explanation in the docs mentioned above.

6.2 Manual Calibration

If the "Set Ground Pane" results aren't satisfactory you can manually adjust the ground plane. To do this:

1. Place your real drone (or any set of markers you can identify) where you would like the xyz origin to be
2. Center and orientate rigid body pivot:
   (a) Select your rigid body in the 3d perspective view pane (click and drag a box around it)
   (b) Go to "Edit" tab in the "Rigid Body" pane ()
   (c) Translate and rotate the pivot point until it is in the center of your object and the xyz axis of the object are what you want them to be.
3. You can now go to the "Ground Plane" tab in the "Calibration" pane () and make sure your plane and camera location is as expected
   (a) Play around with "Capture Volume Translation" and "Capture Volume Rotation"
      i. When you change a variable and apply it then it is not relative to a center point, i.e. it does not mean translate to these xyz coordinates. What it does do is translate/rotate in relation to its current position. Thus translating with x=1 will cancel out a previous movement of x=-1.



   (b) If you look at the vrpn_ros_client topic for your object on your Ubuntu PC you should be able to see the current position and orientation:

```
rostopic echo /vrpn_client_node/<name_of_rigid_body_in_motive>/pose
```

   i.e. for rigid body named "ardrone1"

```
rostopic echo /vrpn_client_node/ardrone1/pose
```

(c) Continue changing the translation and rotation until you get as close to zero as possible on all positions and orientations (except for w)

i. You should see that the xyz axis correspond to those on the world grid in Motive when you get this correct

Notice:
Depending on how many cameras you are using, how well the cameras were calibrated, and other factors (like if the cameras accidentally got moved any tiny bit since calibration) there might be some minor inconsistencies when moving the drone around. We want to minimize these, but getting rid of them completely is nearly impossible.

7 Congratulations

For multiple drones just make multiple rigid bodies with different names, and ROS will create a different topic for each drone according to the rigid body name, and receive data for it through that topic

You are now ready to work with ROS obtaining tracking data from Optitrack Motive Tracker!
Program to your hearts content!