Manual

Xela Sensor Server Node for ROS
v.0.0.9b

Table of contents

Using the Node:

Prerequisites:

Primary requirement is to run the code with **Python 2.7** as the files have been pre-compiled and therefore might give an error if running on different Python version

The following packages are required to run the sensor service and tools:

1) Tkinter
2) numpy
3) matplotlib (version 3+ recommended)
4) python-can
5) psutil (normally installed)
6) requests (to get update notification)

Set up

Manual

First copy the nodes to your catkin workspace folder (src).

Compile the nodes with `catkin_make`

Start `roscore`

Run the configuration tool `user@localhost:~$ rosrun xela_server xela_conf`

Start the server `user@localhost:~$ rosrun xela_server xela_server`

Start the sensor service `user@localhost:~$ rosrun xela_server xela_service`

Start the visualization tool (optional) `user@localhost:~$ rosrun xela_server xela_viz`

Automatic

First copy the nodes to your catkin workspace folder (src).

Compile the nodes with `catkin_make`

Start `roscore`

Run the Xela Command Centre `user@localhost:~$ rosrun xela_server xela`

Run all other elements directly from XCC.

Use

Access the stream by subscribing to the /xServerPub topic

For single set of data, use one of the following service calls:

user@localhost:~$ rosservice call /xServXY 1 2 Get X and Y from taxel 2 on sensor 1	values: [16439, 16647]
user@localhost:~$ rosservice call /xServXYZ 2 6 Get X, Y and Z from taxel 6 on sensor 2	values: [16451, 16517, 35901]
user@localhost:~$ rosservice call /xServX 2 1 Get X from taxel 1 on sensor 2	value: 16681
user@localhost:~$ rosservice call /xServY 2 2 Get Y from taxel 2 on sensor 2	value: 16721
user@localhost:~$ rosservice call /xServZ 2 3 Get Z from taxel 3 on sensor 3	value: 37009
user@localhost:~$ rosservice call /xServStream 1 Get full sensor data from sensor 1	xyz: [1: [16457, 16553, 32057], 2: [16775, 16958, 31886]…]

Supported USB-CAN devices

Following USB-CAN devices are supported by our ROS Node:

- esd CAN-USB/2 (bus: socketcan, default channel: can0)
- VSCom USB-CAN Plus (bus: socketcan, default channel: slcan0)
- CANable/CANable Pro (bus: socketcan, default channel: can0)
  needs candlelight firmware
- PEAK USB-CAN (bus: pcan, default channel: CAN_USBBUS1)

With can0 (or with any can*n* device), you have to make sure that the network is active by running following commands:

```
user@localhost:~$ sudo ip link set can0 type can bitrate 1000000
user@localhost:~$ sudo ip link set up can0
```

With slcan0 (or with any slcan*n* device), you have to make sure that the network is active by running following commands:

```
user@localhost:~$ sudo slcand -o -s8 -t hw -S 3000000 /dev/ttyUSB0
user@localhost:~$ sudo ifconfig slcan0 up
```

Service types

Service name	Service message type	Description
xServX	XelaSensorX	Get x co-ordinate for taxel
xServY	XelaSensorY	Get y co-ordinate for taxel
xServZ	XelaSensorZ	Get z co-ordinate for taxel
xServXY	XelaSensorXY	Get x and y co-ordinates for taxel
xServXYZ	XelaSensorXYZ	Get x, y and z co-ordinates for taxel
xServStream	XelaSensorStream	Get x,y and z co-ordinates for the sensor (all taxels on sensor)

Example usage:

```
#!/usr/bin/env python

import rospy

from xela_sensors.srv import XelaSensorXYZ

import sys

rospy.init_node('use_service')

#wait the service to be advertised, otherwise the service use will fail
rospy.wait_for_service('xServXYZ')

#setup a local proxy for the service (we will ask for X,Y and Z data)
srv=rospy.ServiceProxy('xServXYZ',XelaSensorXYZ)

#use the service and send it a value. In this case, I am sending sensor: 1 and taxel: 3
service_example=srv(1,3)

#print the result from the service
print(service_example)
```

Common errors

Error	Reason
Program not responding to CTRL+C	There are few functions that have disabled interrupt transfer during compiling. To exit, use pkill -9 xela_server (or which ever function is not responding)
Unable to register with master node [http://localhost:11311]: master may not be running yet. Will keep trying.	Node couldn't communicate with the ROS master node. Make sure it is running. If you use Xela Command Centre, you can click Start next to ROS Core
Error connecting to CAN: IOError:[Errno 19] No such device	No CAN device found. Make sure your CAN-USB device is connected, accessible for all users (pulled up) and set in the configuration correctly (see /etc/xela/xServ.ini)
Error writing config file: IOError: [Errno 2] No such file or directory: '/etc/xela/xServ.ini'	Ensure there is /etc/xela folder and that it has 777 permissions
xela_viz is not starting and showing errors with *xkcd*	Your version of matplotlib might be outdated as it doesn't support the colors we use. Please update it and try again

If you find errors, not listed in this file, please send an email regarding it to info@xelarobotics.com
Do not forget to attach files from /etc/xela folder with the terminal log (if you copy-paste, there might be some strange characters, don't worry about them, we can read \033[0m) and description of the failure.
If you have made your own code that is causing errors, please attach it as well.