

Il rover si collega automatica alla rete MiviaLab con l'indirizzo IP presente sul rover. Es. 10.0.1.12

Con il pc collegarsi alla rete MiviaLab password **2018!Mivialab** .

Aprire il terminale e lanciare il comando `ssh mivia@<IP_ADDRESS>` dove <IP\_ADDRESS> deve essere sostituito con l'indirizzo IP presente sul rover.

A questo punto si è avuto accesso al terminale della Jetson.

I comandi da lanciare prima di tutto sono :

```
cd catkin_ws && source devel/setup.bash && export ROS_MASTER_URI=http://<IP_ADDRESS>:11311 && export ROS_IP= <IP_ADDRESS>
```

E poi:

```
roslaunch realsense2_camera rs_camera.launch
```

Questo comando lancia il nodo ros che si occupa dell'acquisizione delle immagini dalla realsense.

Per tale comando è possibile specificare i seguenti argomenti:

```
color_width:=<COLOR_WIDTH>
color_height:=<COLOR_HEIGHT>
depth_width:=<DEPTH_WIDTH>
depth_height:=<DEPTH_HEIGHT>
```

ecc... specificati tutti nel file `rs_camera.launch`

Una volta lanciato il nodo, aprire un altro terminale con gli stessi comandi.

Quindi di nuovo `ssh mivia@<IP_ADDRESS>`

```
cd catkin_ws && source devel/setup.bash && export ROS_MASTER_URI=http://<IP_ADDRESS>:11311 && export ROS_IP= <IP_ADDRESS>
```

Adesso è necessario lanciare il nodo dei motori con il seguente comando:

```
roslaunch mivia_rover rover_mover.py
```

Aprire adesso un altro terminale e lanciare gli stessi comandi ssh e catkin per poter accedere alla Jetson e settare le variabili d'ambiente.

Successivamente per poter lanciare il nodo contenente la logica di controllo, eseguire il seguente comando:

```
roslaunch my_controller my_controller.py
```

Per visualizzare l'immagine processata sul PC in locale è necessario avere installato ROS sul pc e aver scaricato il pacchetto `vision_ws`. Una volta scaricato il pacchetto è necessario eseguire i seguenti comandi:

```
cd vision_ws
catkin clean
catkin build
```

Per l'installazione si rimanda alle slide di ROS.

Per quanto riguarda invece l'avviamento del nodo di debug eseguire i seguenti comandi:

```
cd vision_ws && source devel/setup.bash && export ROS_IP=<IP_PC> && export  
ROS_MASTER_URI=http://<IP_ADDRESS>:11311
```

```
roslaunch debugger_pkg image_processed_listener.py
```

Dove <IP\_PC> è l'ip della macchina connessa alla stessa rete della Jetson su cui si stanno eseguendo i comandi e <IP\_ADDRESS> è l'ip della Jetson .

### **Modificare il file**

Per la modifica della logica di controllo, si consiglia di modificare il file ~/catkin\_ws/src/my\_controller/process.py sul pc in locale e poi caricare il file sulla Jetson. E' possibile caricare dei file sulla Jetson utilizzando un qualsiasi client FTP o utilizzare il comando scp. Ad esempio è possibile eseguire scp my\_file.py mivia@<IP\_ADDRESS>:~/catkin\_ws/src/my\_controller/ per poter copiare il file my\_file.py dal pc alla Jetson nel progetto. Utilizzando il flag -r è possibile caricare anche un'intera cartella.

Altrimenti è possibile utilizzare vim o nano per modificare il file direttamente sulla Jetson.