DJI TELLO EDU* ROS PACKAGE

** We used the DJI TELLO **EDU** version of the drone. The DJI Tello EDU supports all of the same functions and features as the standard Tello, however it also benefits from an upgraded SDK, Mission Pads, and swarm flying.*

We created a ROS package with following **functionalities**:

- Teleop package to control and move 1 Tello.
- Teleop package to control and move 1 Tello with video streaming.
- Teleop package to control and move a Tello-swarm.
- Package that starts a drone show
- Script to test the battery state of the drone.
- Script to connect the DJI TELLO to a custom Wi-Fi network.

Prerequisites:

- **Ubuntu 20.04.6 LTS Desktop Image**
  You need the above version of Ubuntu either installed as your main Operating System or on a Virtual Machine.
  See the following link for installation instructions: Ubuntu 20.04.6 LTS (Focal Fossa)
- **ROS NOETIC**
  We need all components of the ROS framework installed on your Ubuntu Machine.
  See the following link for installation instructions: noetic/Installation/Ubuntu - ROS Wiki

Basic functionalities of the DJI TELLO

- **IP BASED**
  The DJI Tello is an IP based drone. The SDK connects to the drones via the Wi-Fi UDP protocol, allowing users to control drones through text commands.
  There are 2 main ways to connect to the DJI TELLO:
- **DJI TELLO as WIFI HOTSPOT**
  By default the drone acts as his own Wi-Fi hotspot. You can connect your laptop to this Wi-Fi network.
  The SSID of the Wi-Fi network looks like: **TELLO + 6 digit code** =>Example: TELLO-619A37.
  To test the connectivity you can use the default IP address 192.168.10.1 of the drone.

- **DJI TELLO connect to custom WIFI HOTSPOT**

  With my software program “Connect_to_custom_wifi” you can also connect the drone to a custom network.

  The drone gets an IP address from the DHCP server of your router. To find this ip address the easiest way is to check the DHCP pool of your router and filter on MAC address.

  The MAC address of the drone always **starts** with 8 fixed digits that are the same for every drone. And ends with **4 UNIQUE** digits:

  34:d2:62:f0:37:54.

  To use the script follow these steps:

  1) The script can be found in:
     ~/catkin_ws/src/dji_tello_pkg/scripts
     Change the variables SSID and password to the ones of your network.
  2) Connect to the wifi network of the drone.
  3) From your laptop ping to 192.168.10.1
  4) If successful => in command line enter: **python3 connect_to_custom_wifi.py**

IMPORTANT We need the IP address of the DJI Tello to connect via the Wi-Fi UDP protocol.

Remark: to use the drone with OPENCV and AI functionalities the most reliable way is to use the Wi-Fi network of the DJI TELLO.
If the drone is connected to a custom network. There are always a certain number of commands that gets lost in the network.

- **SDK**

  Software Development Kit. Our drone already has a set of commands it understands. We use the DJI TELLO SDK 3.0**.**
  This code will be used in our different ROS LISTENER NODES.
  The SDK can be found at the following link: Tello_SDK_3.0_User_Guide_en.pdf (djicdn.com).

- **BATTERY**

  To test the battery state of the drone follow these steps:

  1) The script can be found in:
     ~/catkin_ws/src/dji_tello_pkg/scripts
  2) Make sure you have a wifi connection to the drone
  3) In command line enter: **python3 checkbattery.py**

- **DJITELLOPY**

We use the djitellopy package for python. This package contains the following code:

- implementation of all Tello commands
- easily retrieve a video stream
- receive and parse state packets.
- control a swarm of drones.

TO install this package, enter the following command:

pip3 install djitellopy

See the following link for more info:

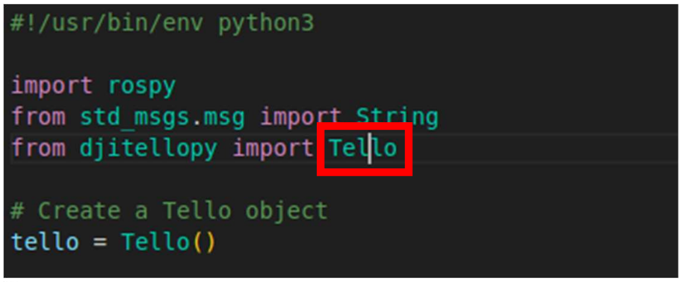
damiafuentes/DJITelloPy: DJI Tello drone python interface using the official Tello SDK. Feel free to contribute! (github.com)

In order to make the IP Connection do the following:

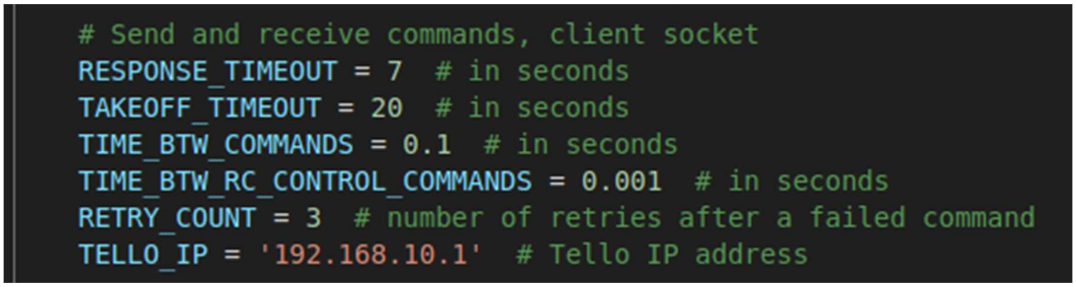
CONTROL 1 TELLO

Open the scripts folder. This can be found in: **catkin_ws/src/dji_tello_pkg/scripts.**
Open the python script with an IDE of your choice. Look where you do the import of the djitellopy package and click on the Tello package.

Change the TELLO_IP to the ip address of your drone.



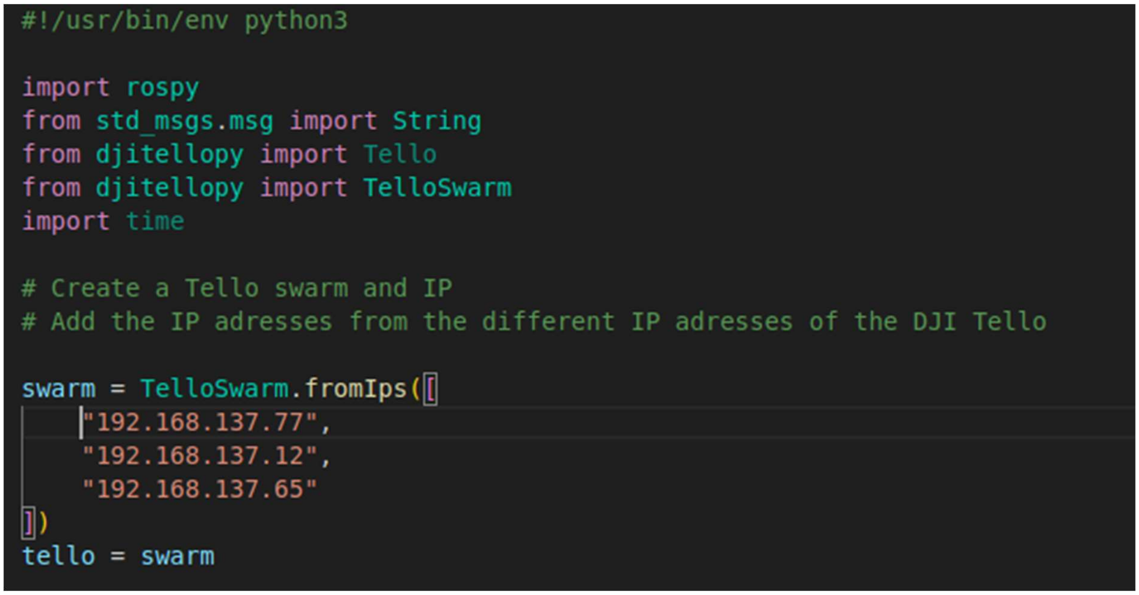
```
#!/usr/bin/env python3

import rospy
from std_msgs.msg import String
from djitellopy import Tello

# Create a Tello object
tello = Tello()
```



```
    # Send and receive commands, client socket
    RESPONSE_TIMEOUT = 7  # in seconds
    TAKEOFF_TIMEOUT = 20  # in seconds
    TIME_BTW_COMMANDS = 0.1  # in seconds
    TIME_BTW_RC_CONTROL_COMMANDS = 0.001  # in seconds
    RETRY_COUNT = 3  # number of retries after a failed command
    TELLO_IP = '192.168.10.1'  # Tello IP address
```

CONTROL SWARM

The same must be done for the drone swarm.
Add the IP address of the different drones to the python file.



```
#!/usr/bin/env python3

import rospy
from std_msgs.msg import String
from djitellopy import Tello
from djitellopy import TelloSwarm
import time

# Create a Tello swarm and IP
# Add the IP adresses from the different IP adresses of the DJI Tello

swarm = TelloSwarm.fromIps([
    "192.168.137.77",
    "192.168.137.12",
    "192.168.137.65"
])
tello = swarm
```

How to use the package?

We make use of the ROSLAUNCH tool to easily launch the necessary nodes.

When using roslaunch it is not necessary to already start the roscore server. A roslaunch will automatically start roscore if it detects that it is not already running.

- **Teleop package to control 1 Tello**

Enter the following command:

roslaunch dji_tello_pkg tello_teleop.launch

The launch file will start a ROS Listener and Publisher.

The lister node will on one hand connect to the DJI Tello with the use of the djitellopy package and on the other hand listen to ROS messages.

The TELEOP publisher will wait till the user send move commands with the keyboard to the DJI Tello. He then publishes the corresponding message.

- **Teleop package to control 1 Tello with video streaming**

Enter the following command:

roslaunch dji_tello_pkg tello_teleop_camara.launch

The launch fill does the same thing as the previous package but will also start the camera of the DJI Tello and open a video stream on your laptop using OpenCV.

- **Teleop package to control multiple DJI Tello drones**

With this package you can control 2 or more drones with one keyboard controller.

There are 2 options parallel or sequential mode:

In Parallel mode all the drone's move on the same time. In Sequential mode the drones move one after the other.

Enter 1 of the following commands:

roslaunch dji_tello_pkg tello_swarm_teleop_parallel.launch

roslaunch dji_tello_pkg tello_swarm_teleop_parallel.launch

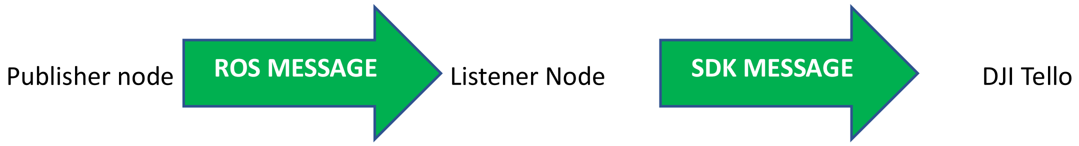
- **Package that starts a drone show**

This package starts a drone show for 1, 2 or more drones. Just add the IP address of the different drones to the python file.(See djitellopy).

Enter the following command:

roslaunch dji_tello_pkg tello_show.launch

ISSUES



The ROS node that publishes or listens to a ROS message works perfectly. The lister node translates this message to an SDK message that the drone can understand.

There is however an issue where the SDK message doesn't always reach the DJI Tello. Some of the following issues we have encountered.

- The SDK message doesn't reach the drone => **the drone does nothing and lands**
- One SDK message is executed multiple times by the drone => **crash**
- In a drone swarm: sometimes one drone reacts to the wrong SDK message =>**crash**
- In a drone swarm: both drones are not in 'ready' state the same moment. Its possible that one drone is already landing when the other is getting ready.