

# SJTU Drone PKG

AI ROBOT

Exported on 06/11/2021

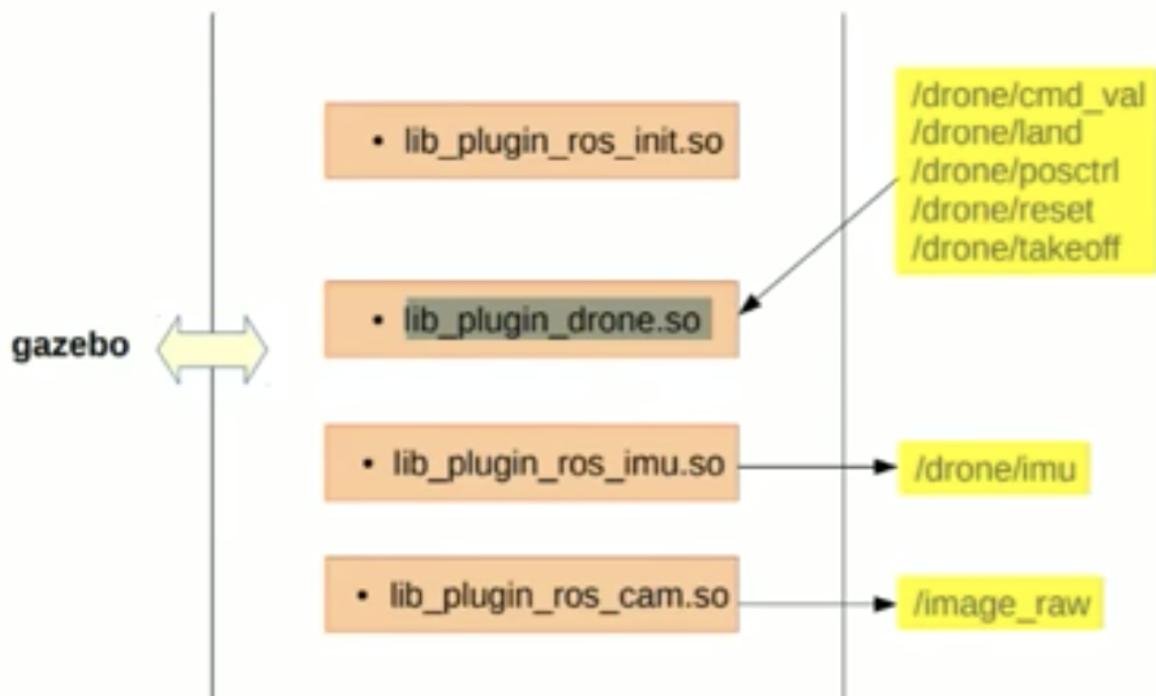
## Table of Contents

1	SJTU Drone pkg.....	3
2	Framework of SJTU Drone PKG.....	4
3	소스코드 clone 후에 build.....	5
4	roslaunch sjtu_drone simple.launch .....	6
5	시뮬레이션 시작 .....	7
6	rostopic list.....	8
7	rqt로 전면, 배면 카메라 관찰.....	9
8	이륙 .....	10
9	teleop_twist_keyboard로 조종이 가능하다 (어차피 cmd_vel을 사용하니까)....	11
10	land .....	12
11	world 폴더의 garching_kitchen.world에서 오류 수정 .....	13
12	simple.launch 수정 .....	14
13	YOLO ros wrapper 적용 .....	15
14	다시 드론 실행.....	16
15	yolo 실행 .....	17
16	키보드로 조종 .....	18

## 1 SJTU Drone pkg

- Gazebo의 Drone 시뮬레이터로 유명한 tum\_simulator를 fork에서 visual SLAM 등을 시뮬레이션하기 위해 Shanghai Jiao Tong University에서 만든 패키지
- 실제 원조는 ardron
- 지금 다룰려는 것은 sjtu drone pkg를 ROS melodic과 GAZEBO9에서 구동되도록 수정된 버전
- 주소는 <https://github.com/tahsinkose/sjtu-drone>

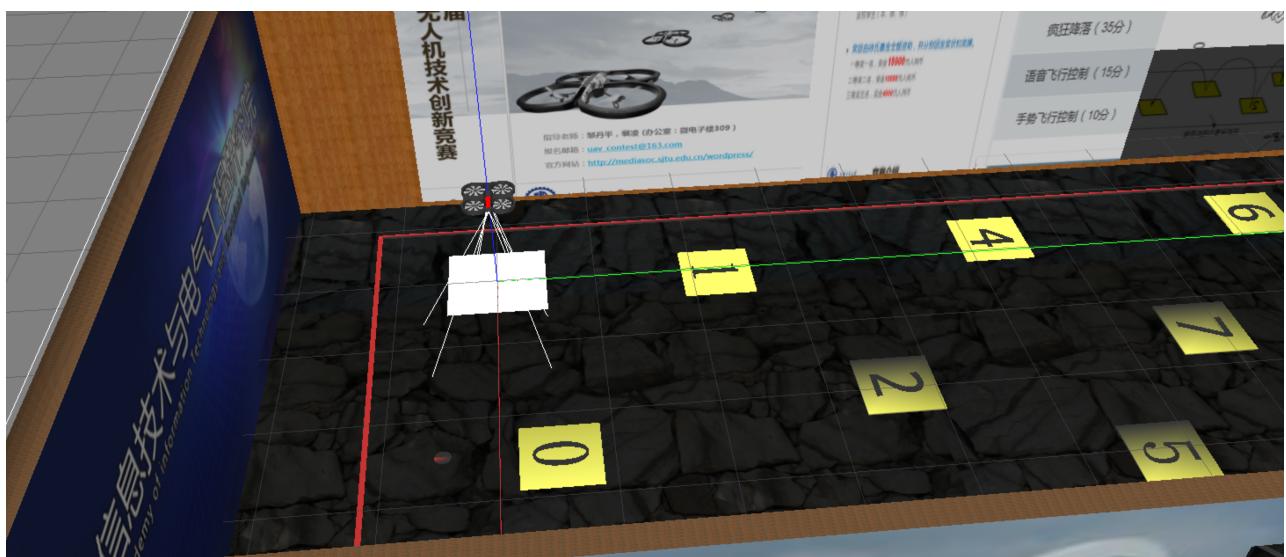
## 2 Framework of SJTU Drone PKG



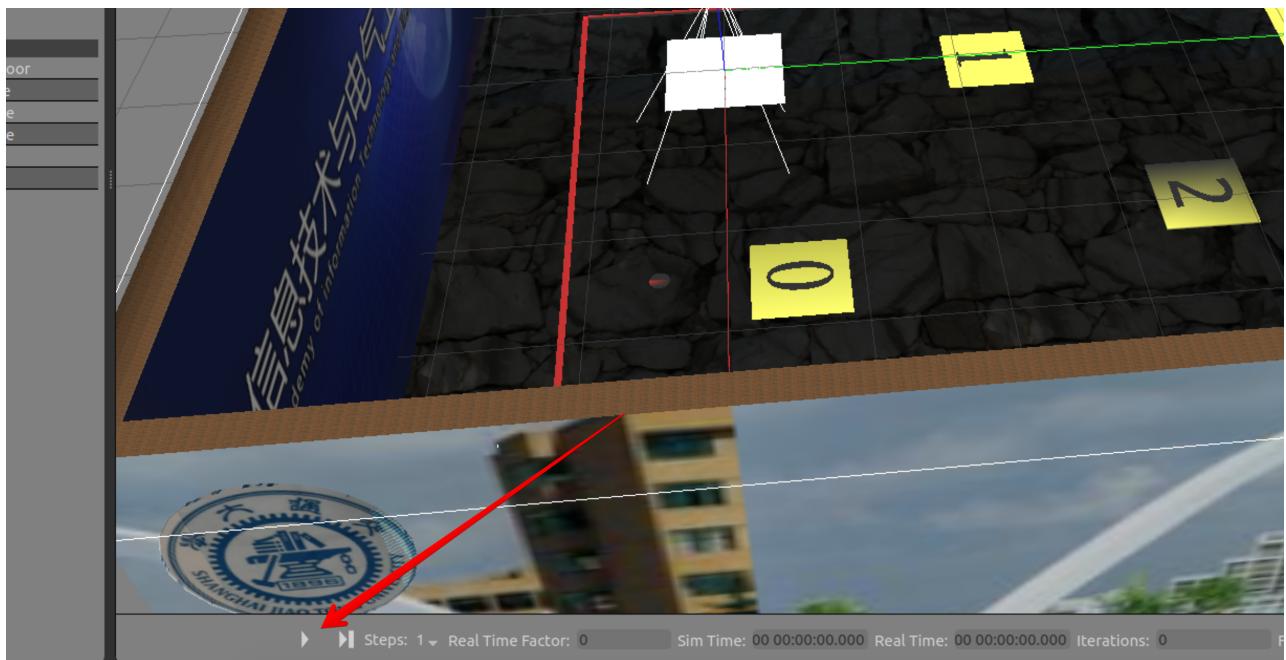
### 3 소스코드 clone 후에 build

```
pw@ros:~$  
pw@ros:~$ cd catkin_ws/src/  
pw@ros:~/catkin_ws/src$ git clone https://github.com/tahsinkose/sjtu-drone.git  
Cloning into 'sjtu-drone'...  
remote: Enumerating objects: 270, done.  
remote: Counting objects: 100% (270/270), done.  
remote: Compressing objects: 100% (211/211), done.  
remote: Total 270 (delta 56), reused 266 (delta 52), pack-reused 0  
Receiving objects: 100% (270/270), 18.23 MiB | 1.63 MiB/s, done.  
Resolving deltas: 100% (56/56), done.  
pw@ros:~/catkin_ws/src$ cd ..  
pw@ros:~/catkin_ws$ cd src/sjtu-drone/  
.git/ include/ launch/ meshes/ models/ src/ worlds/  
pw@ros:~/catkin_ws$ catkin build sjtu_drone  
-----  
Profile: default
```

4 roslaunch sjtu\_drone simple.launch



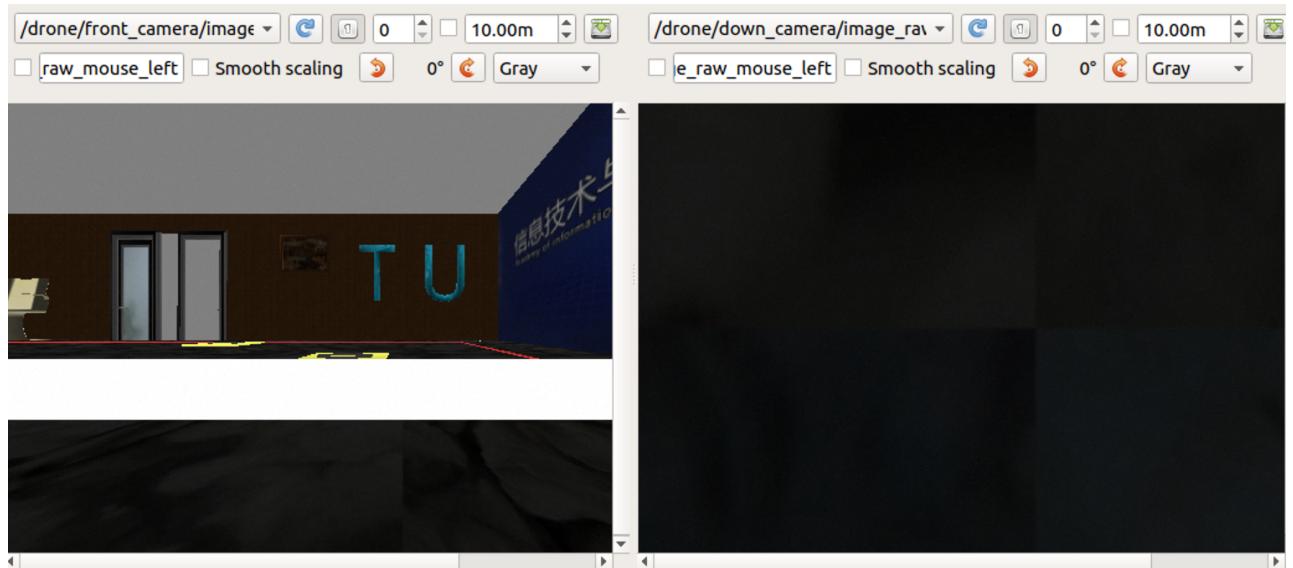
## 5 시뮬레이션 시작



## 6 rostopic list

```
pw@ros:~/catkin_ws$ rostopic list
/camera_info
/clock
/cmd_vel
/drone/down_camera/image_raw
/drone/down_camera/image_raw/compressed
/drone/down_camera/image_raw/compressed/parameter_descriptions
/drone/down_camera/image_raw/compressed/parameter_updates
/drone/down_camera/image_raw/compressedDepth
/drone/down_camera/image_raw/compressedDepth/parameter_descriptions
/drone/down_camera/image_raw/compressedDepth/parameter_updates
/drone/down_camera/image_raw/theora
/drone/down_camera/image_raw/theora/parameter_descriptions
/drone/down_camera/image_raw/theora/parameter_updates
/drone/front_camera/image_raw
/drone/front_camera/image_raw/compressed
/drone/front_camera/image_raw/compressed/parameter_descriptions
/drone/front_camera/image_raw/compressed/parameter_updates
/drone/front_camera/image_raw/compressedDepth
/drone/front_camera/image_raw/compressedDepth/parameter_descriptions
/drone/front_camera/image_raw/compressedDepth/parameter_updates
/drone/front_camera/image_raw/theora
/drone/front_camera/image_raw/theora/parameter_descriptions
/drone/front_camera/image_raw/theora/parameter_updates
/drone/gt_acc
/drone/gt_pose
/drone/gt_vel
/drone/imu
/drone/land
/drone/posctrl
/drone/reset
/drone/sonar
/drone/takeoff
/drone/vel_mode
/gazebo/link_states
/gazebo/model_states
/gazebo/parameter_descriptions
/gazebo/parameter_updates
/gazebo/set_link_state
/gazebo/set_model_state
/rosout
/rosout_agg
```

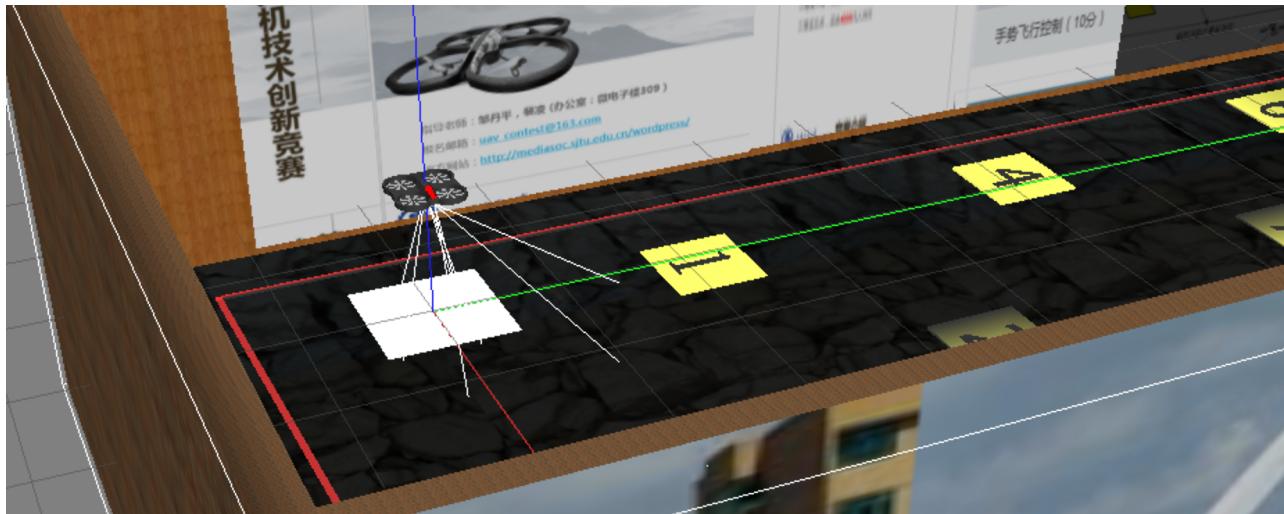
## 7 rqt로 전면, 배면 카메라 관찰



## 8 | 이륙

```
bow@ros:~/catkin_ws$  
bow@ros:~/catkin_ws$  
bow@ros:~/catkin_ws$ rostopic pub -1 /drone/takeoff std_msgs/Empty  
publishing and latching message for 3.0 seconds  
bow@ros:~/catkin_ws$
```

- rostopic pub -1 /drone/takeoff std\_msgs/Empty



## 9 teleop\_twist\_keyboard로 조종이 가능하다 (어차피 cmd\_vel을 사용하니까)

```
pw@ros:~$ rosrun teleop_twist_keyboard teleop_twist_keyboard.py
Reading from the keyboard and Publishing to Twist!
-----
Moving around:
 u   i   o
 j   k   l
 m   ,   .
For Holonomic mode (strafing), hold down the shift key:
-----
 U   I   O
 J   K   L
 M   <   >
+ - up (z)
```

## 10 land

```
pw@ros:~$  
pw@ros:~$ rostopic pub -1 /drone/land std_msgs/Empty "{}"  
publishing and latching message for 3.0 seconds  
pw@ros:~$  
pw@ros:~$  
pw@ros:~$  
pw@ros:~$  
^C
```

- 착륙인지 그냥 드론을 던진건지 구분가지 않아도 놀래지 말자

## 11 world 폴더의 garching\_kitchen.world에서 오류 수정

```
72     |     </mesh>
73     |     </geometry>
74     |     </collision>
75     |     <visual name="garching_kitchen">
76     |     |     <geometry>
77     |     |     |     <mesh>
78     |     |     |     |     <uri>file:///meshes/garching_kitchen.dae</uri>
79     |     |     |     |     <scale>2 2 2</scale>
80     |     |     |     </mesh>
81     |     |     </geometry>
82     |     |     <cast_shadows>false</cast_shadows>
83     |     |     </visual>
84     |     |     </link>
85     |     |     <static>true</static>
86     |     |     </model>
87     |     |     <include>
88     |     |     |     <uri>model://sjtu_drone</uri>
89     |     |     |     <pose>0 0 1 0 0 0</pose>
90     |     |     </include>
91     |     </world>
92     </sdf>
93
```

## 12 simple.launch 수정

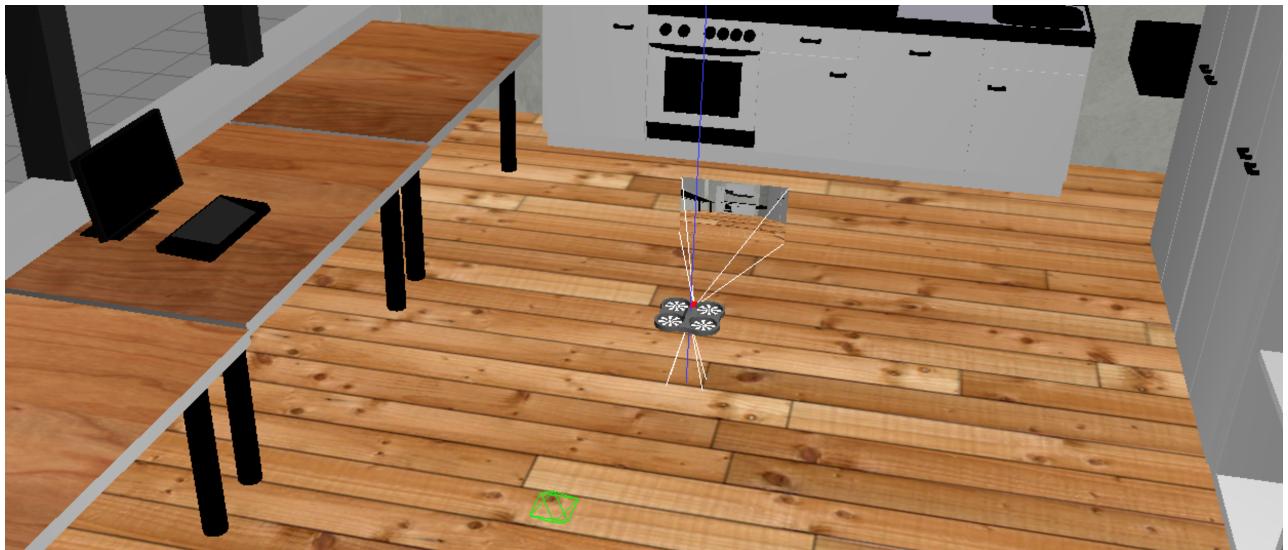


```

1 <launch>
2 <!-- these are the arguments you can pass this launch file, for ex-
3 paused:=true -->
4 <arg name="paused" value="false"/>
5 <arg name="use_sim_time" value="true"/>
6 <arg name="gui" default="false"/>
7 <arg name="headless" value="false"/>
8 <arg name="debug" value="false"/>
9 <arg name="world_name" default="$(find sjtu_drone)/worlds/
10 garching_kitchen.world"/>
11 <!-- launch the custom world -->
12 <include file="$(find gazebo_ros)/launch/empty_world.launch" >
13   <arg name="paused" value="True"/>
14   <arg name="world_name" value="$(arg world_name)"/>
15   <arg name="verbose" value="True"/>
16 </include>
17 </launch>
18

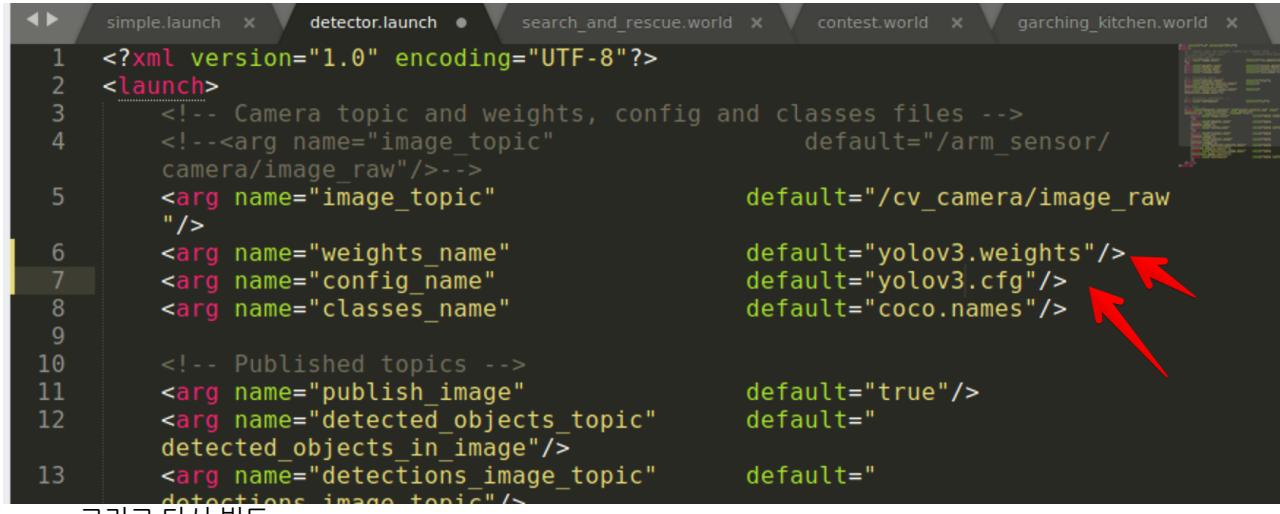
```

- 입력인자를 주는 것이 낫겠지만 그냥~



## 13 YOLO ros wrapper 적용

- weight도 models 폴더안에~
- launch 파일 수정



```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <launch>
3   <!-- Camera topic and weights, config and classes files -->
4   <!<arg name="image_topic"
5     camera/image_raw"/>>-->
6   <arg name="image_topic"
7     "/>
8   <arg name="weights_name"
9     <arg name="config_name"
10    <arg name="classes_name"
11      <!-- Published topics -->
12      <arg name="publish_image"
13        <arg name="detected_objects_topic"
          detected_objects_in_image"/>
          <arg name="detections_image_topic"
            detections_image_topic"/>

```

• 그리고 다시 빌드~

## 14 다시 드론 실행

```
pw@ros:~/catkin_ws$ roslaunch sjtu_drone simple.launch
... logging to /home/pw/.ros/log/82500e48-47cb-11ea-ac52-001c42dcd403/roslaunch-ros-14028.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://ros:33803/

SUMMARY
=====
PARAMETERS
  * /gazebo/enable_ros_network: True
  * /rosdistro: melodic
    • 이륙 : rostopic pub -1 /drone/takeoff std_msgs/Empty
```

## 15 yolo 실행

```
pw@ros:~$ roslaunch yolov3_pytorch_ros detector.launch image_topic:=/drone/front_camera/image_raw
... logging to /home/pw/.ros/log/82500e48-47cb-11ea-ac52-001c42dcd403/roslaunch-ros-15071.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.
```

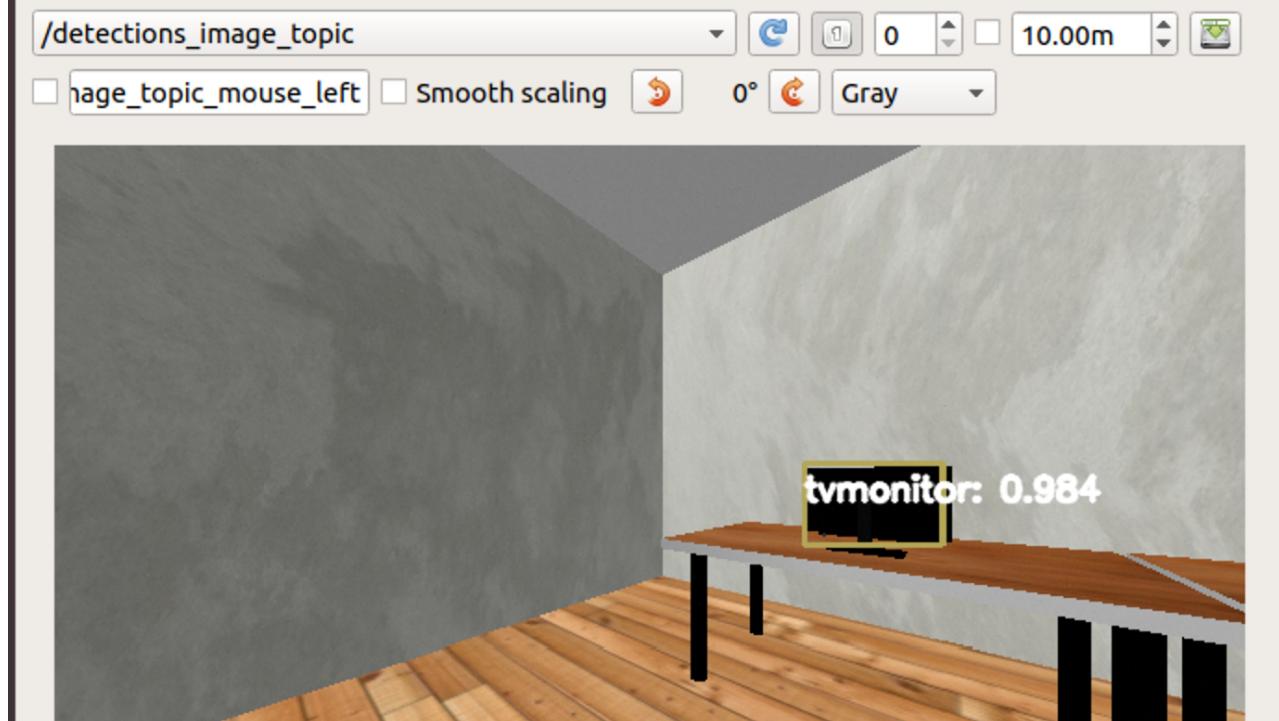
```
started roslaunch server http://ros:33829/
```

```
SUMMARY
```

```
=====
```

```
PARAMETERS
```

```
* /detector_manager/classes_name: coco_names
```



## 16 키보드로 조종

```
pw@ros:~$ rosrun teleop_twist_keyboard teleop_twist_keyboard.py
Reading from the keyboard and Publishing to Twist!
-----
Moving around:
 u    i    o
 j    k    l
 m    ,    .
For Holonomic mode (strafing), hold down the shift key:
-----
 U    I    O
 J    K    L
 M    <    >
t : up (+z)
```

