



ROS, ROBOT,
자율 주행

센서모션로봇공학 실습

수업

이 재 민

010-8992-2675

xoz1206@naver.com

Visionlab (형남공학관 431호)

과제

방 혜 원

sensorrobot1@gmail.com

Visionlab (형남공학관 431호)

강의 내용

JetSon Board 실습

- 01 >> ROS 설치
- 02 >> 카메라 연결
- 03 >> YOLO 설치
- 04 >> 구동
- 05 >> 미션



ROS-YOLO 실습



ROS – YOLO 실습

YOLO

Jetson board에 ROS를 설치하고 USB Camera를 연결해 image Topic을 발행.

Image Topic을 darknet_yolo 딥러닝 ROS package를 통해 이미지 상에 존재하는 객체들을 인식.



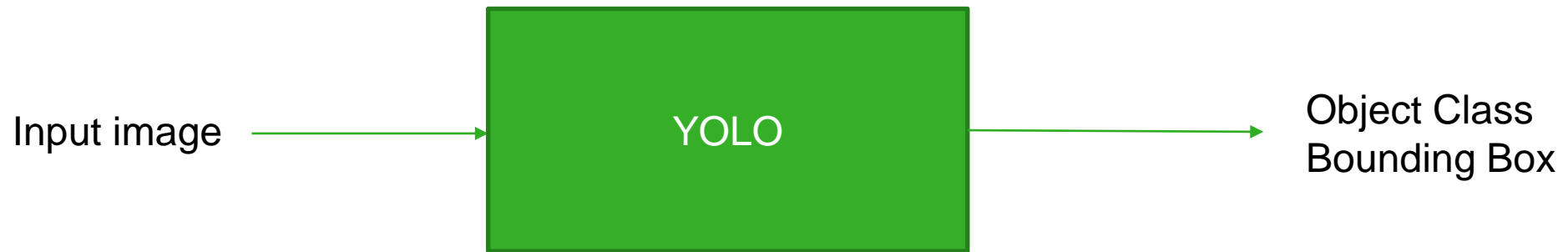
목표 : ROS 기반 이미지 인식

YOLO 란?

YOLO (You Only Look Once) 는 Object detection 분야에서 많이 알려진 모델.

YOLO v1, v2, v3, v4, v5 까지 등장.

빠른 인식이 장점. 정확도가 높지는 않지만, 준수한 성능으로 많이 사용됨.



<https://www.youtube.com/watch?v= JzOFWx1vZg> : YOLO 동영상

<https://ctkim.tistory.com/91> : YOLO 설명 사이트

환경 구축

Terminator 설치

```
sudo apt-get install terminator
```

ROS 설치

<https://doljokilab.tistory.com/1>

Usb cam package 다운로드

```
cd catkin_ws/src
```

```
git clone https://github.com/bosch-ros-pkg/usb\_cam.git
```

```
cd ..
```

```
Catkin_make
```

환경 구축

YOLO package 다운로드

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

```
git clone --recursive https://github.com/leggedrobotics/darknet\_ros.git
```

```
catkin_make
```

“Opencv 패키지를 못 찾는다 “ 에러 발생 시

```
sudo gedit /opt/ros/melodic/share/cv_bridge/cmake/cv_bridgeConfig.cmake
```

```
if(NOT "include:/usr/include:/usr/include/opencv" STREQUAL " ")  
  set(cv_bridge_INCLUDE_DIRS "")  
  set(_include_dirs "include:/usr/include:/usr/include/opencv")
```

opencv4

<https://m.blog.naver.com/uoonm1/221356335476>

환경 구축

홈페이지에서 아래 부분 진행

<https://ropiens.tistory.com/67>

- 빌드에 성공했다면, yolov3-tiny와 관련된 파일을 다운받아야 합니다.
- (1) yolov3-tiny.cfg 다운 : raw.githubusercontent.com/AlexeyAB/darknet/master/cfg/yolov3-tiny.cfg
- (2) yolov3-tiny.weight 파일 다운 : pjreddie.com/media/files/yolov3-tiny.weights
- 다운을 받은 후, 각 cfg, weight 파일은 ~/workspace/src/darknet_ros/darknet_ros/yolo_network_config/ 내의 cfg, weights 폴더에 각각 넣어줍니다.

-
- 이제 yolov3-tiny를 사용하도록 몇 가지 파일을 수정해야 합니다.
 - (1) ~/workspace/src/darknet_ros/darknet_ros/config/ 위치에서 yolov3-tiny.yaml 파일 생성 :
 - yolov3.yaml 파일을 복사해서 yolov3-tiny.yaml 파일로 이름을 바꾼 후에,
 - config_file과 weight_file의 name을 yolov3-tiny.cfg와 yolov3-tiny.weights로 수정해줍니다.

코드 수정

Usb camera 장치 연결 및 해상도 확인

sudo apt install v4l-utils

ls /dev/video*

v4l2-ctl -d /dev/video[x] --list-formats-ext

usb_cam-test.launch 수정

Ubuntu에서 Camera 데이터를 받아 topic으로 발행해주는 package!

```
src > usb_cam > launch > usb_cam-test.launch
1  <launch>
2  <node name="usb_cam" pkg="usb_cam" type="usb_cam_node" output="screen" >
3    <param name="video_device" value="/dev/video0" />
4    <param name="image_width" value="640" />
5    <param name="image_height" value="480" />
6    <param name="pixel_format" value="yuyv" />
7    <param name="camera_frame_id" value="usb_cam" />
8    <param name="io_method" value="mmap"/>
9    <remap from="/usb_cam/image_raw" to="/image_raw" />
10 </node>
11 <node name="image_view" pkg="image_view" type="image_view" respawn="false" output="screen">
12   <remap from="image" to="/image_raw"/>
13   <param name="autosize" value="true" />
14 </node>
15 </launch>
16
```

```
~/my_work/work/opencv_study>ls /dev/video*
/dev/video0 /dev/video1 /dev/video2 /dev/video3
```

< 장치 확인 >

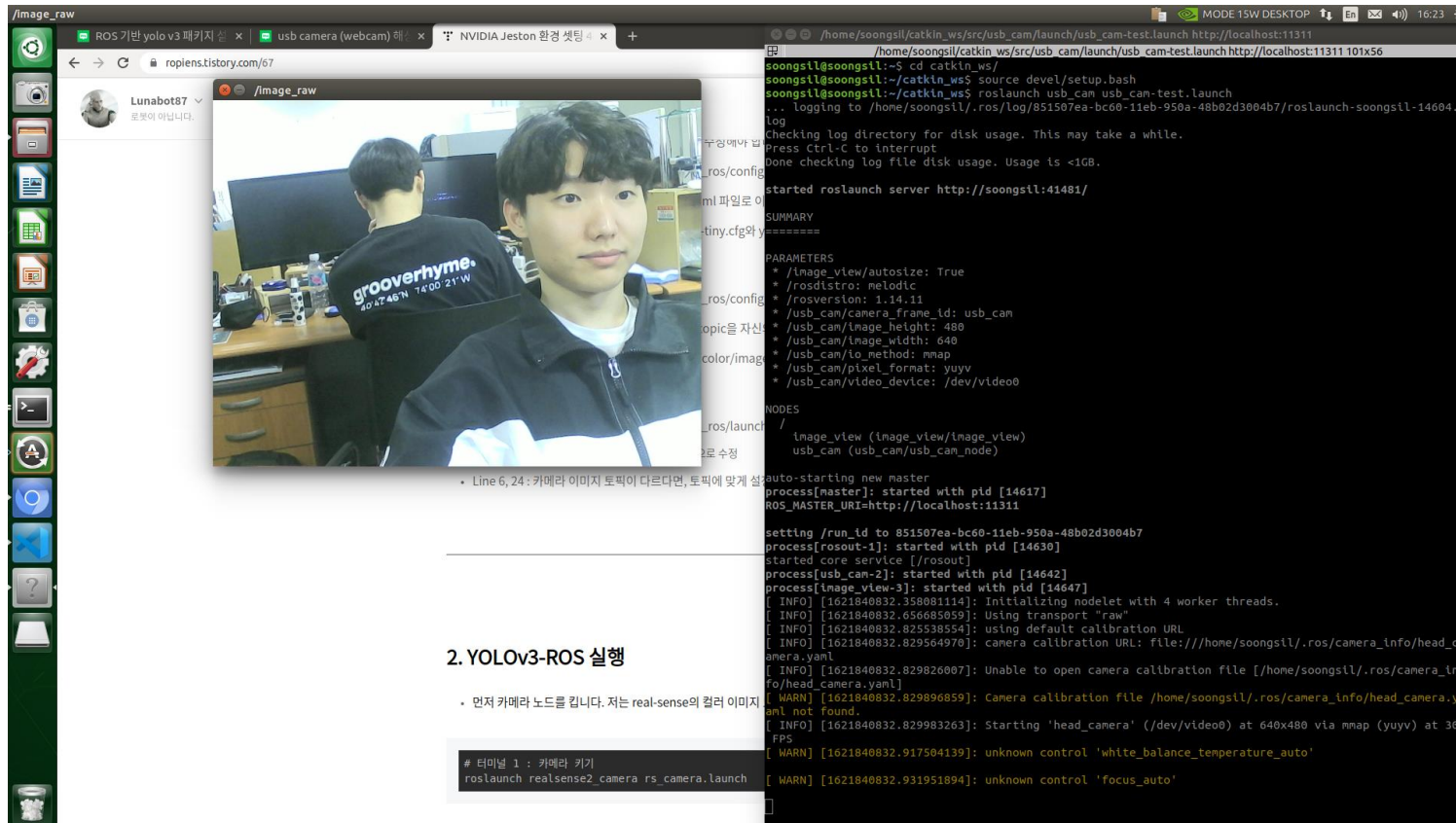
```
~/my_work/work/opencv_study>v4l2-ctl -d /dev/video2 --list-formats-ext
ioctl: VIDIOC_ENUM_FMT
  Index      : 0
  Type       : Video Capture
  Pixel Format: 'YUYV'
  Name       : YUYV 4:2:2
    Size: Discrete 1920x1080
      Interval: Discrete 0.033s (30.000 fps)
      Interval: Discrete 0.067s (15.000 fps)
    Size: Discrete 1280x720
      Interval: Discrete 0.033s (30.000 fps)
      Interval: Discrete 0.067s (15.000 fps)
    Size: Discrete 960x540
      Interval: Discrete 0.033s (30.000 fps)
      Interval: Discrete 0.067s (15.000 fps)
    Size: Discrete 848x480
      Interval: Discrete 0.033s (30.000 fps)
      Interval: Discrete 0.067s (15.000 fps)
    Size: Discrete 640x480
      Interval: Discrete 0.017s (60.000 fps)
      Interval: Discrete 0.033s (30.000 fps)
      Interval: Discrete 0.067s (15.000 fps)
```

< 지원하는 해상도 확인 >

코드 수정

usb_camera 실행

roslaunch usb_cam usb_cam-test.launch



The screenshot shows a ROS environment with a camera feed and terminal output. The camera feed displays a person in a black jacket with 'grooverhyme' text. The terminal output shows the execution of the `roslaunch usb_cam usb_cam-test.launch` command, which starts the `usb_cam` node and the `image_view` node. The terminal output includes the following lines:

```
soongsil@soongsil:~$ cd catkin_ws/
soongsil@soongsil:~/catkin_ws$ source devel/setup.bash
soongsil@soongsil:~/catkin_ws$ roslaunch usb_cam usb_cam-test.launch
... logging to /home/soongsil/.ros/log/851507ea-bc60-11eb-950a-48b02d3004b7/roslaunch-soongsil-14604
log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.
started roslaunch server http://soongsil:41481/
SUMMARY
=====
PARAMETERS
 * /image_view/autosize: True
 * /rostdistro: melodic
 * /rosversion: 1.14.11
 * /usb_cam/camera_frame_id: usb_cam
 * /usb_cam/image_height: 480
 * /usb_cam/image_width: 640
 * /usb_cam/io_method: mmap
 * /usb_cam/pixel_format: yuyv
 * /usb_cam/video_device: /dev/video0
NODES
 /
  image_view (image_view/image_view)
  usb_cam (usb_cam/usb_cam_node)
auto-starting new master
process[master]: started with pid [14617]
ROS_MASTER_URI=http://localhost:11311
setting /run_id to 851507ea-bc60-11eb-950a-48b02d3004b7
process[rosout-1]: started with pid [14630]
started core service [/rosout]
process[usb_cam-2]: started with pid [14642]
process[image_view-3]: started with pid [14647]
[ INFO] [1621840832.358081114]: Initializing nodelet with 4 worker threads.
[ INFO] [1621840832.056085059]: Using transport "raw"
[ INFO] [1621840832.82538954]: using default calibration URL
[ INFO] [1621840832.829564970]: camera calibration URL: file:///home/soongsil/.ros/camera_info/head_camera.yaml
[ INFO] [1621840832.829826007]: Unable to open camera calibration file [/home/soongsil/.ros/camera_info/head_camera.yaml]
[ WARN] [1621840832.829860859]: Camera calibration file /home/soongsil/.ros/camera_info/head_camera.yaml not found.
[ INFO] [1621840832.829983263]: Starting 'head_camera' (/dev/video0) at 640x480 via mmap (yuyv) at 30 FPS
[ WARN] [1621840832.917504139]: unknown control 'white_balance_auto'
[ WARN] [1621840832.931951894]: unknown control 'focus_auto'
```

```
soongsil@soongsil:~/catkin_ws$ rostopic list
/camera_info
/image_raw
/image_raw/compressed
/image_raw/compressed/parameter_descriptions
/image_raw/compressed/parameter_updates
/image_raw/compressedDepth
/image_raw/compressedDepth/parameter_descriptions
/image_raw/compressedDepth/parameter_updates
/image_raw/theora
/image_raw/theora/parameter_descriptions
/image_raw/theora/parameter_updates
/image_view/output
/image_view/parameter_descriptions
/image_view/parameter_updates
/rosout
/rosout_agg
```

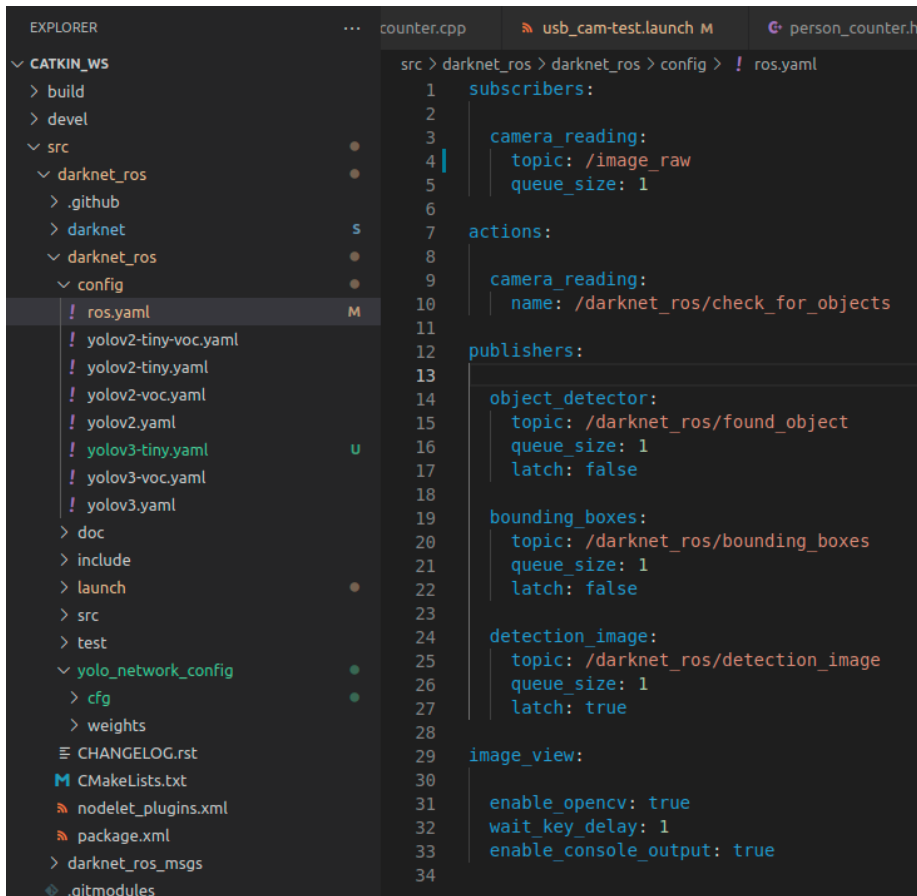
2. YOLOv3-ROS 실행

- 먼저 카메라 노드를 켭니다. 저는 real-sense의 컬러 이미지

```
# 터미널 1 : 카메라 키기
roslaunch realsense2_camera rs_camera.launch
```

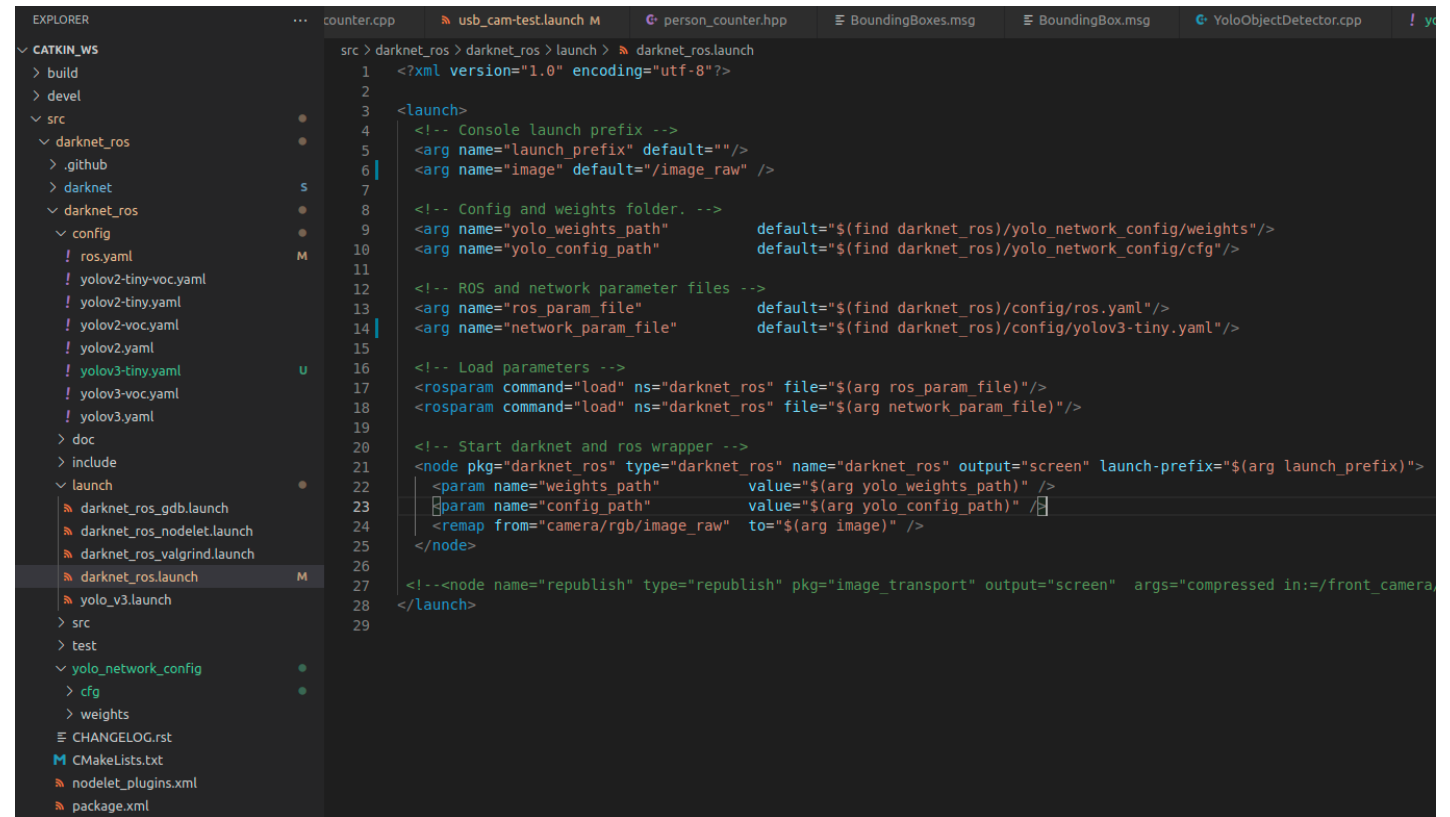
코드 수정

Darknet_ros 수정



The screenshot shows the VS Code interface with the Explorer on the left displaying the file structure of the `CATKIN_WS` workspace. The file `ros.yaml` under `src/darknet_ros/config` is selected. The main editor shows the content of `ros.yaml` with the following structure:

```
1 subscribers:
2
3 camera_reading:
4   topic: /image_raw
5   queue_size: 1
6
7 actions:
8   camera_reading:
9     name: /darknet_ros/check_for_objects
10
11 publishers:
12
13   object_detector:
14     topic: /darknet_ros/found_object
15     queue_size: 1
16     latch: false
17
18   bounding_boxes:
19     topic: /darknet_ros/bounding_boxes
20     queue_size: 1
21     latch: false
22
23   detection_image:
24     topic: /darknet_ros/detection_image
25     queue_size: 1
26     latch: true
27
28 image_view:
29   enable_opencv: true
30   wait_key_delay: 1
31   enable_console_output: true
32
33
```



The screenshot shows the VS Code interface with the Explorer on the left displaying the file structure of the `CATKIN_WS` workspace. The file `darknet_ros.launch` under `src/darknet_ros/launch` is selected. The main editor shows the content of `darknet_ros.launch` with the following structure:

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <launch>
4   <!-- Console launch prefix -->
5   <arg name="launch_prefix" default="" />
6   <arg name="image" default="/image_raw" />
7
8   <!-- Config and weights folder. -->
9   <arg name="yolo_weights_path" default="$(find darknet_ros)/yolo_network_config/weights"/>
10  <arg name="yolo_config_path" default="$(find darknet_ros)/yolo_network_config/cfg"/>
11
12  <!-- ROS and network parameter files -->
13  <arg name="ros_param_file" default="$(find darknet_ros)/config/ros.yaml"/>
14  <arg name="network_param_file" default="$(find darknet_ros)/config/yolov3-tiny.yaml"/>
15
16  <!-- Load parameters -->
17  <rosparam command="load" ns="darknet_ros" file="$(arg ros_param_file)"/>
18  <rosparam command="load" ns="darknet_ros" file="$(arg network_param_file)"/>
19
20  <!-- Start darknet and ros wrapper -->
21  <node pkg="darknet_ros" type="darknet_ros" name="darknet_ros" output="screen" launch-prefix="$(arg launch_prefix)">
22    <param name="weights_path" value="$(arg yolo_weights_path)" />
23    <param name="config_path" value="$(arg yolo_config_path)" />
24    <remap from="camera/rgb/image_raw" to="$(arg image)" />
25  </node>
26
27  <!--<node name="republish" type="republish" pkg="image_transport" output="screen" args="compressed in:=/front_camera
28  </launch>
29
```

Darknet_ros / config / launch / darknet_ros.launch 수정

Darknet_ros / config / ros.yaml file 수정

코드 수정

Darknet_ros msg

```
find_package(catkin REQUIRED
  COMPONENTS
    cv_bridge
    roscpp
    rospy
    std_msgs
    actionlib
    darknet_ros_msgs
    image_transport
    nodelet
)
```

Darknet_ros CMakeLists.txt

```
darknet_ros_msgs
├── action
├── msg
│   ├── BoundingBox.msg
│   ├── BoundingBoxes.msg
│   ├── ObjectCount.msg
│   ├── CHANGELOG.rst
│   ├── CMakeLists.txt
│   └── package.xml
```

```
BoundingBoxes.msg
src > darknet_ros > darknet_ros_msgs > msg > BoundingBoxes.msg
1 Header header
2 Header image_header
3 BoundingBox[] bounding_boxes
4
```

```
BoundingBox.msg
src > darknet_ros > darknet_ros_msgs >
1 float64 probability
2 int64 xmin
3 int64 ymin
4 int64 xmax
5 int64 ymax
6 int16 id
7 string Class
8
```

YOLO에서 사용하는 msg 자료구조! -> darknet_ros의 최종 output

코드 수정

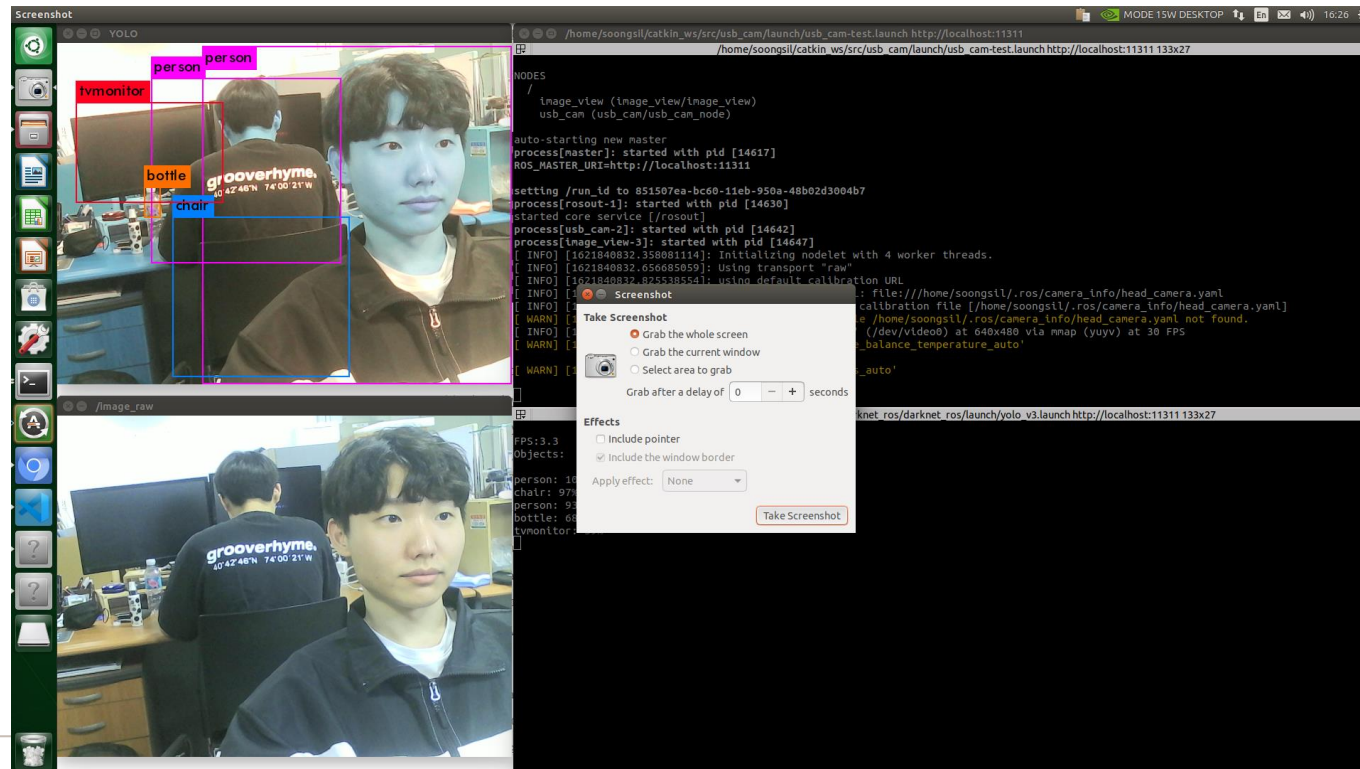
Darknet_ros 실행

roslaunch darknet_ros darknet_ros.launch

tiny version

roslaunch darknet_ros yolo_v3.launch

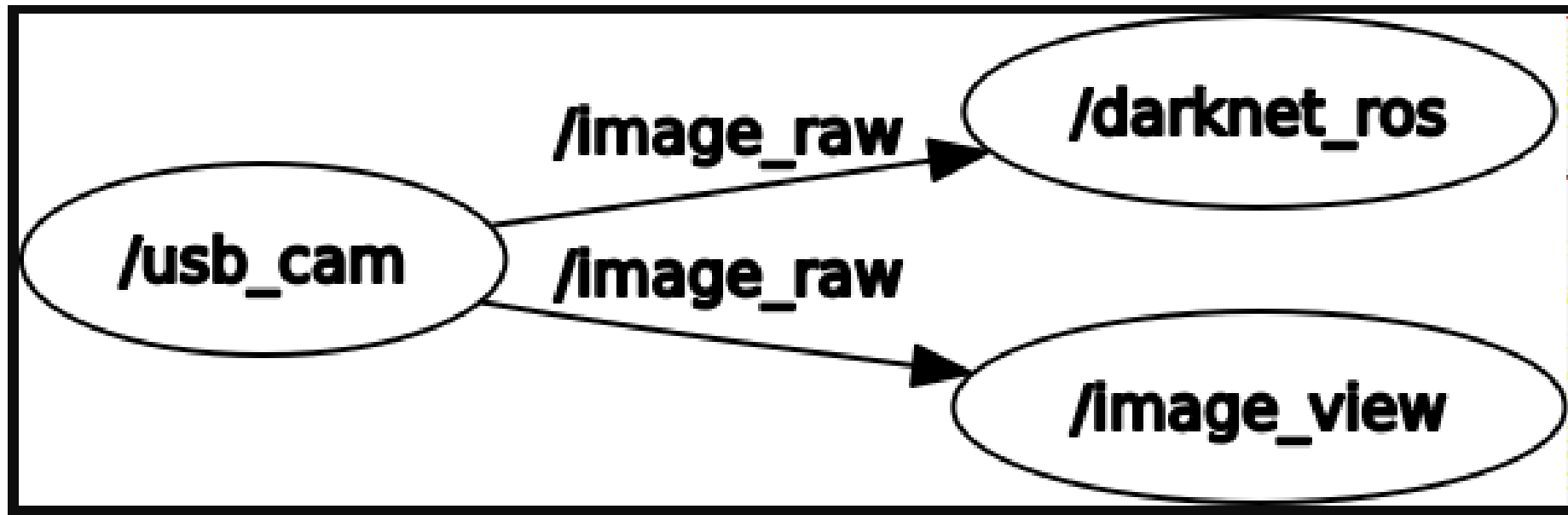
default version



```
soongsil@soongsil:~/catkin_ws$ rostopic list
/camera_info
/darknet_ros/bounding_boxes
/darknet_ros/check_for_objects/cancel
/darknet_ros/check_for_objects/feedback
/darknet_ros/check_for_objects/goal
/darknet_ros/check_for_objects/result
/darknet_ros/check_for_objects/status
/darknet_ros/detection_image
/darknet_ros/found_object
/image_raw
/image_raw/compressed
/image_raw/compressed/parameter_descriptions
/image_raw/compressed/parameter_updates
/image_raw/compressedDepth
/image_raw/compressedDepth/parameter_descriptions
/image_raw/compressedDepth/parameter_updates
/image_raw/theora
/image_raw/theora/parameter_descriptions
/image_raw/theora/parameter_updates
/image_view/output
/image_view/parameter_descriptions
/image_view/parameter_updates
/rosout
/rosout_agg
/statistics
```


코드 수정

rqt_graph



미션

count_person 패키지 작성

- YOLO 결과로 나오는 data를 subscribe.
- 사람(person)의 수를 counting.
- 사람의 수를 Int32 자료형 사용자 정의 msg 로 publish.

요구사항

- Publish Topic Name : /person_num
- Msg 자료형 : int32
- Class 형식으로 작성할 것.

```
FPS:20.3
Objects:

tvmonitor: 66%
tvmonitor: 55%
bottle: 38%
person: 45%
bottle: 35%
bottle: 35%
shutting down processing monitor...
... shutting down processing monitor complete
done
soongsil@soongsil:~/catkin_ws$
```

```
soongsil@soongsil: ~/catkin
[ INFO] [1621852691.458749499]: Person_num : 1
[ INFO] [1621852691.490956622]: Person_num : 1
[ INFO] [1621852691.530755271]: Person_num : 1
[ INFO] [1621852691.576178400]: Person_num : 1
[ INFO] [1621852691.618451288]: Person_num : 1
[ INFO] [1621852691.659899589]: Person_num : 1
[ INFO] [1621852691.711579808]: Person_num : 1
[ INFO] [1621852691.755874399]: Person_num : 1
[ INFO] [1621852691.794496827]: Person_num : 1
[ INFO] [1621852691.840531028]: Person_num : 1
[ INFO] [1621852691.888878275]: Person_num : 1
[ INFO] [1621852691.933226661]: Person_num : 1
```