Docker image setup notes:

Pull docker image from Docker hub:

```
$ sudo docker pull waggle/plugin-tensorflow:2.0.0
```

Launch docker container:

```
$ sudo docker run -it waggle/plugin-tensorflow:2.0.0 /bin/bash
```

Install lsb_release for bash:

```
# apt-get update && apt-get install -y lsb-release && apt-get clean all
```

Install ROS melodic, refer to melodic/Installation/Ubuntu - ROS Wiki
Note: Choose ros-melodic-ros-base

Set up ROS workspace refer to ROS/Tutorials/InstallingandConfiguringROSEnvironment - ROS Wiki
Note: choose Python3 as executor and put carkin_ws/devel/setup.bash into .bashrc
Here python3 needs several packages installed:

```
# pip3 install catkin_pkg
```

Get video_stream_opencv package into ROS workspace:

```
# git clone https://github.com/ros-drivers/video_stream_opencv.git
~/catkin_ws# rosdep install video_stream_opencv
# pip3 install rospkg
# catkin_make -DPYTHON_EXECUTABLE=/usr/bin/python3
```

Rosdep install is equivalent to:

```
# apt install ros-melodic-cv-bridge
# apt install ros-melodic-image-transport
# apt install ros-melodic-camera-info-manager
```

Set up darknet object detection demo in ROS:

```
# git clone --recursive https://github.com/leggedrobotics/darknet_ros.git
# pip install empy
```

Then build the darnet ros package:

```
# catkin_make -DPYTHON_EXECUTABLE=/usr/bin/python3 -DCMAKE_BUILD_TYPE=Release
```

Download weights:

```
cd catkin_ws/src/darknet_ros/darknet_ros/yolo_network_config/weights/
apt install wget
wget http://pjreddie.com/media/files/yolov3.weights
wget http://pjreddie.com/media/files/yolov3-tiny.weights
```

Set up Tensorflow object detection with ROS:
Ref: https://github.com/osrf/tensorflow_object_detector

```
sudo apt-get install python3-opencv
sudo apt-get install ros-melodic-image-view
pip3 install Pillow

# Allow containers to communicate with Xorg
$ sudo xhost +si:localuser:root
$ sudo docker run --runtime nvidia --network host -it -e DISPLAY=$DISPLAY -v
/tmp/.X11-unix/:/tmp/.X11-unix --device=/dev/video0:/dev/video1
liangkailiu/plugin-tensorflow-ros:v2
```

TF2 and TF1 syntax:
1. od_graph_def = tf.GraphDef() AttributeError: module 'tensorflow' has no attribute 'GraphDef'
2. module 'tensorflow' has no attribute 'ConfigProto' · Issue #33504 · tensorflow/tensorflow
3. AttributeError: module 'tensorflow' has no attribute 'Session' · Issue #18538 · tensorflow/tensorflow

CUDA cmake issue:

```
[43%] Linking CXX shared library
/root/catkin_ws/devel/lib/libdarknet_ros_lib.so
/usr/bin/ld: cannot find -lcuda
collect2: error: ld returned 1 exit status
darknet_ros/darknet_ros/CMakeFiles/darknet_ros_lib.dir/build.make:3733: recipe
for target '/root/catkin_ws/devel/lib/libdarknet_ros_lib.so' failed
make[2]: *** [/root/catkin_ws/devel/lib/libdarknet_ros_lib.so] Error 1
CMakeFiles/Makefile2:2708: recipe for target
'darknet_ros/darknet_ros/CMakeFiles/darknet_ros_lib.dir/all' failed
make[1]: *** [darknet_ros/darknet_ros/CMakeFiles/darknet_ros_lib.dir/all] Error
2
Makefile:140: recipe for target 'all' failed
make: *** [all] Error 2
Invoking "make -j4 -l4" failed
```

Solved. Refer to https://github.com/apache/incubator-mxnet/issues/8436
Find and link *libcuda.so.1* to */usr/lib/aarch64-linux-gnu/libcuda.so*

```
sudo ln -s /usr/local/lib/libcuda.so.1 /usr/lib/aarch64-linux-gnu/libcuda.so
```

OpenCV4 Issue with ROS:

```
nvidia@nvidia-desktop:~$ sudo docker run -it plugin-tensorflow-ros /bin/bash
[sudo] password for nvidia:
root@e2e7604628c5:/# roslaunch darknet_ros
```

```
darknet_ros.launch      object_detection.test
darknet_ros_gdb.launch  yolo_v3.launch
root@e2e7604628c5:/# roslaunch darknet_ros
darknet_ros.launch      object_detection.test
darknet_ros_gdb.launch  yolo_v3.launch
root@e2e7604628c5:/# roslaunch darknet_ros darknet_ros.launch
... logging to
/root/.ros/log/5ab37520-9c47-11ea-a3ab-0242ac110002/roslaunch-e2e7604628c5-211.
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://e2e7604628c5:35317/

SUMMARY
========

PARAMETERS
 * /darknet_ros/actions/camera_reading/name: /darknet_ros/chec...
 * /darknet_ros/config_path: /root/catkin_ws/s...
 * /darknet_ros/image_view/enable_console_output: True
 * /darknet_ros/image_view/enable_opencv: True
 * /darknet_ros/image_view/wait_key_delay: 1
 * /darknet_ros/publishers/bounding_boxes/latch: False
 * /darknet_ros/publishers/bounding_boxes/queue_size: 1
 * /darknet_ros/publishers/bounding_boxes/topic: /darknet_ros/boun...
 * /darknet_ros/publishers/detection_image/latch: True
 * /darknet_ros/publishers/detection_image/queue_size: 1
 * /darknet_ros/publishers/detection_image/topic: /darknet_ros/dete...
 * /darknet_ros/publishers/object_detector/latch: False
 * /darknet_ros/publishers/object_detector/queue_size: 1
 * /darknet_ros/publishers/object_detector/topic: /darknet_ros/foun...
 * /darknet_ros/subscribers/camera_reading/queue_size: 1
 * /darknet_ros/subscribers/camera_reading/topic: /camera/rgb/image...
 * /darknet_ros/weights_path: /root/catkin_ws/s...
 * /darknet_ros/yolo_model/config_file/name: yolov2-tiny.cfg
 * /darknet_ros/yolo_model/detection_classes/names: ['person', 'bicyc...
 * /darknet_ros/yolo_model/threshold/value: 0.3
 * /darknet_ros/yolo_model/weight_file/name: yolov2-tiny.weights
 * /rosdistro: melodic
 * /rosversion: 1.14.5

NODES
  /
    darknet_ros (darknet_ros/darknet_ros)

auto-starting new master
```

```
process[master]: started with pid [221]
ROS_MASTER_URI=http://localhost:11311

setting /run_id to 5ab37520-9c47-11ea-a3ab-0242ac110002
process[rosout-1]: started with pid [232]
started core service [/rosout]
process[darknet_ros-2]: started with pid [238]
[ INFO] [1590164093.347287816]: [YoloObjectDetector] Node started.
[ INFO] [1590164093.363454414]: [YoloObjectDetector] Xserver is not running.
[ INFO] [1590164093.375057429]: [YoloObjectDetector] init().
YOLO V3
layer     filters    size              input                output
    0 CUDA Error: CUDA driver version is insufficient for CUDA runtime version
CUDA Error: CUDA driver version is insufficient for CUDA runtime version:
Resource temporarily unavailable
[darknet_ros-2] process has died [pid 238, exit code 255, cmd
/root/catkin_ws/devel/lib/darknet_ros/darknet_ros
camera/rgb/image_raw:=/camera/rgb/image_raw __name:=darknet_ros
__log:=/root/.ros/log/5ab37520-9c47-11ea-a3ab-0242ac110002/darknet_ros-2.log].
log file:
/root/.ros/log/5ab37520-9c47-11ea-a3ab-0242ac110002/darknet_ros-2*.log
^C[rosout-1] killing on exit
[master] killing on exit
shutting down processing monitor...
... shutting down processing monitor complete
```

Nvidia Jetson docker link issue:

```
nvidia@nvidia-desktop:/usr/local/cuda-10.2/targets/aarch64-linux/lib$ sudo
nvidia-container-cli -k -d /dev/tty info

-- WARNING, the following logs are for debugging purposes only --

I0522 15:58:24.937181 20852 nvc.c:281] initializing library context
(version=1.1.1, build=e5d6156aba457559979597c8e3d22c5d8d0622db)
I0522 15:58:24.937470 20852 nvc.c:255] using root /
I0522 15:58:24.937503 20852 nvc.c:256] using ldcache /etc/ld.so.cache
I0522 15:58:24.937540 20852 nvc.c:257] using unprivileged user 65534:65534
W0522 15:58:24.938437 20852 nvc.c:171] failed to detect NVIDIA devices
I0522 15:58:24.939162 20853 nvc.c:191] loading kernel module nvidia
E0522 15:58:24.940468 20853 nvc.c:193] could not load kernel module nvidia
I0522 15:58:24.940512 20853 nvc.c:203] loading kernel module nvidia_uvm
E0522 15:58:24.941286 20853 nvc.c:205] could not load kernel module nvidia_uvm
I0522 15:58:24.941325 20853 nvc.c:211] loading kernel module nvidia_modeset
E0522 15:58:24.942163 20853 nvc.c:213] could not load kernel module
nvidia_modeset
I0522 15:58:24.943107 20854 driver.c:101] starting driver service
```

```
E0522 15:58:24.943938 20854 driver.c:161] could not start driver service: load
library failed: libnvidia-ml.so.1: cannot open shared object file: no such file
or directory
I0522 15:58:24.944365 20852 driver.c:196] driver service terminated
successfully
nvidia-container-cli: initialization error: driver error: failed to process
request
```

Solved. After reflashing the whole system on Jetson AGX board, the issue is solved.

ROS cv_bridge issue with OpenCV4:

```
nvidia@nvidia-desktop:~/projects/catkin_ws$ catkin_make
-DPYTHON_EXECUTABLE=/usr/bin/python3
Base path: /home/nvidia/projects/catkin_ws
Source space: /home/nvidia/projects/catkin_ws/src
Build space: /home/nvidia/projects/catkin_ws/build
Devel space: /home/nvidia/projects/catkin_ws/devel
Install space: /home/nvidia/projects/catkin_ws/install
####
#### Running command: "make cmake_check_build_system" in
"/home/nvidia/projects/catkin_ws/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/nvidia/projects/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH:
/home/nvidia/projects/catkin_ws/devel;/opt/ros/melodic
-- This workspace overlays:
/home/nvidia/projects/catkin_ws/devel;/opt/ros/melodic
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.6.9",
minimum required is "2")
-- Using PYTHON_EXECUTABLE: /usr/bin/python3
-- Using Debian Python package layout
-- Using empy: /usr/bin/empy
-- Using CATKIN_ENABLE_TESTING: ON
-- Call enable_testing()
-- Using CATKIN_TEST_RESULTS_DIR:
/home/nvidia/projects/catkin_ws/build/test_results
-- Found gtest sources under '/usr/src/googletest': gtests will be built
-- Found gmock sources under '/usr/src/googletest': gmock will be built
-- Found PythonInterp: /usr/bin/python3 (found version "3.6.9")
-- Using Python nosetests: /usr/bin/nosetests
-- catkin 0.7.23
-- BUILD_SHARED_LIBS is on
-- BUILD_SHARED_LIBS is on
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- ~~  traversing 1 packages in topological order:
```

```
-- ~~  - video_stream_opencv
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- +++ processing catkin package: 'video_stream_opencv'
-- ==> add_subdirectory(video_stream_opencv)
CMake Error at /opt/ros/melodic/share/cv_bridge/cmake/cv_bridgeConfig.cmake:113
(message):
  Project 'cv_bridge' specifies '/usr/include/opencv' as an include dir,
  which is not found.  It does neither exist as an absolute directory nor in
  '${{prefix}}//usr/include/opencv'.  Check the issue tracker
  'https://github.com/ros-perception/vision_opencv/issues' and consider
  creating a ticket if the problem has not been reported yet.
Call Stack (most recent call first):
  /opt/ros/melodic/share/catkin/cmake/catkinConfig.cmake:76 (find_package)
  video_stream_opencv/CMakeLists.txt:5 (find_package)


-- Configuring incomplete, errors occurred!
See also "/home/nvidia/projects/catkin_ws/build/CMakeFiles/CMakeOutput.log".
See also "/home/nvidia/projects/catkin_ws/build/CMakeFiles/CMakeError.log".
Makefile:320: recipe for target 'cmake_check_build_system' failed
make: *** [cmake_check_build_system] Error 1
Invoking "make cmake_check_build_system" failed
```

Solved. Solution:

```
nvidia@nvidia-desktop:~/projects/catkin_ws$ roscd cv_bridge/
nvidia@nvidia-desktop:/opt/ros/melodic/share/cv_bridge$ cd cmake/
nvidia@nvidia-desktop:/opt/ros/melodic/share/cv_bridge/cmake$ sudo vim
cv_bridgeConfig.cmake
```

In cv_bridgeConfig.cmake, change “/usr/include/opencv” to “/usr/include/opencv4”:

```
set(_include_dirs "include;/usr/include;/usr/include/opencv4")
```

cv_bridge issue with python3:

```
[ERROR] [1590549315.289397]: bad callback: <bound method Detector.image_cb of
<__main__.Detector object at 0x7f71147128>>
Traceback (most recent call last):
  File "/opt/ros/melodic/lib/python2.7/dist-packages/rospy/topics.py", line
750, in _invoke_callback
    cb(msg)
  File "/root/catkin_ws/src/tensorflow_object_detector/scripts/detect_ros.py",
line 80, in image_cb
    cv_image = self.bridge.imgmsg_to_cv2(data, "bgr8")
  File "/opt/ros/melodic/lib/python2.7/dist-packages/cv_bridge/core.py", line
163, in imgmsg_to_cv2
    dtype, n_channels = self.encoding_to_dtype_with_channels(img_msg.encoding)
```

```
  File "/opt/ros/melodic/lib/python2.7/dist-packages/cv_bridge/core.py", line
99, in encoding_to_dtype_with_channels
    return
self.cvtype2_to_dtype_with_channels(self.encoding_to_cvtype2(encoding))
  File "/opt/ros/melodic/lib/python2.7/dist-packages/cv_bridge/core.py", line
91, in encoding_to_cvtype2
    from cv_bridge.boost.cv_bridge_boost import getCvType
ImportError: dynamic module does not define module export function
(PyInit_cv_bridge_boost)
```

Ref: Unable to use cv_bridge with ROS Kinetic and Python3

NVIDIA Container Runtime on Jetson:

Ref: https://github.com/NVIDIA/nvidia-docker/wiki/NVIDIA-Container-Runtime-on-Jetson
Requires JetPack 4.2.1+