MRT Assignment

Real-Time Edge Detection using ROS & OpenCV

Saumya Dharmesh Shah

22B1238

Electrical Engineering, IIT Bombay

Summary:

In this task, I learned how to use ROS and OpenCV to develop a Python publisher node and a Python subscriber node for image processing. Also, I learnt how to convert between ROS and OpenCV picture formats using the `cv_bridge` library.

The publisher node publishes the frames it has captured from the default camera to the topic '`frames`'. The subscriber node processes each frame by applying Gaussian blur(using `cv2.GaussianBlur()`) and Canny edge detection(using `cv2.Canny()`) while subscribing to the '`frames`' topic. A single window is then used to show both the original and edge pictures that were produced.

To test my implementation, I created a launch file that will launch the publisher and subscriber simultaneously; then, I ran the roslaunch command. I also used the `rqt_graph` command to visualize the communication between nodes and topics.

Workflow:

I begin with opening the terminal, there I run `roslaunch mrt_assignment edge_detect.launch`, which executes the launch file '`edge_detect.launch`', which in turn launches the `publisher.py` and `subscriber.py` files which initialises publisher and subscriber nodes which communicate on a topic '`frames`'.

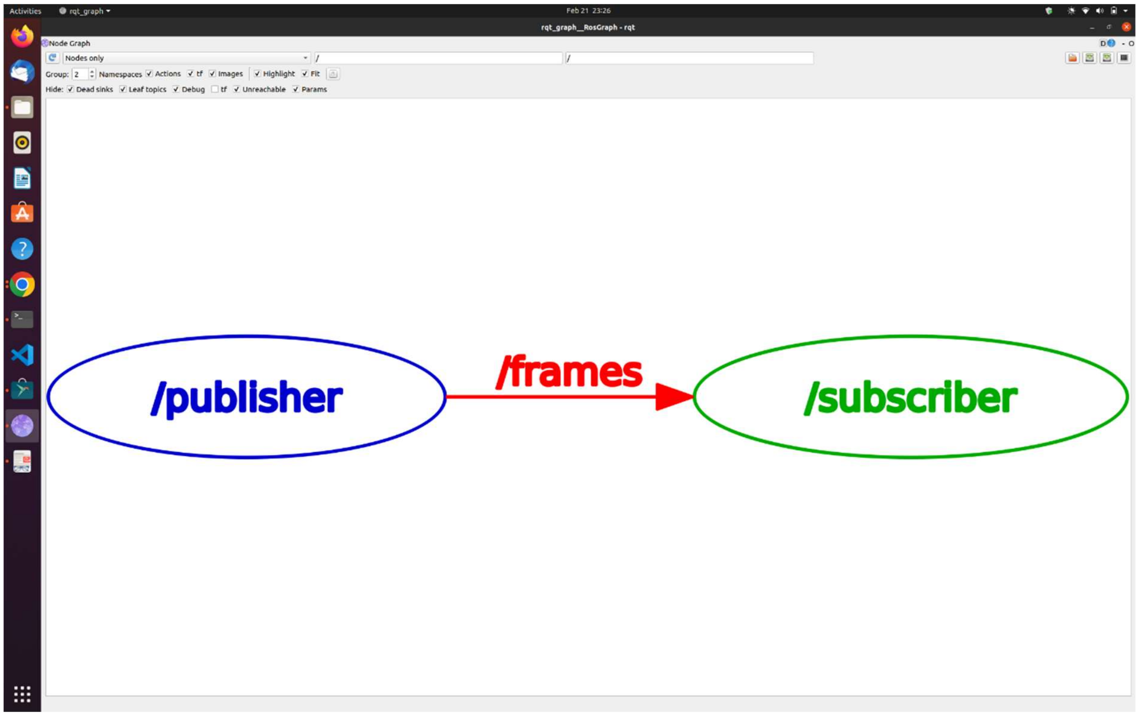
In the program, `cv_bridge` converts ROS message to OpenCV message and vice-versa.

I faced many bugs during the execution of my program.

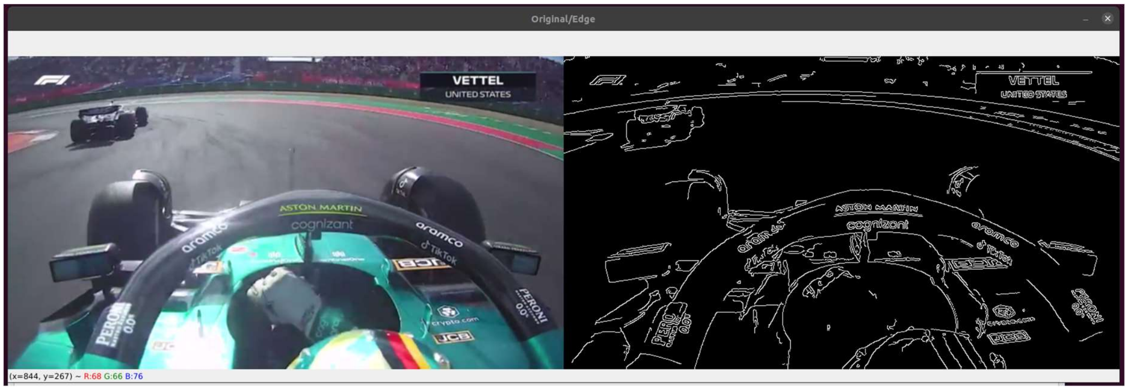
- I was using just the name of the video instead of the whole path, so it was showing a black screen in the window
- My subscriber wasn't working correctly, which was because I hadn't used the `cv2.waitKey()` function in my processer
- I was getting an error in the `np.hstack` line of my subscriber code, which was due to the `edge_img` being grayscale which is a 2D array, but the original image was a 3D array, so I used `cv2.cvtColor(edge_img,cv2.COLOR_GRAY2BGR)` which converts the image from grayscale to BGR(i.e. from 2D to 3D array)
- I kept on forgetting to add dependencies to my package `mrt_assignment`. After surfing the internet for a few hours, I returned to the ROS Tutorials, where I realised I had to change the `CMakeLists.txt` and `package.xml` files.

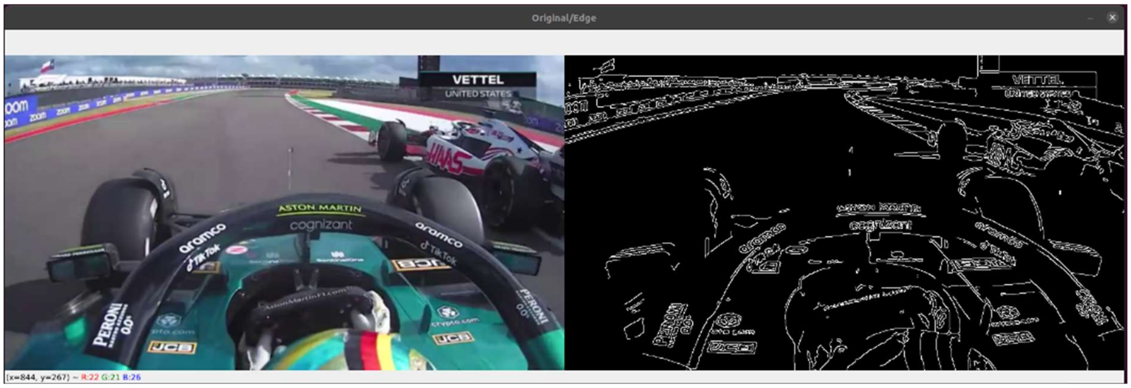
rqt_graph image:

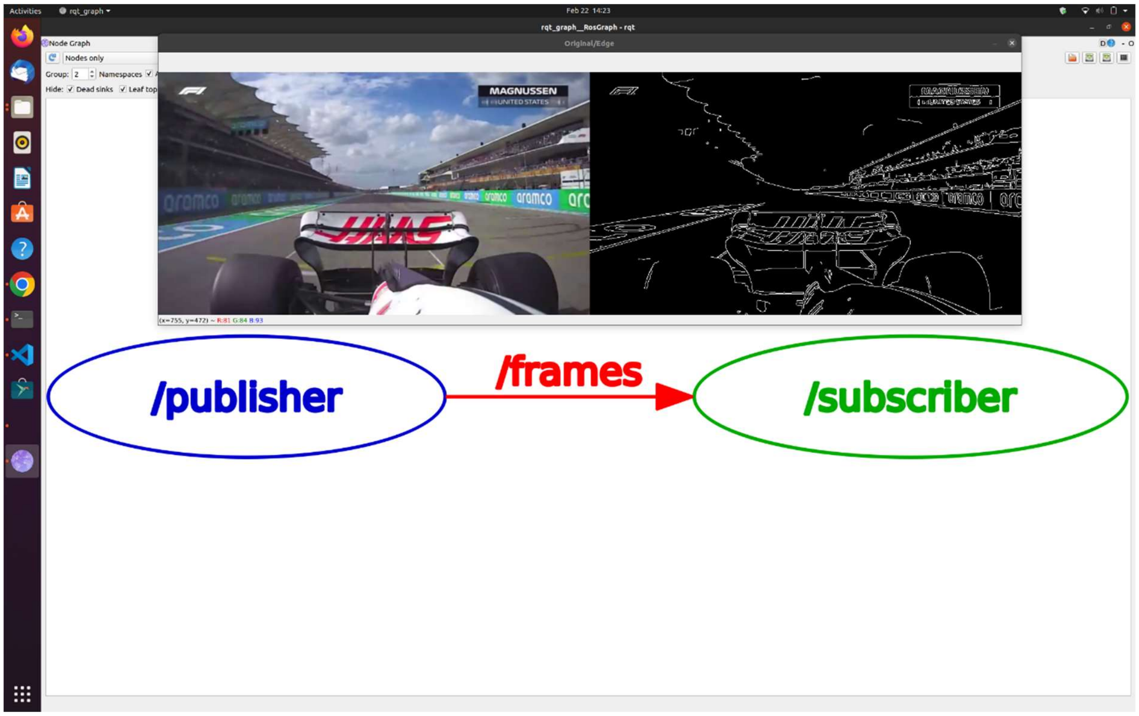
(Two nodes *publisher* & *subscriber* communicating through the topic *frames*)



Output images:


Original/Edge
VETTEL
UNITED STATES
(x=844, y=267) ~ R:68 G:66 B:76


Original/Edge
VETTEL
UNITED STATES
(x=844, y=267) ~ R:22 G:21 B:26


Activities
rqt_graph
Feb 22 14:23
rqt_graph__RosGraph - rqt
Node Graph
Nodes only
Group: 2
Namespaces
Hide: Dead sinks
Leaf top
Original/Edge
MAGNUSSEN
UNITED STATES
(x=755, y=472) ~ R:81 G:84 B:93
/publisher
/frames
/subscriber