Depth Distance Detection

Library Used

- *cv2*
- *pyrealsense2*

Python Scripts

Before encapsulating into a ROS 2 node, a few Python scripts were created to experiment with the RealSense depth camera and test the feasibility of depth information detection.

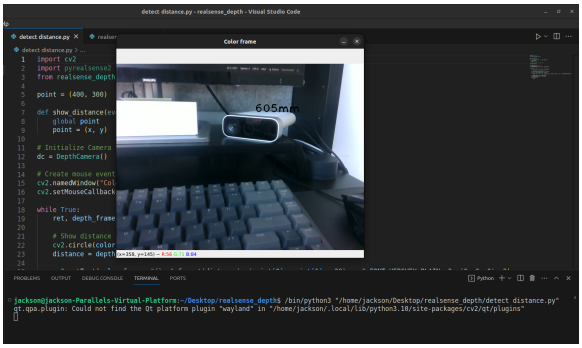
By utilising the features provided by pyrealsense2, the connection to the RealSense depth camera was successfully established.

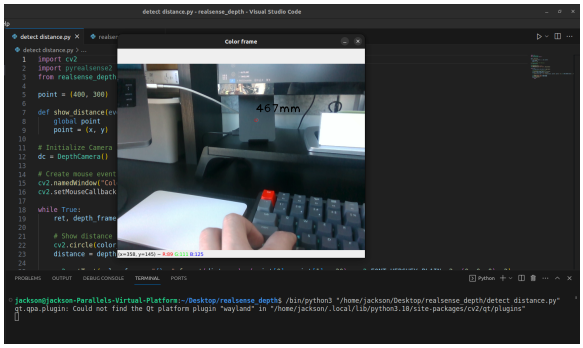
A few things we can achieve by now:

- Establish a connection to the camera
- Enable video streaming at **30 FPS**
- Get **colour** frames (streaming)
- Get **depth** frames (streaming)
- Extract the depth information at a given pixel position

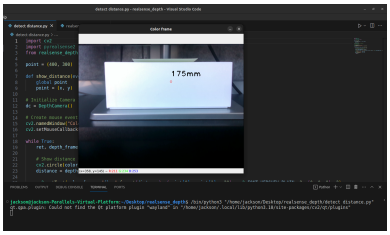
Quick Demo

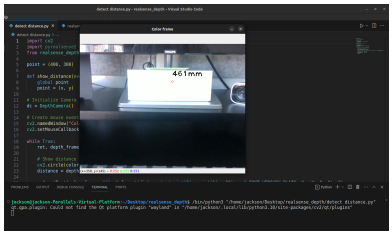
The estimated distance of the corresponding pixel position was marked on the colour frame as a clear indicator (for testing) at the moment.

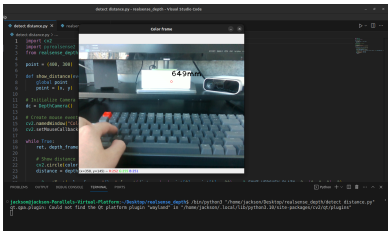




And when moving closer or further, the estimated distance would be **updated in real-time**. The following screenshots reflect the process of **moving away** from the object:







Known Issues

1. "Frame didn't arrive"

This issue happens when close the video streaming with Ctrl + C and it seems like the connection with the camera can't be re-established when trying to executing the script again. This may due to the "hard shut down" of the camera communication and requires graceful exit handling when intend to close the streaming. Current workaround would be reconnect the camera cable with the computer before executing the script next time.

```
jackson@jackson-Parallels-Virtual-Platform:~/Desktop/realsense_depth$ /bin/python3 "/home/jackson/Desktop/realsense_depth/detect distance.py"
qt.qpa.plugin: Could not find the Qt platform plugin "wayland" in "/home/jackson/.local/lib/python3.10/site-packages/cv2/qt/plugins"
Traceback (most recent call last):
  File "/home/jackson/Desktop/realsense_depth/detect distance.py", line 19, in <module>
    ret, depth_frame, color_frame = dc.get_frame()
  File "/home/jackson/Desktop/realsense_depth/realsense_depth.py", line 25, in get_frame
    frames = self.pipeline.wait_for_frames()
RuntimeError: Frame didn't arrive within 5000
```

2. Invalid Distance

Sometimes the estimated distance returned would be **zero** even though the camera is not close to the target object. This issue is not frequent and can not be reproduced every time. The valid reading would usually be returned immediately at next frame so it didn't affect human judgment but may impact the messages published to ROS topic (when it's been encapsulated into a ROS 2 node).