

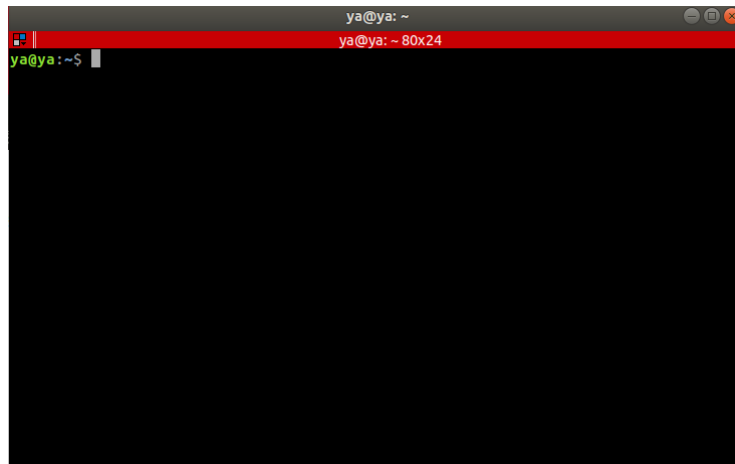
使用docker运行R2DIO

0. docker安装

- 官方安装链接: <https://docs.docker.com>

1. 终端解释

- 宿主机即自己的终端



- 容器即下面的终端



2. 运行程序

Step1 宿主机执行命令，下载镜像

```
sudo docker pull coolaogege/docker_r2dio:v.1.0
```

Step2 宿主机执行命令，开启xhost，使得docker里面可视化界面

```
xhost +
```

Step3 宿主机执行命令，创建容器，挂载数据容器卷

- 在 `/home/ya/` 目录下创建 `docker_share` 文件夹，这里 `/home/ya/` 改为自己的文件目录
- 将 `/home/ya/docker_share` 改为自己的文件夹路径 执行该命令

```
sudo docker run -it -v /home/ya/docker_share:/data --device=/dev/dri --group-add video --  
volume=/tmp/.X11-unix:/tmp/.X11-unix --env="DISPLAY=$DISPLAY" --env="QT_X11_NO_MITSHM=1" --  
name=docker_r2dio coolaoege/docker_r2dio:v.1.0 /bin/bash
```

Step4 容器内执行命令，退出容器

```
exit
```

Step5 宿主机命令，再次进入该容器

- 每次在重启之前一定要在宿主机进行 `xhost +`
- `5e918411fd45` 改为自己的容器id

```
xhost +  
sudo docker start -ia 5e918411fd45
```

Step6 编译

```
cd ~/catkin_ws/  
catkin_make  
source ./devel/setup.bash
```

Step7 将数据包移至宿主机的文件夹

- 之前设置的文件目录 `/home/ya/docker_share`

Step8 更改该容器终端的launch文件，运行程序

```
cd src/R2DIO/launch/  
gedit r2dio.launch
```

- 将该launch中这句话的 `subdata4` 改为自己的 数据包名字 然后保存

```
Open  ▾  [icon]  *r2dio.launch  Save  [icon]  -  [icon]  x
~/catkin_ws/src/r2dio/launch

1 <?xml version="1.0"?>
2 <launch>
3   <node pkg="roscpp" type="play" name="roscpp_play" args="--clock -r 1 /data/subdata4.bag"/>
4   <param name="/use_sim_time" value="true" />
5
6   <node pkg="tf" type="static_transform_publisher" name="word2map_tf" args="0 0 0 1.57 3.14 1.57 /world /map 100" />
7   <node pkg="tf" type="static_transform_publisher" name="base2imu_tf" args="0 0 0 0 0 0 /odom /camera_depth_optical_frame 100" />
8 <!--
9   <include file="$(find r2dio)/launch/rs_camera.launch">
10     <arg name="color_width" value="1280" />
11     <arg name="color_height" value="720" />
12     <arg name="filters" value="pointcloud" />
13   </include>
14   -->
15   <param name="file_path" value="$(find r2dio)/config/params.yaml"/>
16
17   <node pkg="r2dio" type="r2dio_laser_processing_node" name="r2dio_laser_processing_node" output="screen" />
18   <node pkg="r2dio" type="r2dio_odom_estimation_node" name="r2dio_odom_estimation_node" output="screen" />
19   <node pkg="r2dio" type="r2dio_laser_mapping_node" name="r2dio_laser_mapping_node" output="screen"/>
20
21   <arg name="rviz" default="true" />
22   <group if="$(arg rviz)">
23     <node launch-prefix="nice" pkg="rviz" type="rviz" name="rviz" args="-d $(find r2dio)/rviz/r2dio.rviz" />
24   </group>
25
26   <node pkg="hector_trajectory_server" type="hector_trajectory_server" name="trajectory_server_loam" ns="r2dio" >
27     <param name="/target_frame_name" value="map" />
28     <param name="/source_frame_name" value="odom" />
29     <param name="/trajectory_update_rate" value="10.0" />
30     <param name="/trajectory_publish_rate" value="10.0" />
31   </node>
32
33 </launch>
```

- 运行程序

```
cd ../../..
roslaunch r2dio r2dio.launch
```