AR Vision

This tutorial takes usb no-drive camera as an example. All codes run in docker. When entering docker, you need to mount the camera. For details, please refer to docker tutorial [5、Enter the docker Container].

1、overview

Augmented Reality，Abbreviated as "AR" technology，It is a technology that cleverly integrates virtual information with the real world, widely utilizing various technical means such as multimedia, 3D modeling, real-time tracking and registration, intelligent interaction, and sensing.It can simulate and simulate virtual information such as computer-generated text, images, 3D models, music, videos, etc., and apply them to the real world. The two types of information complement each other, thereby achieving "enhancement" of the real world.

The AR system has three prominent characteristics：① Information integration between real and virtual worlds; ② Real time interactivity; ③ It is the addition of positioning virtual objects in three-dimensional scale space.

Augmented reality technology includes new technologies and new approaches such as multimedia, 3D modeling, real-time video display and control, multi-sensor fusion, real-time tracking and registration, and scene fusion.

2、Usage method

When using AR cases, it is necessary to have camera internal parameters, otherwise it cannot run. The internal parameter file is the same directory as the code, and different cameras correspond to different internal parameters. In this lesson, the usb no-drive camera is used as an example, and the internal calibration can be quickly calibrated with checkerboard (**this step is already done in docker images**)

There are a total of 12 effects in this section.

```
["Triangle", "Rectangle", "Parallelogram","WindMill","TableTennisTable",
"Ball","Arrow", "Knife", "Desk","Bench", "Stickman", "ParallelBars"]
```

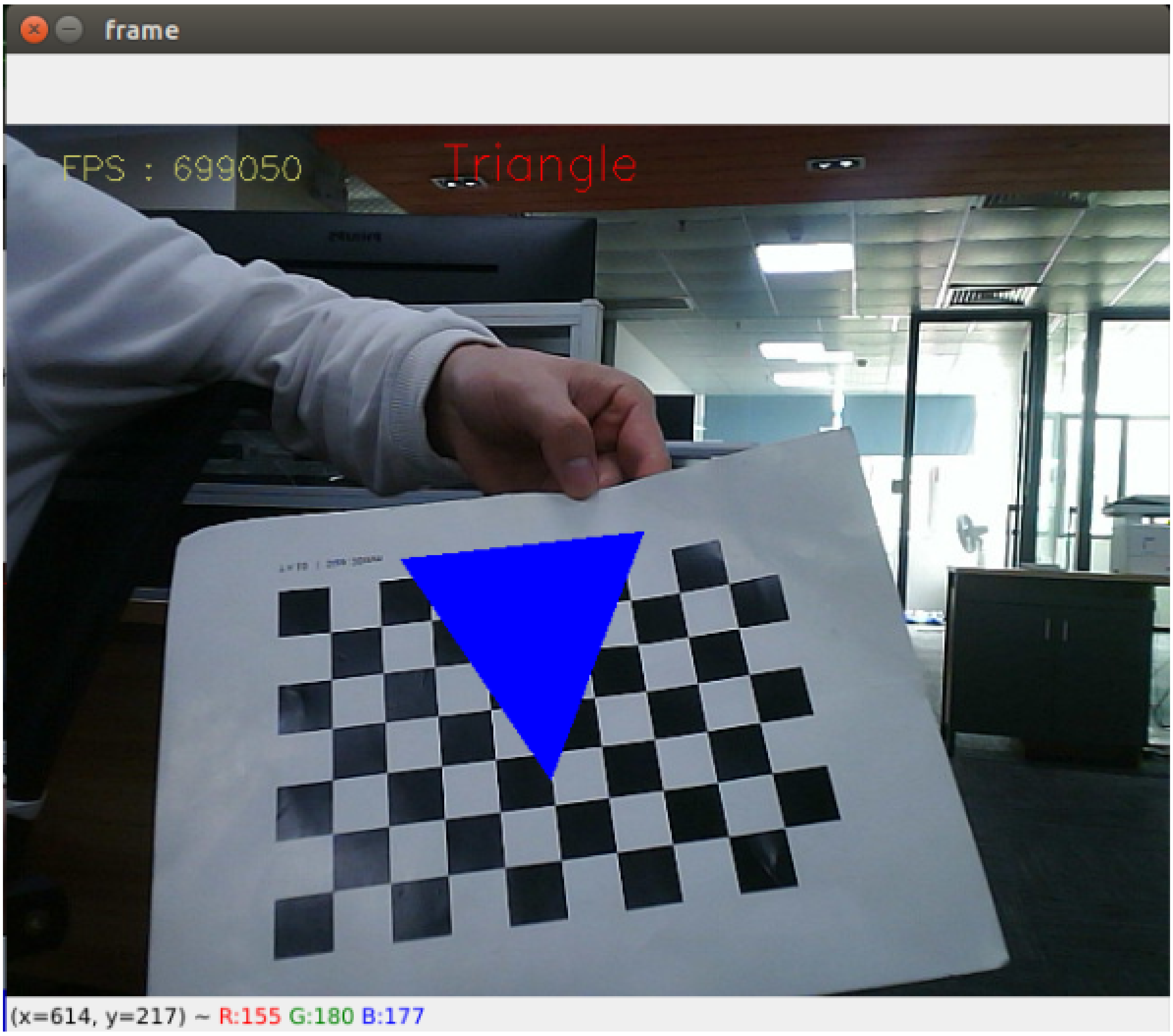
3、start command

Code reference path

```
/root/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_visual/yahboomcar_visual/simple_AR.py
```

After entering the Docker container, the Docker terminal inputs

```
ros2 run yahboomcar_visual simple_AR
```



Press【q】key to exit，Press【f】key to switch between different effects.

Note: Some motherboards due to general performance, GUI display very card, resulting in key function failure, sometimes can use the key switch and sometimes press no response, if you need to switch different effects, you can refer to the following release of different topics to achieve the switching effect

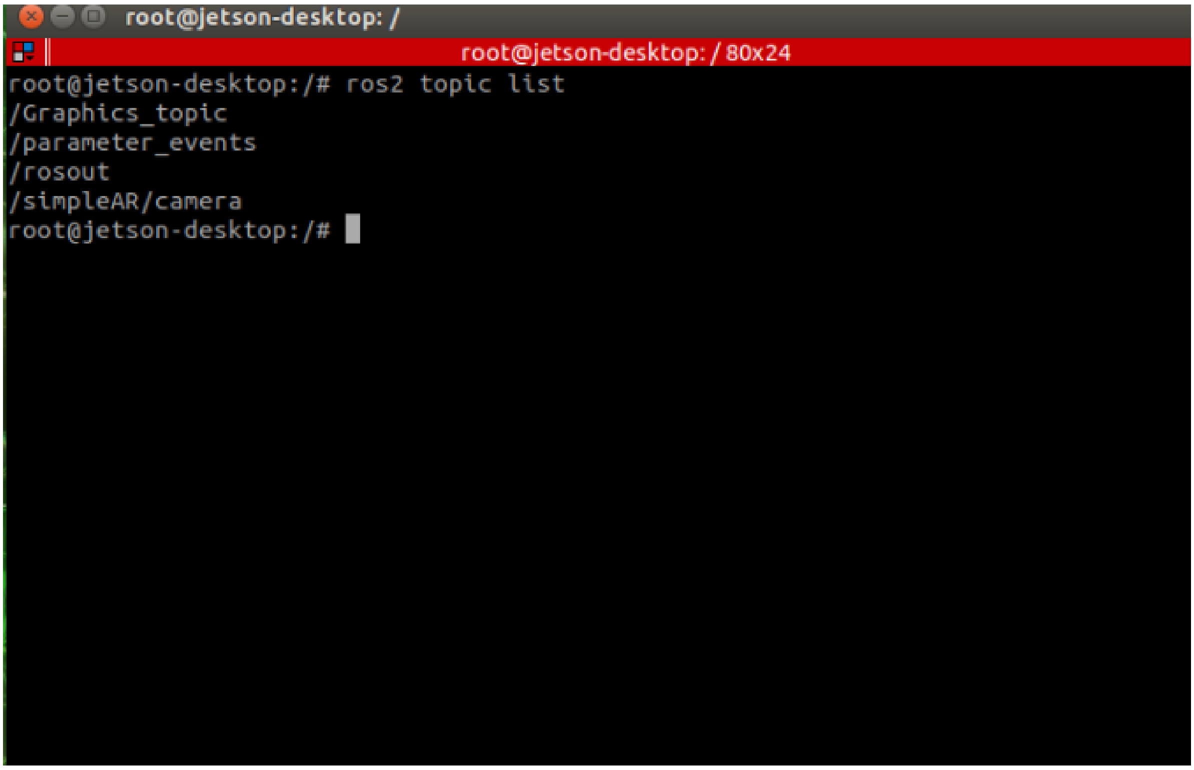
3.1.1、ROS deployment

This course also deployed ROS, which mainly has the following two functions:

- Subscribe to topic data and switch between different effects
- Publish Image

View the ROS topic through the following command, enter it on the Docker terminal

```
ros2 topic list
```



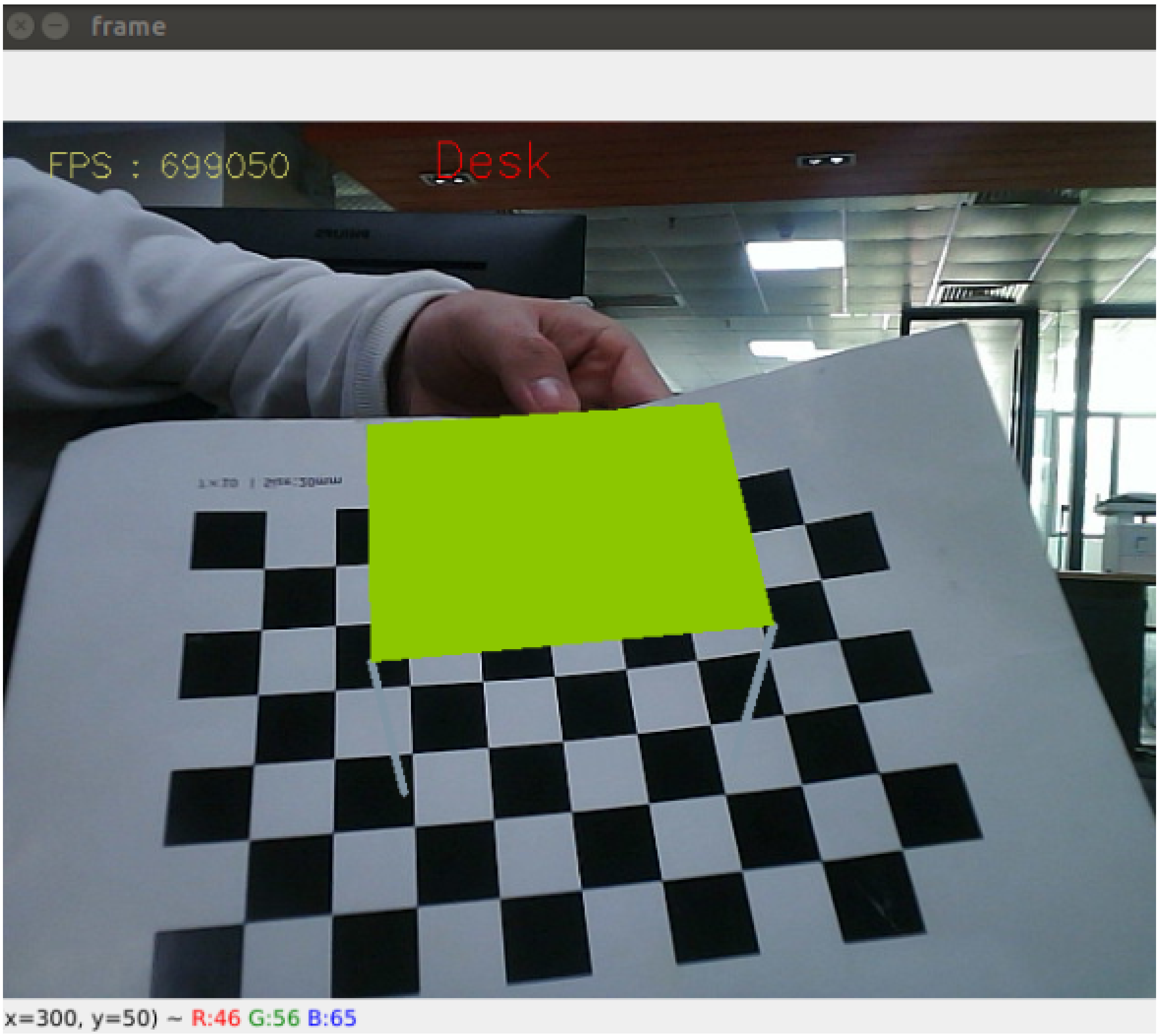
```
root@jetson-desktop:/# ros2 topic list
/Graphics_topic
/parameter_events
/rosout
/simpleAR/camera
root@jetson-desktop:/#
```

- /Graphics_topic： The topic name of the effect and the effect that needs to be recognized for subscription.
- /simpleAR/camera： The topic name of the image, publishing the image.

The effect can be modified using the following command. For example, I will first modify it to Desk (The effect name has been given above, you can modify it yourself)

enter it on the Docker terminal

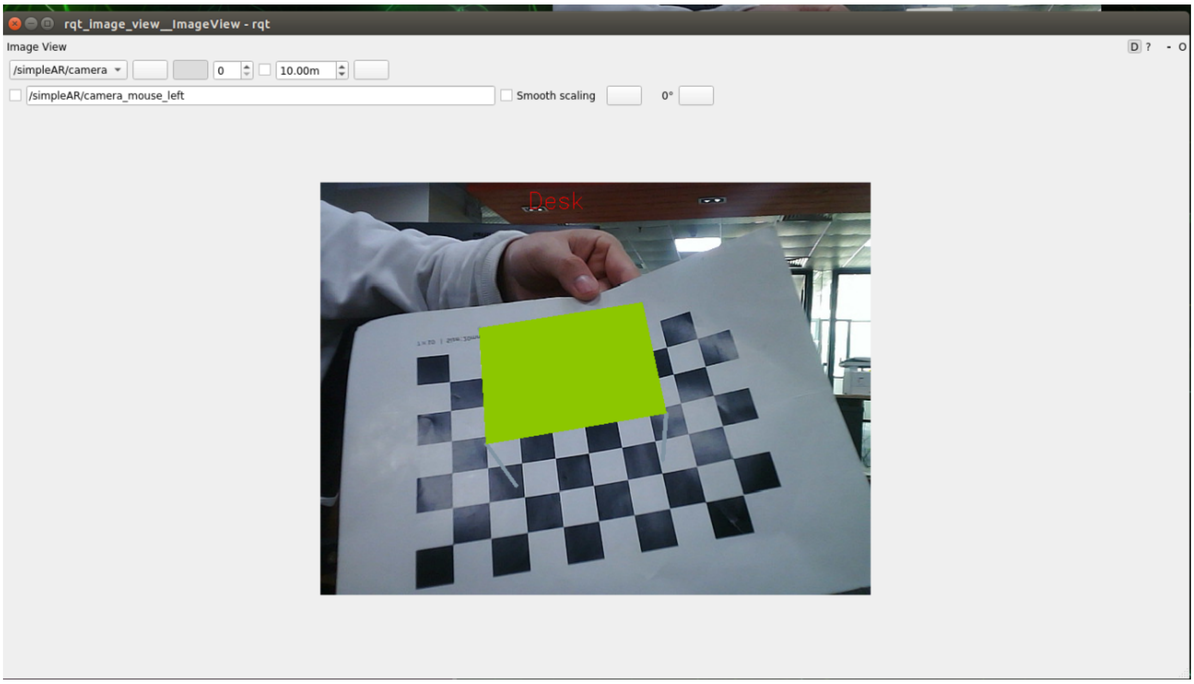
```
ros2 topic pub /Graphics_topic std_msgs/msg/String "data: Desk"
```



x=300, y=50) ~ R:46 G:56 B:65

Viewing published images can be done using rqt_ Image_ View to view, Docker terminal input

```
ros2 run rqt_image_view rqt_image_view
```



/simpleAR/camera

0

10.00m

□ /simpleAR/camera_mouse_left

□ Smooth scaling

0°

Select /simpleAR/camera in the top left corner of the topic to view the image.