3、Astra object tracking

3.1、Introduction

Function package：~/transbot_ws/src/transbot_astra

Website：https://learnopencv.com/object-tracking-using-opencv-cpp-python/#opencv-tracking-api

Object tracking is to locate an object in consecutive video frames.

- Comparison of OpenCV algorithms

Algorithm	Speed	Accuracy	Description
BOOSTING	Slow	Low	It is the same as the machine learning algorithm behind Haar casades (AdaBoost), but it has been born for more than ten years, a veteran algorithm.
MIL	Slow	Low	It is more accurate than BOOSTING, but the failure rate is higher.
KCF	Fast	High	Faster than BOOSTING and MIL, but it is not effective when there is occlusion
TLD	Middle	Middle	There are a lot of erro
MEDIANFLOW	Middle+	Middle	The model will fail for fast-jumping or fast-moving objects.
GOTURN	Middle	Middle	A deep learning-based object detector requires additional models to run.
MOSSE	Fastest	High	The speed is really fast, but not as high as the accuracy of CSRT and KCF. If you are looking for speed, you can choose it.
CSRT	Fast -	Higher	Slightly more accurate than KCF, but not as fast as KCF.

3.2、Steps

Note: The [R2] of the handle remote controller can [Pause/Open] for all functions of robot car

3.2.1、Start up

Method 1

Start up Astra camera（jetson nano side）

```
roslaunch transbot_astra DepthSrv.launch colorCamera:=true
```

Start color object control (virtual machine)

```
roslaunch transbot_astra AstraTracker.launch VideoSwitch:=true tracker_type:=KCF
```

- colorCamera: Whether to start the color camera. true, start the color camera; false, do not start.

Method 2

Start up Astra camera（jetson nano side）

```
roslaunch transbot_astra DepthSrv.launch colorCamera:=false
```

Note: press【q】key to exit.

```
python3 ~/transbot_ws/src/transbot_astra/scripts/astra_Tracker.py
```

This method can only be activated in the master controller that the camera is connected.

- colorCamera parameter: whether to start the color camera; start is True; otherwise, it is False.
- VideoSwitch parameter: whether to use the camera function package to start; it corresponds to the colorCamera parameter and cannot be the same.

Set the parameters according to your needs, and you can also modify the launch file directly, so you don't need to attach parameters when you start.

3.2.2、Identify

After starting, enter the selection mode, use the mouse to select the location of the object, as shown in the figure below, release it to start recognize.

Keyboard key control:

【r】: Color selection mode, the mouse can be used to select the area of the color to be recognized (cannot exceed the area range).

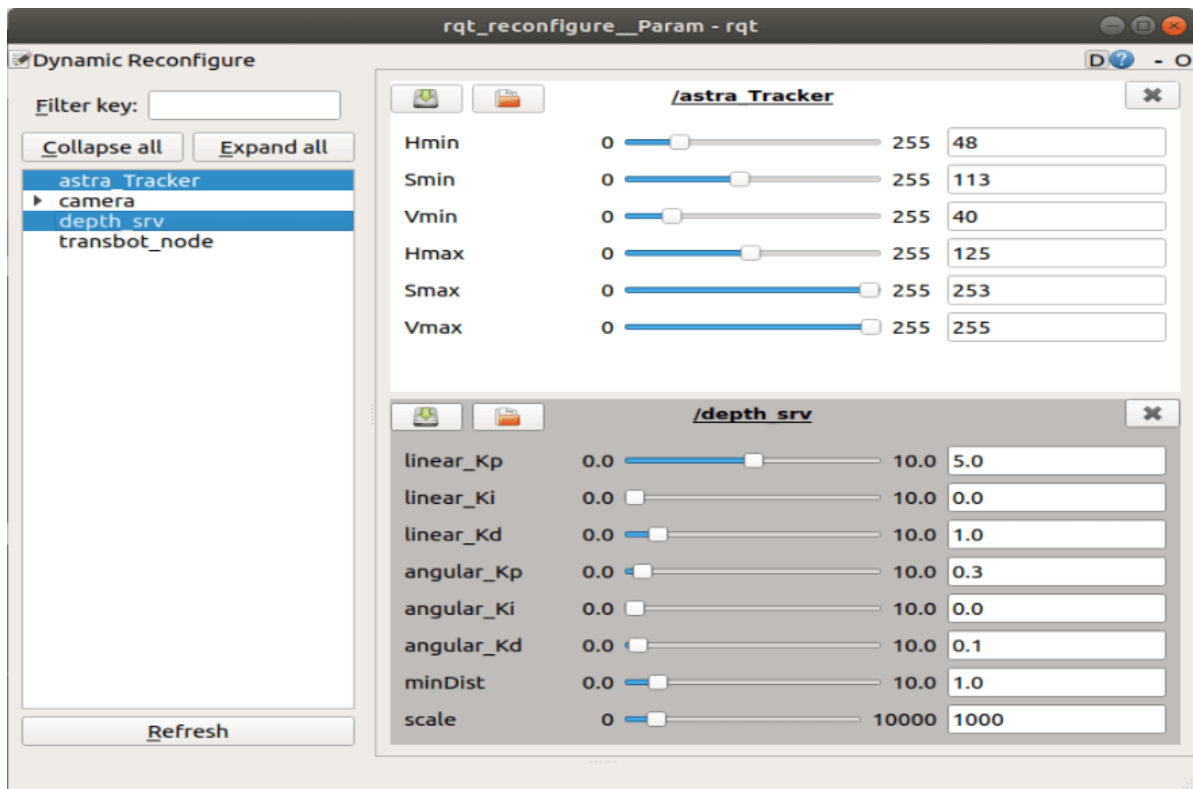【f】: Switching algorithm: ['BOOSTING','MIL','KCF','TLD','MEDIANFLOW','MOSSE','CSRT','color'].

【q】: Exit the program.

【Space key】: Color follow.

3.2.3、 PID adjustment

Dynamic parameter

```
rosrun rqt_reconfigure rqt_reconfigure
```

Select【depth_srv】node.

The slider is always in the dragging state, and no data will be transferred to the system. The data will actually be transferred to the system when you release it; you can also select a row and then slide the mouse wheel.

Parameter analysis:

[Linear_Kp], [linear_Ki], [linear_Kd]: PID control of linear speed in the process of car following.

[Angular_Kp], [angular_Ki], [angular_Kd]: PID control of angular velocity during the car following process.

[MinDist]: Follow the distance and keep this distance all the time.

[Scale]: PID scaling.

3.2.4、 Target follow

After identifying is ok, click [Space key] on the keyboard to execute the color following program.

- View node

```
rqt_graph
```

The node [astra_Tracker] will publish the position of the identified object in the image to the node [depth_srv], and the node [depth_srv] will issue the car following control instructions.
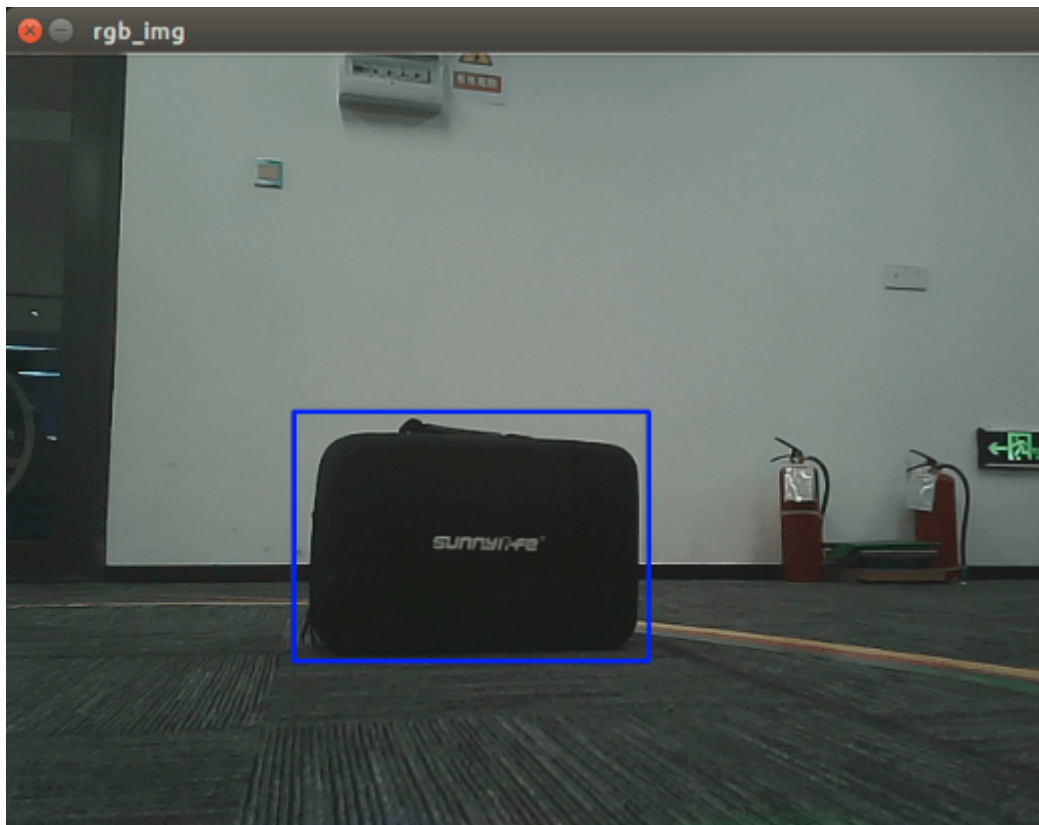
3.3、 C++ KCF

3.3.1、 Usage

Start up (jetson nano side)

```
roslaunch transbot_astra KCFTracker.launch
```

After starting, enter the selection mode, use the mouse to select the location of the object, as shown below, release it to start recognition.
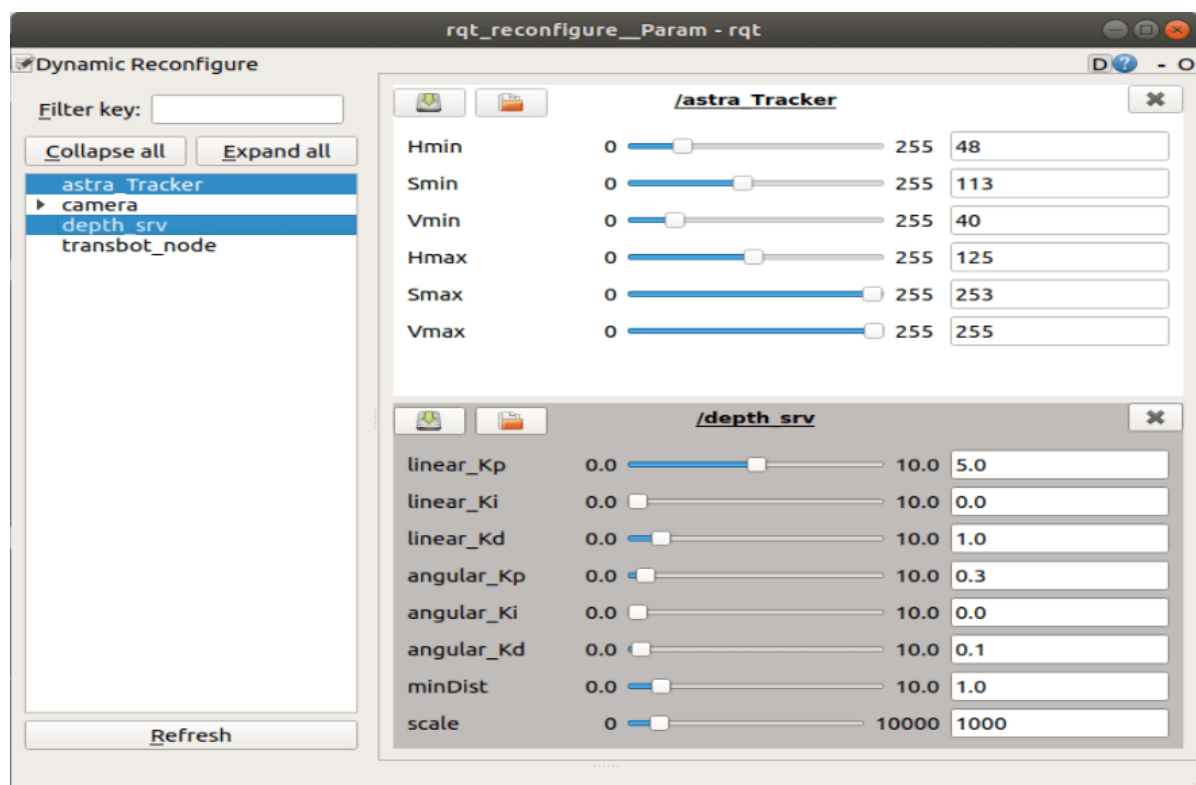

3.3.2、keyboard control

【r】：Selection mode, you can use the mouse to select the area to be identified.

【q】：Exit the program.

【Space bar】：Target tracking; move the target slowly while following, and the target will be lost if you move too fast.

Parameter analysis:

【linear_Kp】、【linear_Ki】、【linear_Kd】：PID control of linear speed during robot following process.

【angular_Kp】、【angular_Ki】、【angular_Kd】：The angular velocity PID control during the following process of the robot.

【minDist】：Follow the distance and keep this distance all the time.

3.3.3、Node relationship graph

```
rqt_graph
```