

Yolov5 Garbage Recognition with Voice Broadcast (Jetson Nano)

This course is exclusive to Jetson-Nano board users. Orin board users please refer to the course content in [11.2 Garbage Recognition with Voice Broadcast (Jetson Orin)]

Before running the function, you need to close the App and large programs. For the closing method, refer to [4.Preparation] - [1. Manage APP control services].

1. Function Description

Voice broadcast the name and type of recognized garbage.

2. Startup and Operation

2.1. Startup

This case runs on the host machine.

First, you need to start roscore. Open the terminal and enter the following:

```
roscore
```

Then you need to start the ROS node service. Open the terminal and enter the following:

```
rosrun dofbots_pro_info kinematics_dofbot_pro
```

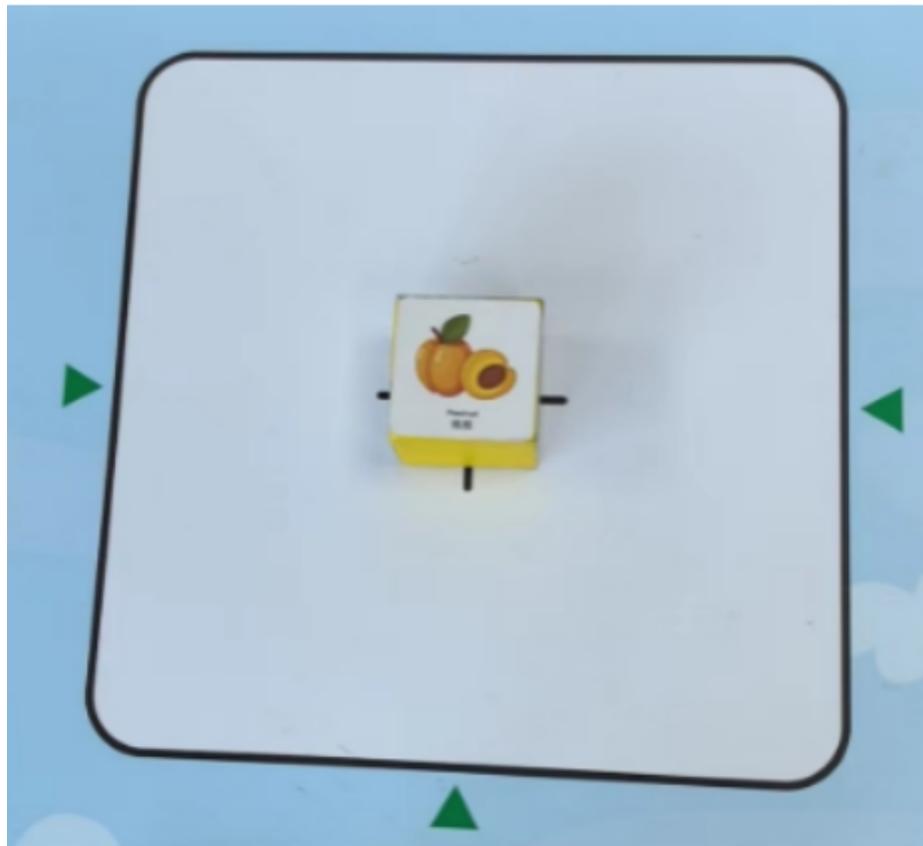
Open a third terminal and enter the following command:

```
python3 ~/dofbot_voice/scripts/garbage_broadcast.py
```

The Jetson-Nano board startup will be relatively slow. Please wait patiently until the [Model-Loading] on the image screen disappears, indicating initialization is complete.

2.2. Operation Steps

Place the garbage label code block in the center of the image, then say "Hello, yahboom" to the voice module. The voice module will reply "here" to indicate successful wake-up. Then say "What garbage is this?" to the voice module. After the program recognizes it, the voice module will reply and broadcast what this garbage is and its type. Taking the figure below as an example, according to the recognition result, it will reply "This is a peach pit, which belongs to dry garbage".



3. Core Code Analysis

Source code path: ~/dofbot_voice/scripts/garbage_broadcast.py

```
#Import voice broadcast garbage type library, this library is located at
#/home/jetson/dofbot_pro/dofbot_garbage_yolov5/speech_garbage.py
from speech_garbage import speech_garbage

#Mainly look at the content of this library
#Import garbage recognition library, this library is located at
#/home/jetson/dofbot_pro/dofbot_garbage_yolov5/garbage_identify.py
from garbage_identify import garbage_identify

#Call the garbage recognition function, the input parameter is the current image,
then return the recognized processed image and recognition results
self.frame, msg = self.garbage_identify.garbage_run(self.frame)

#Traverse the recognition results msg, determine the number self.garbage_num and
type self.garbage_class represented by the garbage name according to the value of
name
for key, pos in msg.items(): name = key
if name == "zip_top_can":           (self.garbage_num, self.garbage_class) =
('00', '01')
if name == "old_school_bag":        (self.garbage_num, self.garbage_class) =
('01', '01')
if name == "Newspaper":             (self.garbage_num, self.garbage_class) =
('02', '01')
if name == "Book":                 (self.garbage_num, self.garbage_class) =
('03', '01')
if name == "Toilet_paper":          (self.garbage_num, self.garbage_class) =
('04', '02')
if name == "Peach_pit":             (self.garbage_num, self.garbage_class) =
('05', '02')
```

```
if name == "Cigarette_butts":           (self.garbage_num, self.garbage_class) =  
('06', '02')  
if name == "Disposable_chopsticks":     (self.garbage_num, self.garbage_class) =  
('07', '02')  
if name == "Egg_shell":                  (self.garbage_num, self.garbage_class) =  
('08', '03')  
if name == "Apple_core":                (self.garbage_num, self.garbage_class) =  
('09', '03')  
if name == "Watermelon_rind":          (self.garbage_num, self.garbage_class) =  
('10', '03')  
if name == "Fish_bone":                 (self.garbage_num, self.garbage_class) =  
('11', '03')  
if name == "Expired_tablets":          (self.garbage_num, self.garbage_class) =  
('12', '04')  
if name == "Expired_cosmetics":        (self.garbage_num, self.garbage_class) =  
('13', '04')  
if name == "Used_batteries":           (self.garbage_num, self.garbage_class) =  
('14', '04')  
if name == "Syringe":                  (self.garbage_num, self.garbage_class) =  
('15', '04')  
if name == "None":                     (self.garbage_num, self.garbage_class) =  
('None', 'None')  
#Get voice recognition result  
result = mySpeech.speech_read()  
#If the current voice recognition result is 94, it means asking what the current  
garbage is  
if result == 94:  
    if self.garbage_num == '00':  
        mySpeech void_write(94)  
  
    elif self.garbage_num == '01':  
        mySpeech void_write(95)  
  
    elif self.garbage_num == '02':  
        mySpeech void_write(96)  
  
    elif self.garbage_num == '03':  
        mySpeech void_write(97)  
  
    elif self.garbage_num == '04':  
        mySpeech void_write(109)  
  
    elif self.garbage_num == '05':  
        mySpeech void_write(108)  
  
    elif self.garbage_num == '06':  
        mySpeech void_write(107)  
  
    elif self.garbage_num == '07':  
        mySpeech void_write(106)  
  
    elif self.garbage_num == '08':  
        mySpeech void_write(105)  
  
    elif self.garbage_num == '09':  
        mySpeech void_write(104)  
  
    elif self.garbage_num == '10':
```

```
mySpeech.void_write(103)

elif self.garbage_num == '11':
    mySpeech.void_write(102)

elif self.garbage_num == '12':
    mySpeech.void_write(101)

elif self.garbage_num == '13':
    mySpeech.void_write(100)

elif self.garbage_num == '14':
    mySpeech.void_write(99)

elif self.garbage_num == '15':
    mySpeech.void_write(98)
```