

Palm control car movement

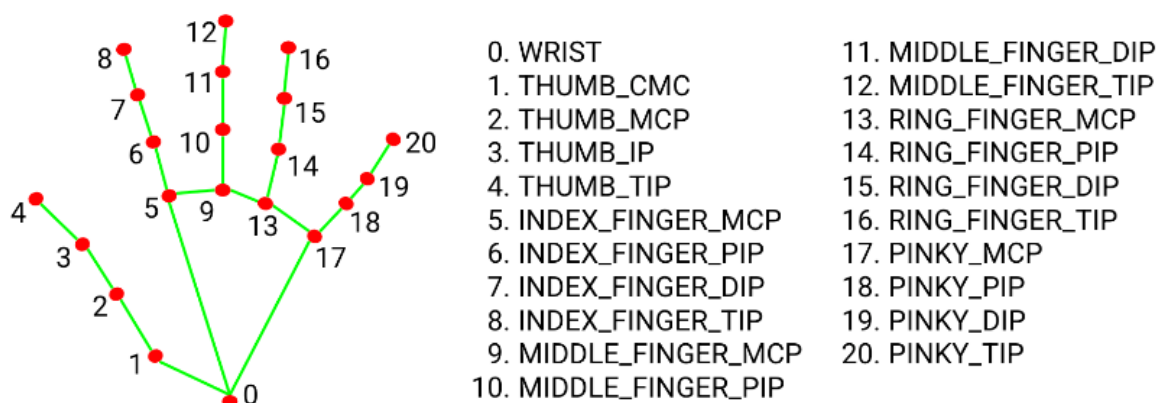
1. Introduction to mediapipe

MediaPipe is a data stream processing machine learning application development framework developed and open sourced by Google. It is a graph-based data processing pipeline used to build data sources in various forms, such as video, audio, sensor data, and any time series data. MediaPipe is cross-platform and can run on embedded platforms (Raspberry Pi, etc.), mobile devices (iOS and Android), workstations, and servers, and supports mobile GPU acceleration. MediaPipe provides cross-platform, customizable ML solutions for real-time and streaming media.

MediaPipe Hands is a high-fidelity hand and finger tracking solution. It uses machine learning (ML) to infer 21 3D coordinates of hands from a frame.

After palm detection of the entire image, the 21 3D hand joint coordinates in the detected hand area are accurately located by regression based on the hand marker model, that is, direct coordinate prediction. The model learns a consistent internal hand posture representation that is robust even to partially visible hands and self-occlusion.

To obtain ground truth data, about 30K real-world images were manually annotated with 21 3D coordinates, as shown below (Z values were obtained from the image depth map, if each corresponding coordinate had a Z value). In order to better cover possible hand poses and provide additional supervision on the properties of the hand geometry, high-quality synthetic hand models were also drawn in various backgrounds and mapped to the corresponding 3D coordinates.



2. Program Description

The examples in this section may run very slowly on the robot main control. After the palm can be recognized, the car can be set up for testing first, which will have a better effect.

The car will control the movement of the chassis according to the position of the palm in the picture.

Palm on the top of the screen -> Car moves forward

Palm on the bottom of the screen -> Car moves backward

Palm on the left of the screen -> Car moves left

Palm on the bottom of the screen -> Car moves right

2.1, Source code path

After entering the docker container, the source code of this function is located at,

```
/root/yahboomcar_ws/src/yahboomcar_mediapipe/yahboomcar_mediapipe/
```

3. Program startup

3.1. Startup command

Open a terminal and enter the following command to enter docker,

```
./docker_ros2.sh
```

The following interface appears, which means that you have successfully entered docker

```
pi@yahboom:~ $ ./docker_ros2.sh
access control disabled, clients can connect from any host
root@yahboom:/#
```

Start the chassis

```
ros2 launch yahboomcar_bringup bringup.launch.py
```

Open a new terminal, enter the same docker, and change the following da8c4f47020a to the ID displayed in the actual terminal

```
docker ps
```

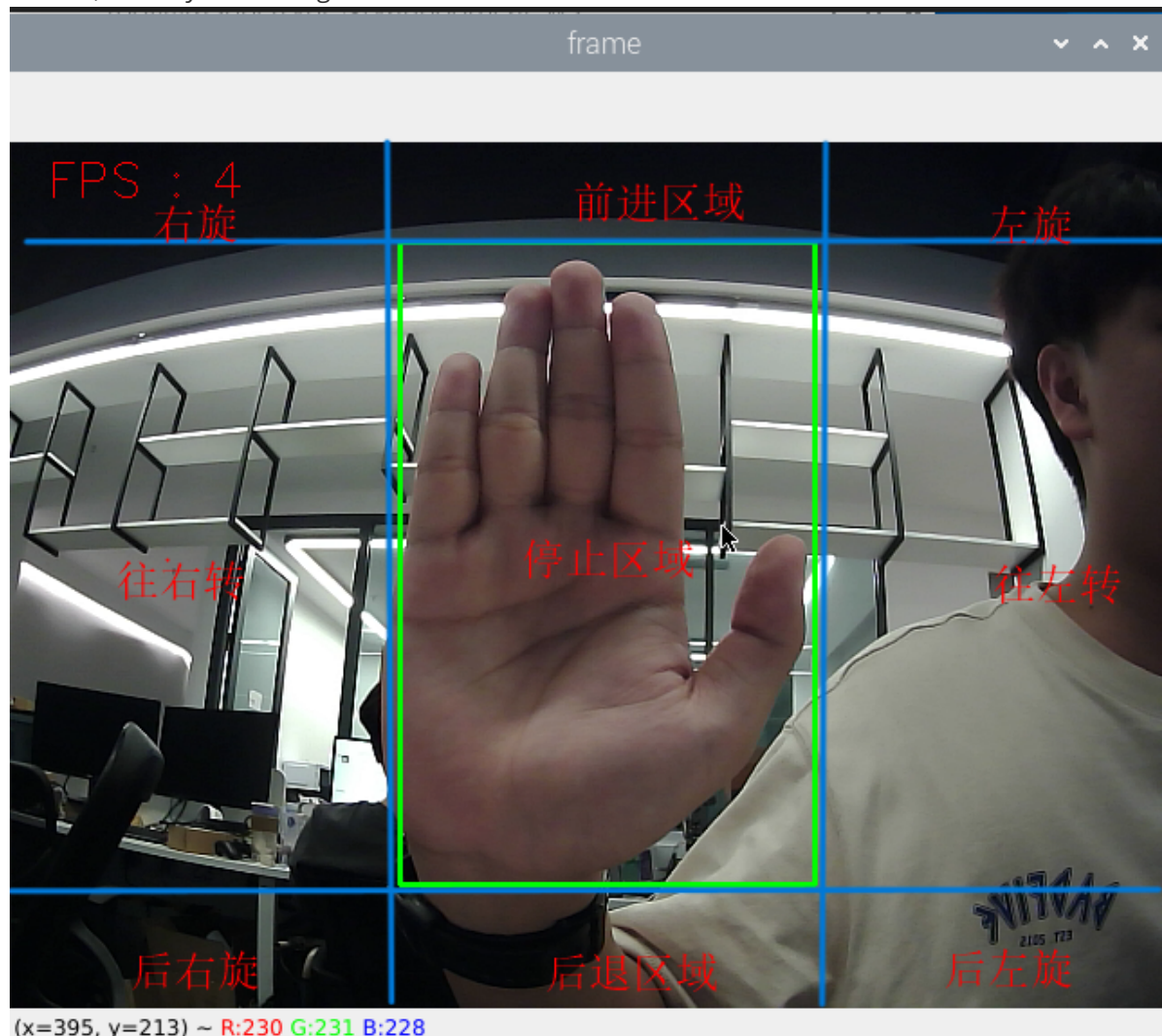
```
docker exec -it da8c4f47020a /bin/bash
```

```
pi@yahboom:~ $ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS
PORTS         NAMES
da8c4f47020a   yahboomtechnology/ros-humble:0.0.4 "/ros_entrypoint.sh ..." 8 hours ago   Up 45 minute
s
festive_payne
pi@yahboom:~ $ docker exec -it da8c4f47020a /bin/bash
root@yahboom:/#
```

After entering the docker container, enter in the terminal,

```
ros2 run yahboomcar_mediapipe RobotCtrl
```

Turn on this function, then put your hand in front of the camera, the screen will recognize your palm, and after the program recognizes the position of your palm, it will send the speed to the chassis, thereby controlling the movement of the car.



4. Core code

4.1. RobotCtrl.py

- Code reference location

```
/root/yahboomcar_ws/src/yahboomcar_mediapipe/yahboomcar_mediapipe
```

- Code analysis

1. Import the corresponding library file

```
from media_library import *
```

This library file mainly includes detecting palms, fingers and obtaining the coordinates of each finger joint.

2. Detect the palm and obtain the coordinates of the fingers

```
fingers = self.hand_detector.fingersUp(lmList)
point_x = lmList[9][1] #x value
point_y = lmList[9][2] #y value
```

Combining the picture in 1. Introduction, we can know that what we get is actually the coordinates of the first joint of the middle finger of our palm. By judging the position of this coordinate in the picture and sending it to the speed in the xy direction of the chassis, control can be achieved.

4.2, Flowchart

