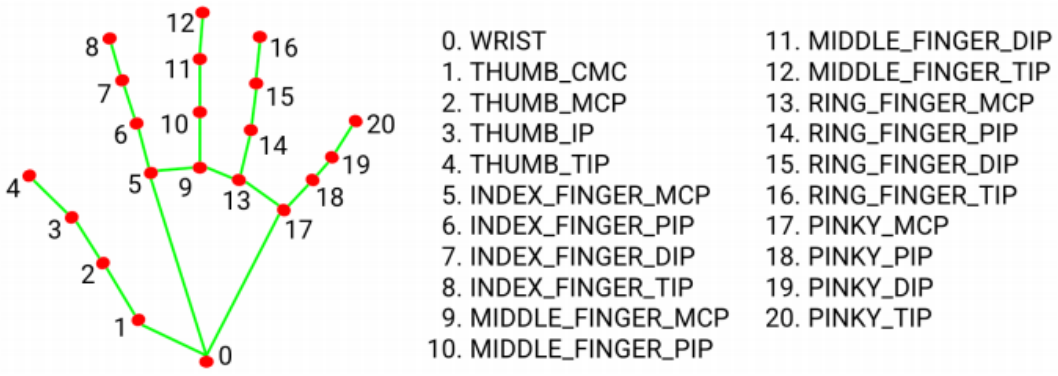
Gesture control car movement

1. Program function description

After the function is turned on, the camera captures images and recognizes gestures to control the movement of the car.

Gesture "5"	Car moves forward
Fist	Car moves backward
Gesture "1"	Car rotates left
Gesture "2"	Car rotates right

MediaPipe Hands infers the 3D coordinates of 21 hand value joints from one frame



2. Program code reference path

After entering the docker container, the location of the function source code is,

```
/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:/#
```

After entering the docker container, enter the terminal,

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

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

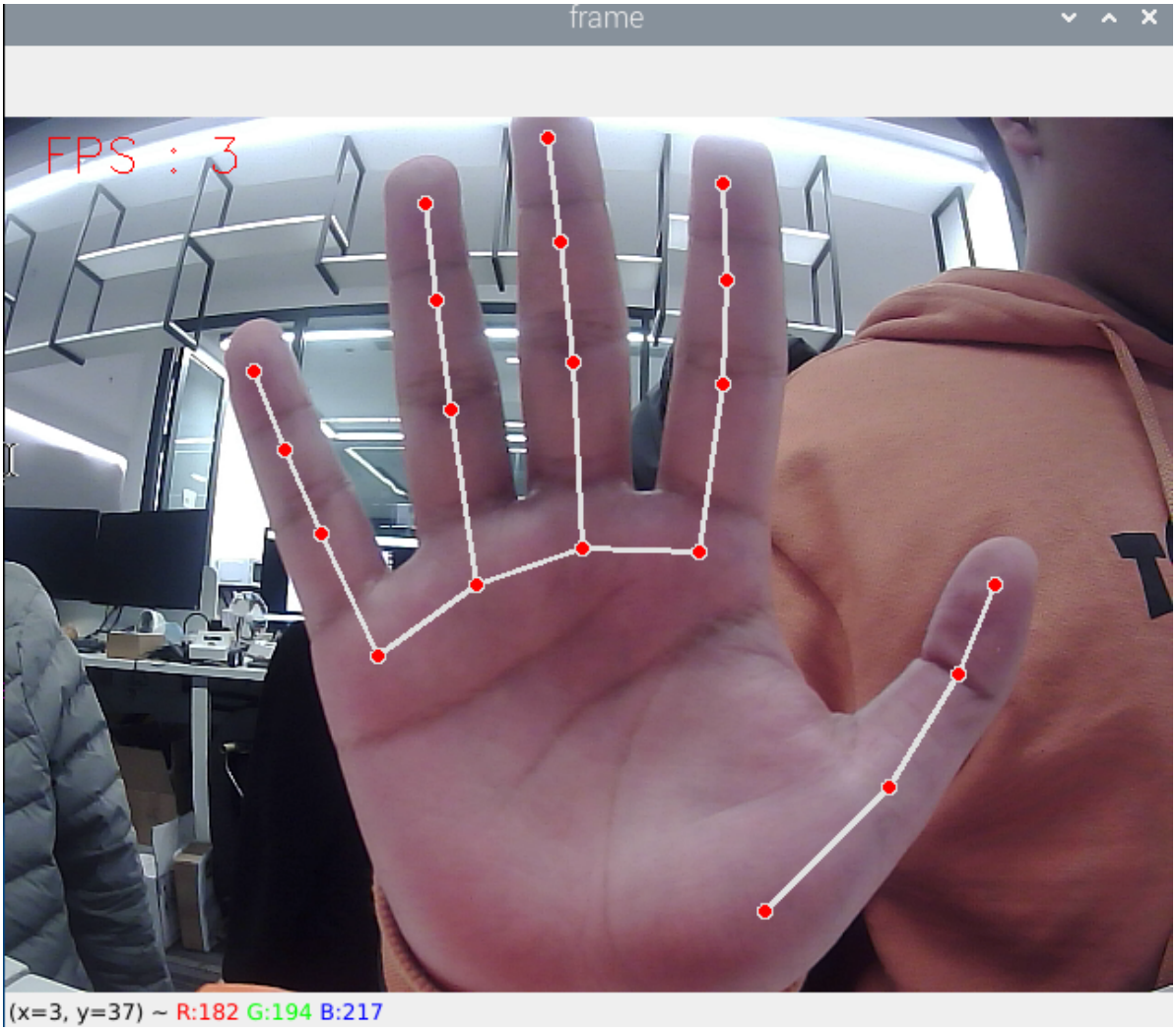
```
docker ps
```

```
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:/#
```

```
ros2 run yahboomcar_mediapipe HandCtrl
```

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



4. Core code

4.1、HandCtrl.py

- **Import key libraries**

```
from media_library import * #This library contains functions such as
detecting hands and getting gestures
```

- Get finger data

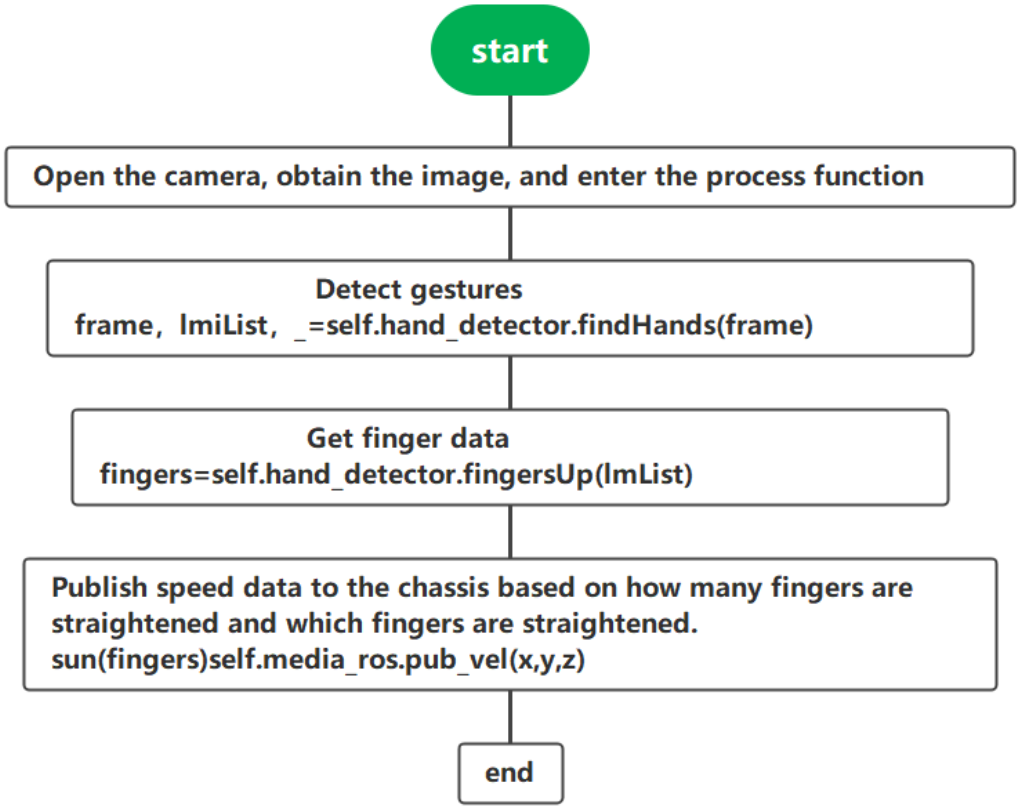
```
frame, lmList, _ = self.hand_detector.findHands(frame)
fingers = self.hand_detector.fingersUp(lmList)
sum(fingers)
fingers[]
```

It can be seen that the hand is detected first, the value of lmList is obtained, and then it is passed to the fingersUp function. The fingersUp function is used to detect which fingers are stretched. The value of a stretched finger is 1. The specific code here can also be seen in the media_library,py function, which has a detailed explanation. In fact, it is to judge the xy value of the finger joint to determine when it is stretched. The sum(fingers) function is used to count the number of stretched fingers. fingers[] can be used to enumerate fingers. For example, the index finger is represented by fingers[1].

- **Release speed to chassis**

```
self.media_ros.pub_vel(x,y,z) #This function is also in media_library,py
```

**4.2、flow chart


start
Open the camera, obtain the image, and enter the process function
Detect gestures
frame，lmiList，_=self.hand_detector.findHands(frame)
Get finger data
fingers=self.hand_detector.fingersUp(lmList)
Publish speed data to the chassis based on how many fingers are straightened and which fingers are straightened.
sun(fingers)self.media_ros.pub_vel(x,y,z)
end