9.Mediapipe palm control car

9.1. Introduction

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 for building data sources that use many 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.

9.2. Using

Note: [R2] of the remote controller has the function of [pause/on] for this gameplay.

The case in this section may run very slow on the robot master. It is recommended to connect the camera on the virtual machine side and run the [01_HandCtrlArm.launch] file. The NX master will work better and you can try it.

```
#You need to enter docker first, perform this step more
#If running the script to enter docker fails, please refer to 07.Docker-orin/05,
Enter the robot's docker container
~/run_docker.sh
roslaunch arm_mediapipe mediaArm.launch        #robot
rosrun arm_mediapipe RobotCtrl.py  #recommended virtual machine (with camera)
```

After startup, press R2 of the handle to turn on this function, and you can use the web page to view the screen.

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

The palm at the top of the screen -> the car advances

The palm is at the bottom of the screen -> the car moves back

The palm is on the left side of the screen -> the cart moves to the left

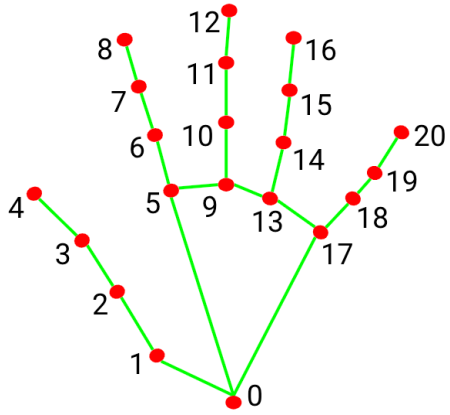
The palm is at the bottom of the screen -> the cart moves to the right

9.3、MediaPipe Hands

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

After palm detection is performed on the entire image, accurate key point positioning is performed on the 21 3D hand joint coordinates in the detected hand region by regression according to the hand marker model, that is, direct coordinate prediction. The model learns consistent internal hand pose representations and is robust to even partially visible hands and self-occlusion.

To obtain the 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). To better cover the possible hand poses and provide additional supervision on the nature of the hand geometry, high-quality synthetic hand models in various contexts are also drawn and mapped to the corresponding 3D coordinates.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

9.4. Core files

9.4.1、mediaArm.launch

```
<launch>
    <include file="$(find yahboomcar_ctrl)/launch/yahboom_joy.launch"/>
    <include file="$(find yahboomcar_bringup)/launch/yahboomcar.launch"/>
    <node pkg="web_video_server" type="web_video_server" name="web_video_server"
output="screen"/>
    <node name="msgToimg" pkg="arm_mediapipe" type="msgToimg.py" output="screen"
required="true"/>
</launch>
```

9.4.2、RobotCtrl.py

- Code reference location

  ```
  ~/yahboomcar_ws/src/arm_mediapipe/scripts
  ```

- 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, get the finger coordinates

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

Combining with the pictures in 9.3, we can know that what is actually obtained is 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 of the chassis in the xy direction, the control can be realized.

9.5. Flowchart