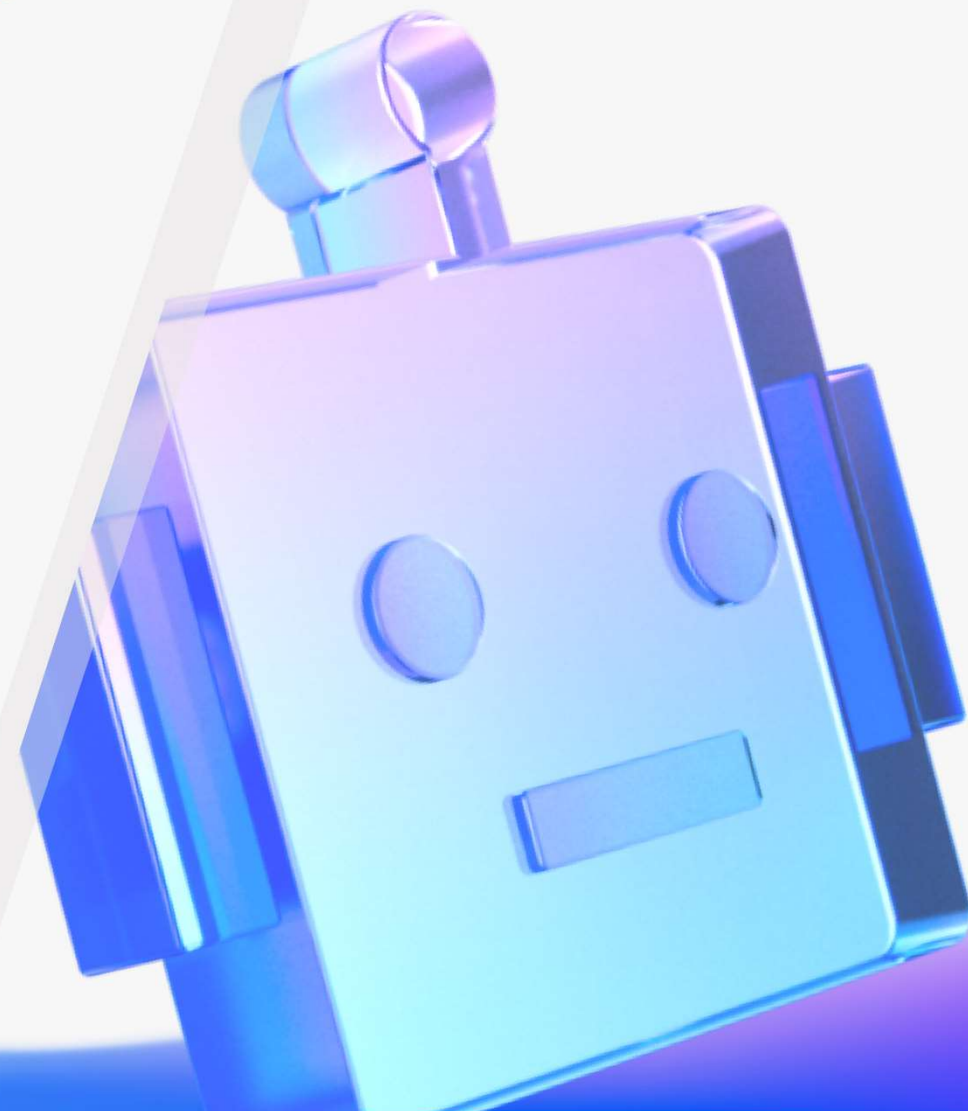


嵌入式开发板 应用与实践



地平线——全球领先的智能计算平台提供商

中国第一家

深度神经网络芯片公司

中国第一颗

已量产车规级计算芯片

全球唯三

规模化量产专用ADAS芯片厂商

全球第一款

支持8M像素的ADAS芯片

全球技术领导者



1000+
专利



2000
研发人员

商业落地先行者



120+
车型前装定点



500 万
征程及旭日芯片出货



100+
生态合作伙伴

Global Recognition



2020

4项冠军

Waymo 全球自动
驾驶算法挑战赛



2019

Top 50

全球50家
最聪明的企业



2019

2019CES创新奖

Matrix™自动驾
驶计算平台



Top 30

世界经济论坛
科技领军者

创始人余凯博士



余凯, 博士
地平线创始人
首席科学家

学术论文被国际同行引用超**30000**次
科技部新一代人工智能战略规划委员会委员
中国证监会科技监管委员会委员

学术与从业经历



南京大学



慕尼黑大学



斯坦福大学

SIEMENS

NEC Laboratories
America

Baidu 百度

Microsoft

创始人成就



第一届ImageNet挑战赛全球冠军 (2010)



创立中国第一家深度学习研发机构(2012)



创立中国第一个自动驾驶项目 (2013)



创立中国第一家汽车智能芯片公司(2015)



首位在国际AI竞赛中夺冠的**华人科学家**



PASCAL VOC
2009



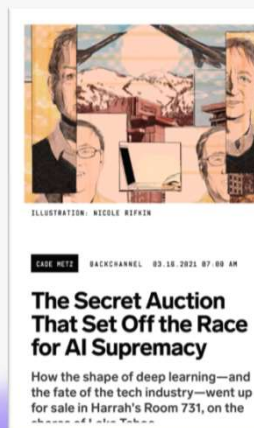
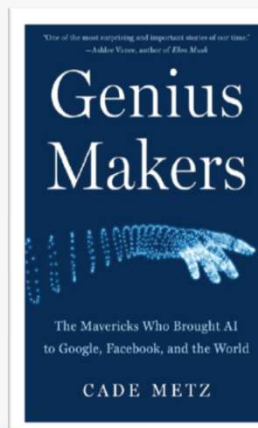
第一届 ImageNet
2010

社会影响力与行业评价

“Kai is a pioneer in the deployment of deep learning and convolution nets”



—— Yann LeCun
卷积神经网络奠基人
Facebook AI实验室创始人



自动驾驶汽车是人类工业史上最有挑战的系统工程





深蓝 S7



理想 L8



理想 ONE



上汽大通 MAXUS MIFA 9



理想 L7



哪吒 GT



欧萌达 OMODA 5



哈弗H9



长安 UNI-K



传祺 影豹



江淮QX



瑞虎8 PRO



哪吒 U-II



凯捷 HEV

50+
量产上市车型

300万+
征程芯片出货

120+
前装定点车型



深蓝SL03



博越 L



荣威 RX5



长安 UNI-V



江淮 X8 PLUS



岚图 FREE



传祺 M6



埃安 AION Y



长安 UNI-T



传祺 GS8



传祺 GS4 PLUS



传祺 GS4

陪伴用户行驶 **百亿** 公里里程

从 0 到 1 的突破
从 1 到 N 的开放共赢

地平线征程系列自动驾驶芯片

占据L2+NoA市场**49.05%**的份额，排名**第一**

地平线+英伟达两家一起占据**95%**的市场份额



征程5定点车型10+个

征程5出货量达10万+片

合作车企

理想

BYD

SAIC 上汽集团
SAIC MOTOR

长安汽车
CHANGAN

+其他多家车企

AION

红旗

哪吒汽车

CHERY

目录

contents

1.0

嵌入式智能开发板介绍

2.0

视觉感知原理

3.0

算法模型量化

4.0

计算机视觉应用实践

嵌入式

嵌入式系统是一种专门设计用于特定任务的计算机系统。它通常被嵌入在另一个设备中，以**控制、监测或执行某些特定功能**。嵌入式系统通常包含**一个或多个微处理器或微控制器**，以及与其配套的硬件和软件。嵌入式系统可以用于各种不同的应用领域，例如**汽车、家电、医疗设备、电子玩具、军事设备**等等。

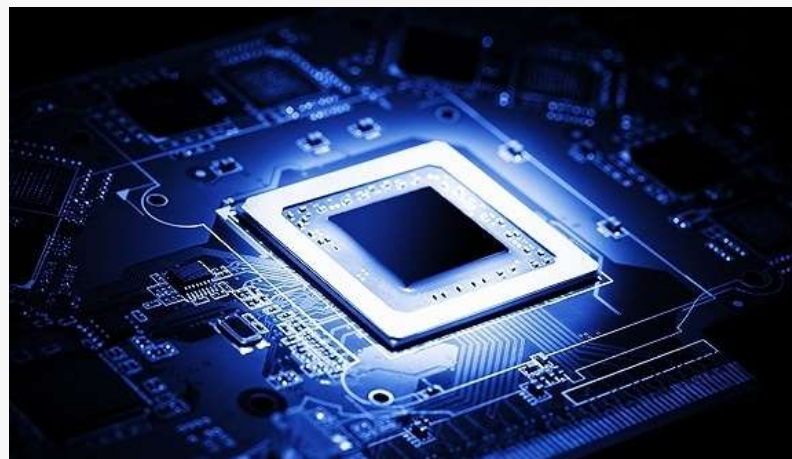
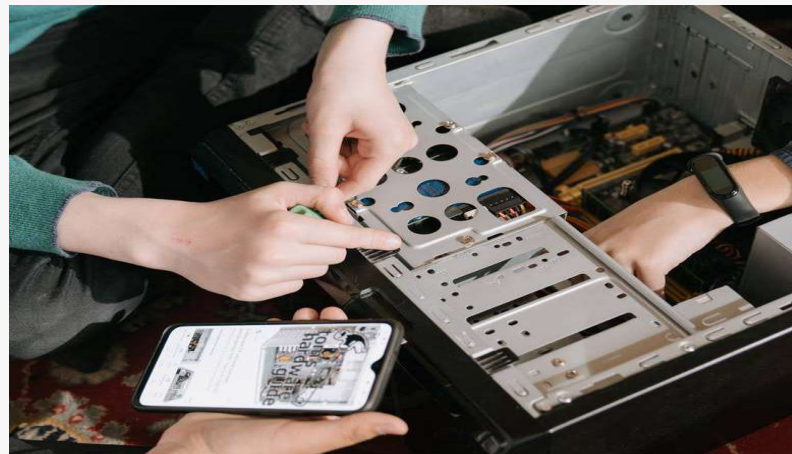
嵌入式系统通常需要满足以下特定要求：

实时性：嵌入式系统通常需要能够在严格的时间限制下运行。

可靠性：由于嵌入式系统通常被嵌入到其他设备中，因此其稳定性和可靠性非常重要。

功耗：嵌入式系统通常需要在低功耗模式下运行，以便延长其电池寿命或降低功耗成本。

成本：嵌入式系统通常需要在成本可控的情况下设计和制造。



边云计算对比

	嵌入式系统	云计算
定义	嵌入在其他设备中的专用计算机系统	基于互联网的分布式计算模型
设计目标	实时性、可靠性、低功耗、成本可控	提高资源利用率、可扩展性、灵活性
实现方式	专用硬件和软件	通用硬件和软件
应用场景	汽车、家电、医疗设备等	数据存储、互联网服务、大数据分析、人工智能等
处理能力	通常较弱，适合处理简单任务	通常较强，适合处理复杂任务
存储容量	通常较小，适合存储少量数据	通常较大，适合存储大量数据
通信能力	通常较弱，适合与本地设备通信	通常较强，适合进行网络通信
可靠性、安全性	- 高，因为需要保证设备的稳定运行和安全性 - 高，因为需要保护设备和数据的安全性	- 高，因为需要保证系统的稳定运行和安全性 - 高，因为需要保护用户数据和隐私
能耗、成本	低，因为需要控制成本和适应设备的价格	高，因为需要建设大型数据中心和购买高端设备

堪比计算机诞生的史诗级创新



个人电脑时代



智能手机时代



云计算时代



智能机器人时代

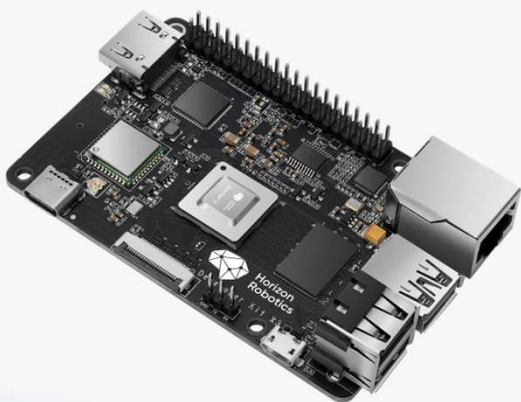
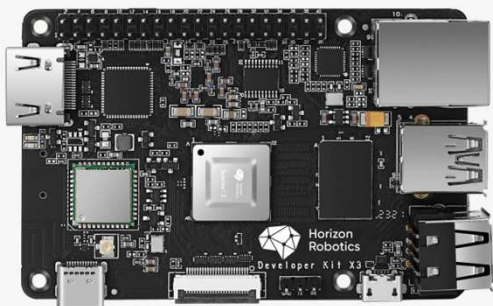
自动驾驶域控制器应用演示



嵌入式产品介绍



嵌入式产品介绍



RDK X3是一款高性能的嵌入式开发板，能够提供出色的计算性能和能效比。它还拥有丰富的接口和扩展槽，支持多种外设和传感器，可以满足机器人、无人驾驶、智能家居等领域的应用需求。

除了高性能和丰富的接口外，RDK X3还配备了**TogetherROS.Bot机器人开发平台**，这是一个专门为机器人开发而设计的开发平台，可以帮助开发者构建高效、可靠的机器人应用程序。通过使用TogetherROS.Bot，开发者可以快速地构建机器人应用程序，并轻松地利用社区中已经构建好的模块和库来简化开发过程。

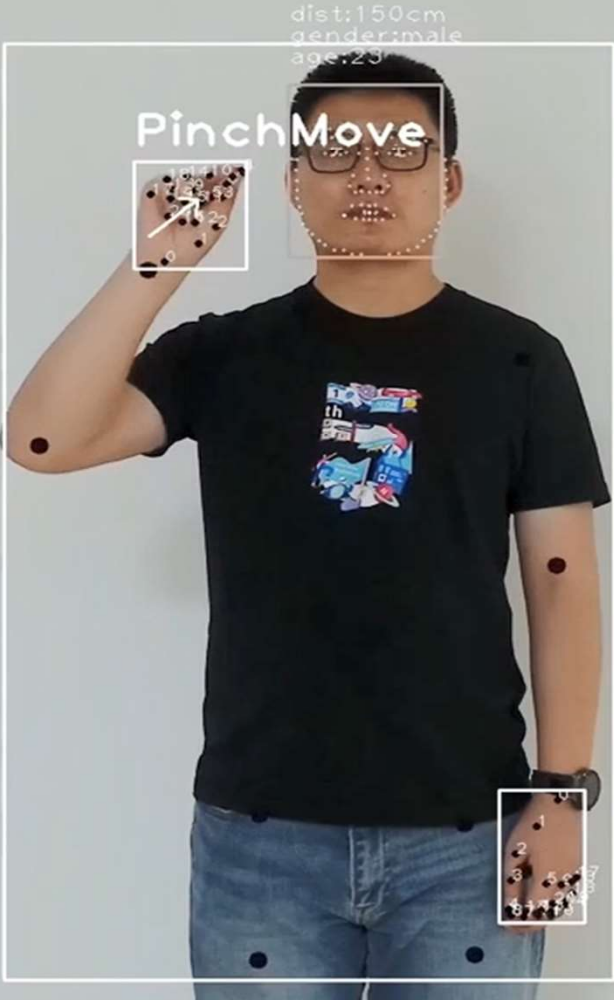
RDK X3在**机器人开发领域**拥有广泛的应用前景，它可以用于开发机器人控制系统、机器人视觉系统、机器人导航系统、机器人智能算法等多个方面，能够让生态开发者在机器人领域中实现更加丰富和高效的创新和发展。

地平线RDK X3



地平线旭日® 3 人机交互解决方案
Horizon sunrise 3 human computer interaction solution

AI ON HORIZON



参数对比



开发板	树莓派4B	Jeston Nano	RDK X3
CPU	4核A72	4核A57	4核A53
内存	2GB/4GB/8GB	2GB/4GB	2GB/4GB
AI引擎	无	CPU/CUDA	BPU
算力	0.2T (FP16)	0.5T (FP16)	5T (INT8)
编解码	1080P@30fps	4K@30fps	4K@60fps
扩展接口	USB/ETH/CSI/HDMI		
功耗	Max 15W	Max 20W	Max 10W
操作系统	Ubuntu 20.04	Ubuntu 18.04	Ubuntu 20.04
机器人扩展框架	ROS	ROS、Isaac SDKROS、Isaac SDK	ROS、TogetherROS

常用开发方法介绍

串口通信

- 按位 (bit) 发送和接收字节
- 波特率**: 单位时间内载波参数变化的次数, 代表传输速率
- 数据位**: 通信中实际数据位的个数, 一般为6、7或8bit
- 停止位**: 用于表示单个包的最后一位。典型的值为1, 1.5和2位
- 奇偶校验位**: 在串口通信中一种简单的检错方式。对于偶和奇校验的情况, 串口会设置校验位 (数据位后面的一位), 用一个值确保传输的数据有偶个或者奇个逻辑高位



SSH安全外壳协议

- 是一种在不安全网络上用于安全远程登录和其他安全网络服务的协议。
- 通过网络的方式访问设备, 与串口通信方式相比拥有更大的带宽和自由度
- 使用方式: `ssh username@IP`
- 举例: `ssh root@192.168.1.10`
- IP 可以通过以下命令看到: `ifconfig`



RDK X3系统配置

Step1

Quick View安装 串口驱动和MobaXterm工具

Step2

将烧写好的SD卡插入RDK X3

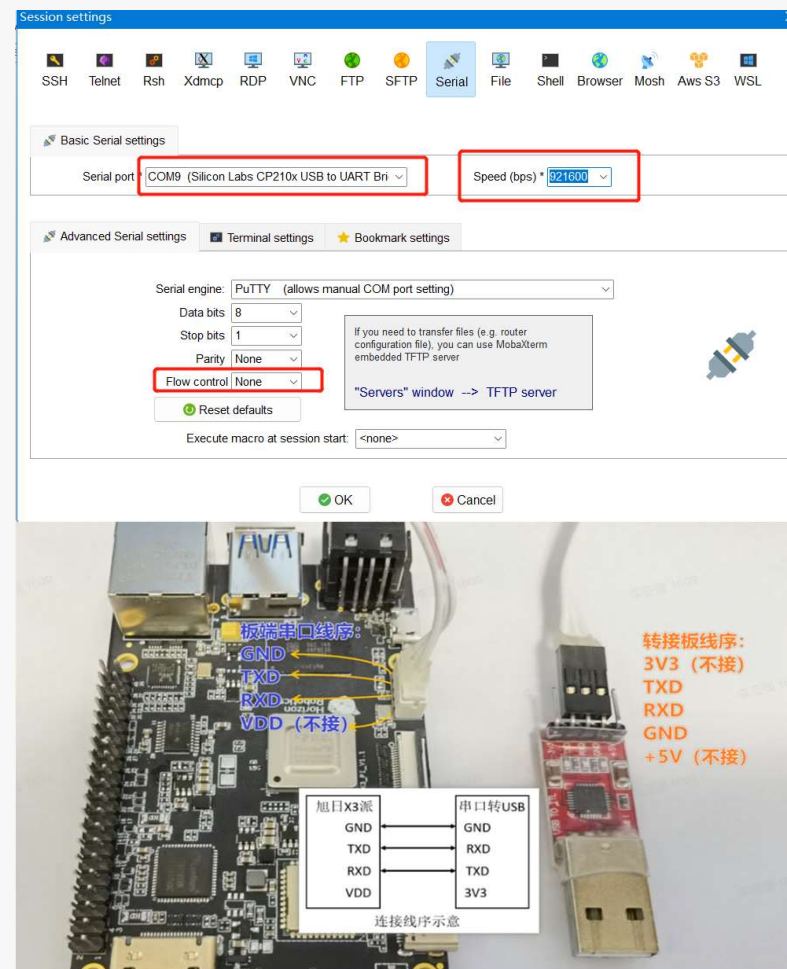
将串口转接板与RDK X3正确连接

Step3

Session->new session

设置MobaXterm串口通信参数

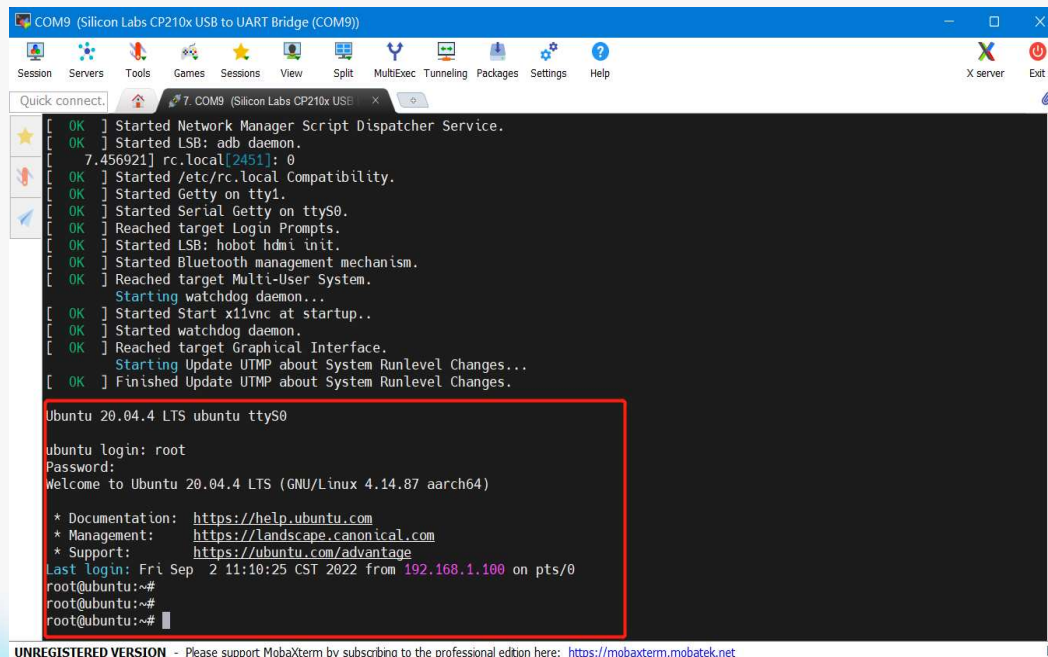
波特率921600



运行环境配置

Step1

输入用户名root



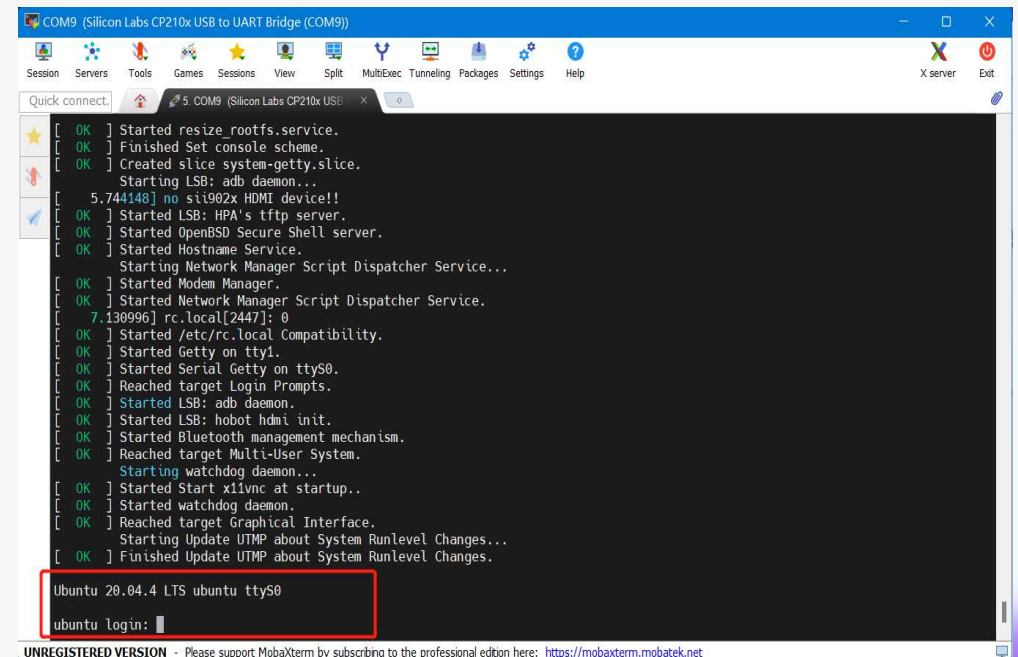
```
COM9 (Silicon Labs CP210x USB to UART Bridge (COM9))
[ OK ] Started Network Manager Script Dispatcher Service.
[ OK ] Started LSB: adb daemon.
[ OK ] 7.456921] rc.local[2451]: 0
[ OK ] Started /etc/rc.local Compatibility.
[ OK ] Started Getty on tty1.
[ OK ] Started Serial Getty on ttyS0.
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: hobot hdmi init.
[ OK ] Started Bluetooth management mechanism.
[ OK ] Reached target Multi-User System.
[ OK ] Starting watchdog daemon...
[ OK ] Started Start x11vnc at startup..
[ OK ] Started watchdog daemon.
[ OK ] Reached target Graphical Interface.
[ OK ] Starting Update UTMP about System Runlevel Changes...
[ OK ] Finished Update UTMP about System Runlevel Changes.

Ubuntu 20.04.4 LTS ubuntu ttyS0
ubuntu login: root
Password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 4.14.87 aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
Last login: Fri Sep  2 11:10:25 CST 2022 from 192.168.1.100 on pts/0
root@ubuntu:~#
root@ubuntu:~#
root@ubuntu:~#
```

Step2

输入密码root



```
COM9 (Silicon Labs CP210x USB to UART Bridge (COM9))
[ OK ] Started resize_rootfs.service.
[ OK ] Finished Set console scheme.
[ OK ] Created slice system-getty.slice.
[ OK ] Starting LSB: adb daemon...
[ OK ] 5.744148] no si902x HDMI device!!
[ OK ] Started LSB: HPA's tftp server.
[ OK ] Started OpenBSD Secure Shell server.
[ OK ] Started Hostname Service.
[ OK ] Starting Network Manager Script Dispatcher Service...
[ OK ] Started Modem Manager.
[ OK ] Started Network Manager Script Dispatcher Service.
[ OK ] 7.130996] rc.local[2447]: 0
[ OK ] Started /etc/rc.local Compatibility.
[ OK ] Started Getty on tty1.
[ OK ] Started Serial Getty on ttyS0.
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: adb daemon.
[ OK ] Started LSB: hobot hdmi init.
[ OK ] Started Bluetooth management mechanism.
[ OK ] Reached target Multi-User System.
[ OK ] Starting watchdog daemon...
[ OK ] Started Start x11vnc at startup..
[ OK ] Started watchdog daemon.
[ OK ] Reached target Graphical Interface.
[ OK ] Starting Update UTMP about System Runlevel Changes...
[ OK ] Finished Update UTMP about System Runlevel Changes.

Ubuntu 20.04.4 LTS ubuntu ttyS0
ubuntu login:
```

利用笔记本创建Wifi热点

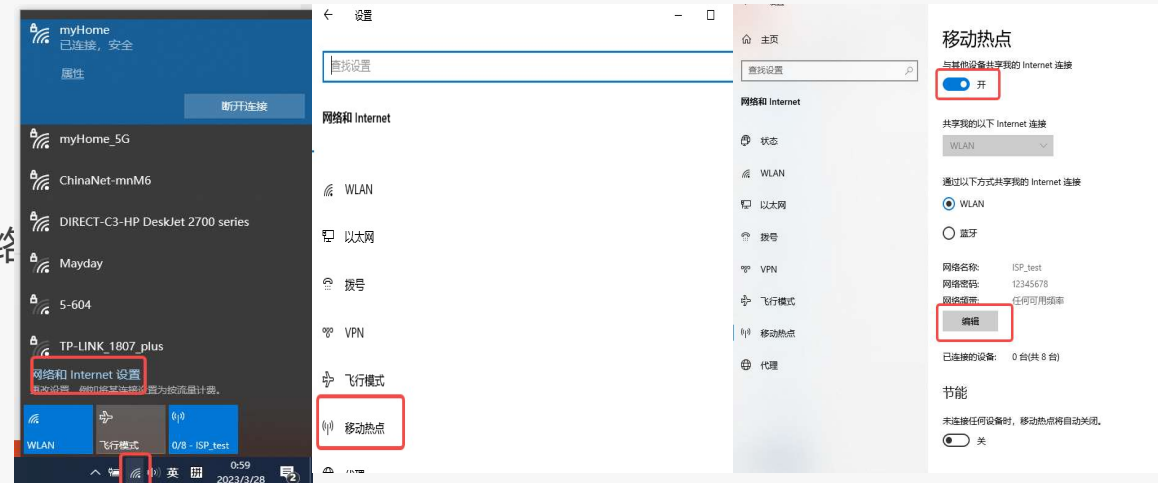
Step1

进行Wifi连接

```
$ sudo nmcli device wifi rescan    # 扫描wifi网络
$ sudo nmcli device wifi list      # 列出找到的wifi网络
$ sudo wifi_connect "SSID" "PASSWD" # 连接某指定的wifi网络
```

Step2

输入ifconfig, 查看IP地址



```
root@ubuntu:~# sudo nmcli device wifi rescan
root@ubuntu:~# sudo nmcli device wifi list
IN-USE BSSID SSID MODE CHAN RATE SIGNAL BARS SECURITY
*      A2:9D:7E:55:0A:AA -- Infra 2 130 Mbit/s 94 -- --
      50:2D:BB:D0:0B:7A midea_ca_0019 Infra 2 65 Mbit/s 82 -- WPA2
      34:FC:A1:9C:A7:AB 602 Infra 1 130 Mbit/s 79 -- WPA1 WPA2
      9C:9D:7E:55:0A:AA XH-Home Infra 2 130 Mbit/s 72 -- WPA1 WPA2
      74:05:A5:93:24:2B D2-501 Infra 11 270 Mbit/s 65 -- WPA1 WPA2
      9C:D8:63:DA:4C:22 HF-LPT130 Infra 6 135 Mbit/s 49 -- --
      DC:FE:18:88:30:1B THINK-Network Infra 11 405 Mbit/s 37 -- WPA1 WPA2
      FC:7C:02:40:FD:B7 quer770503 Infra 3 270 Mbit/s 29 -- WPA1 WPA2
      C8:8F:26:19:DC:4F Topway_19DC4F Infra 1 130 Mbit/s 22 -- WPA2

root@ubuntu:~# sudo wifi_connect "" ""
Device 'wlan0' successfully activated with '4ea86192-91fa-4cd0-bdd7-ae08ff69c1d7'.
```

通过SSH访问RDK X3

Step1

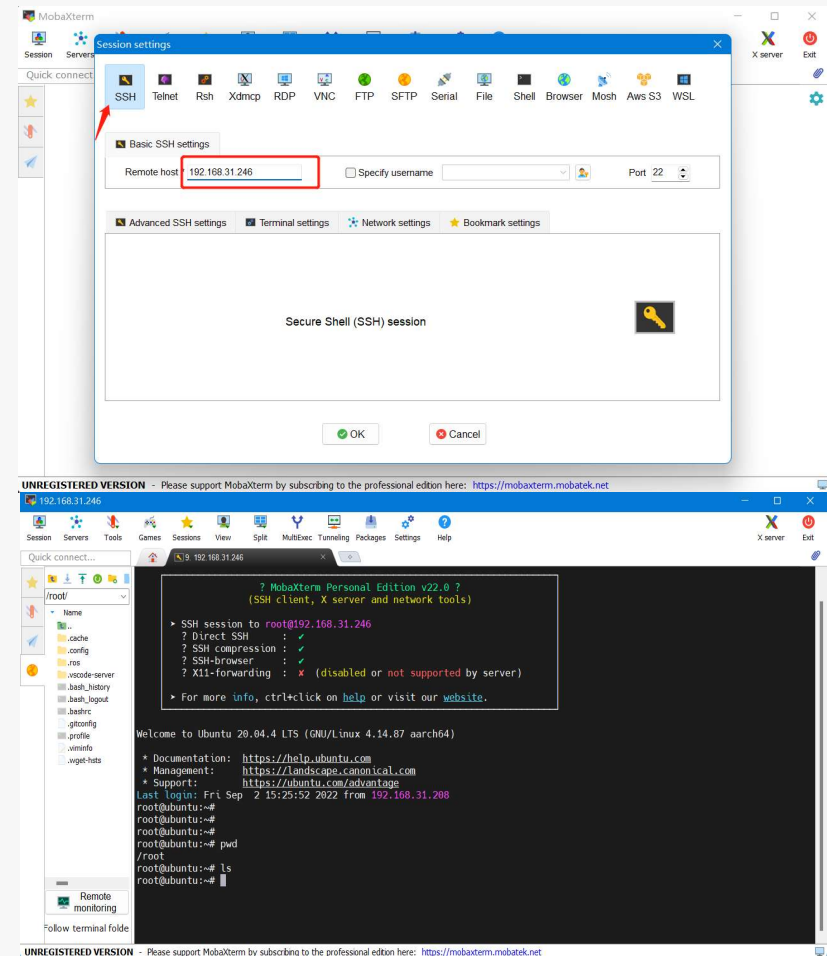
配置mobaxterm, 填入RDK X3的IP地址

Step2

用户名与密码均输入root

Step3

输入 top 指令, 查看系统状态



VNC 连接

Step1

确认开发板IP地址

可通过串口登录、SSH登录设备后
使用ifconfig命令查询

Step2

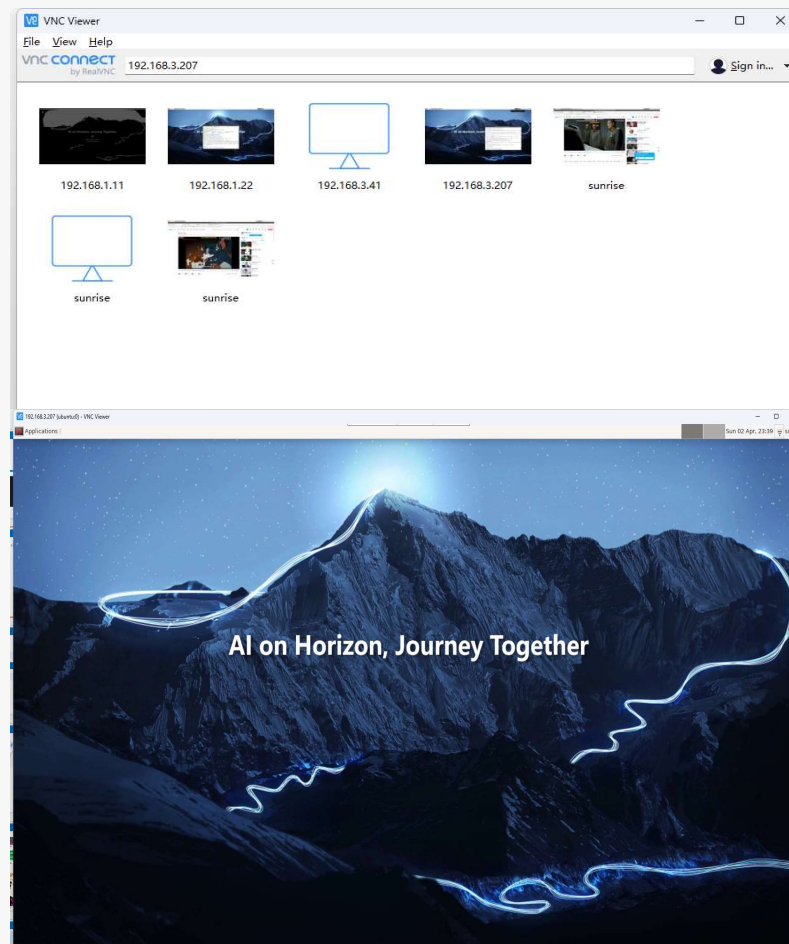
安装VNC软件

Step3

打开VNC Viewer

输入开发板IP地址和VNC登录密码

默认密码为sunrise



目录

contents

1.0

嵌入式智能开发板介绍

2.0

视觉感知原理

3.0

算法模型量化

4.0

计算机视觉应用实践

人工智能演变史

生成式AI

- 根据提示或现有数据创建新的书面、视觉和听觉内容

深度学习

- 一种使用多层神经网络处理数据并做出决策的机器学习技术

机器学习

- 人工智能的子集，使机器能够从现有数据中学习，并改进该数据以做出决策或预测

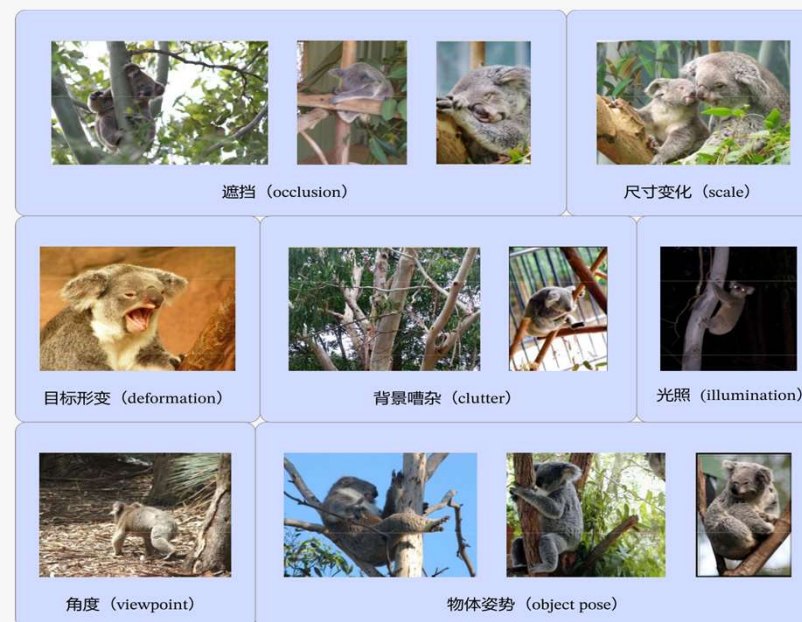
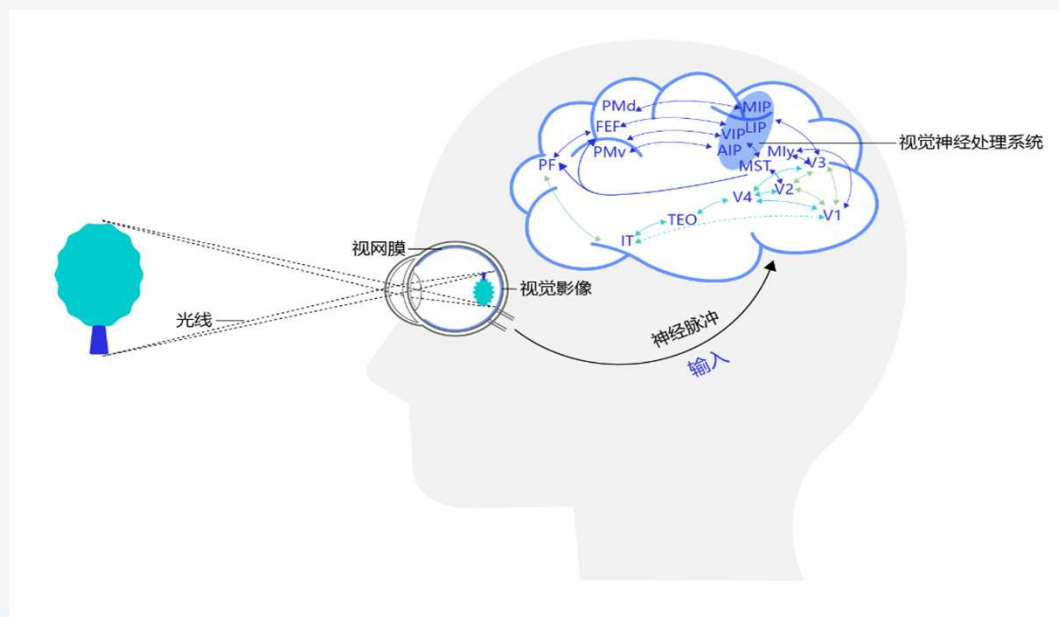
人工智能

- 计算机科学领域，旨在创造能够复制或超越人类智能的智能机器

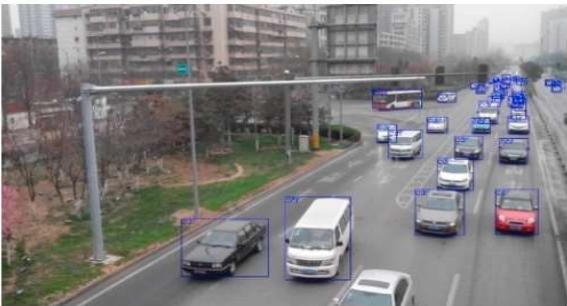
计算机视觉

计算机视觉是一种人工智能领域，旨在使计算机能够像人一样理解和解释视觉信息。计算机视觉的研究主要关注**如何使计算机理解和处理数字图像和视频，并从中提取有用的信息。**

计算机视觉涉及到很多不同的任务，包括图像分类、目标检测、图像分割、姿态估计、人脸识别、运动跟踪等。



视觉任务



图像分类
What

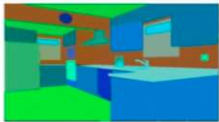
目标检测
What + Where



人脸检测



行人检测



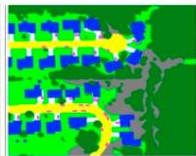
生活场景分割



城市街景分割

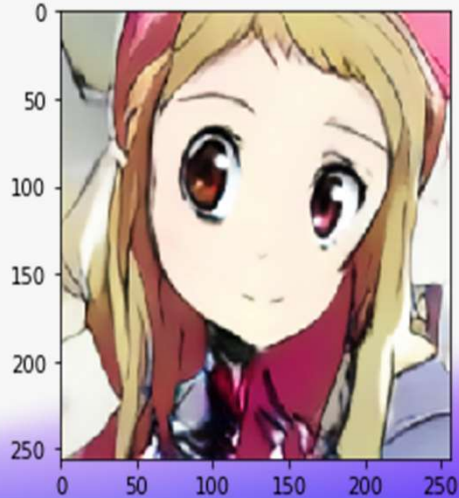
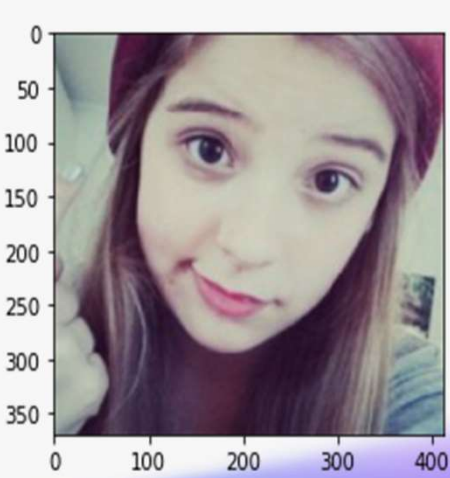


人体部件分割



遥感地块分割

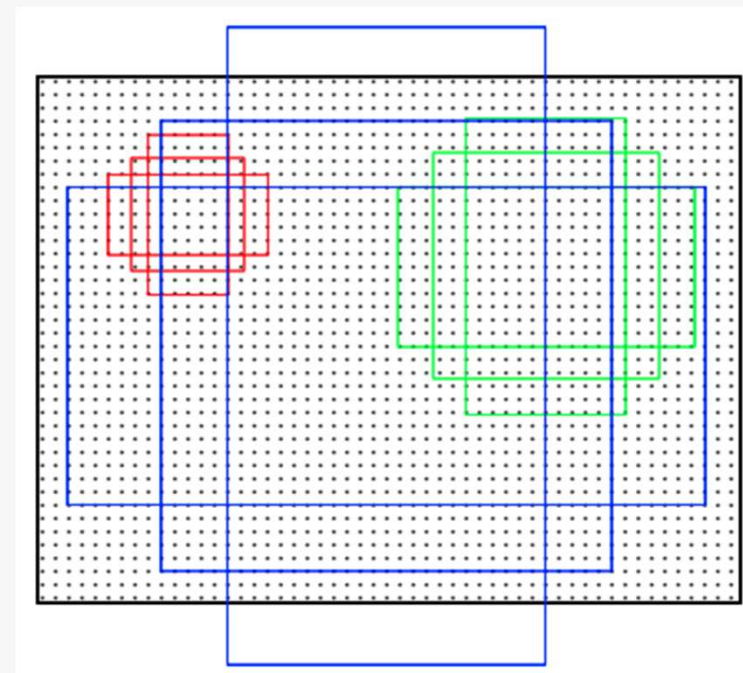
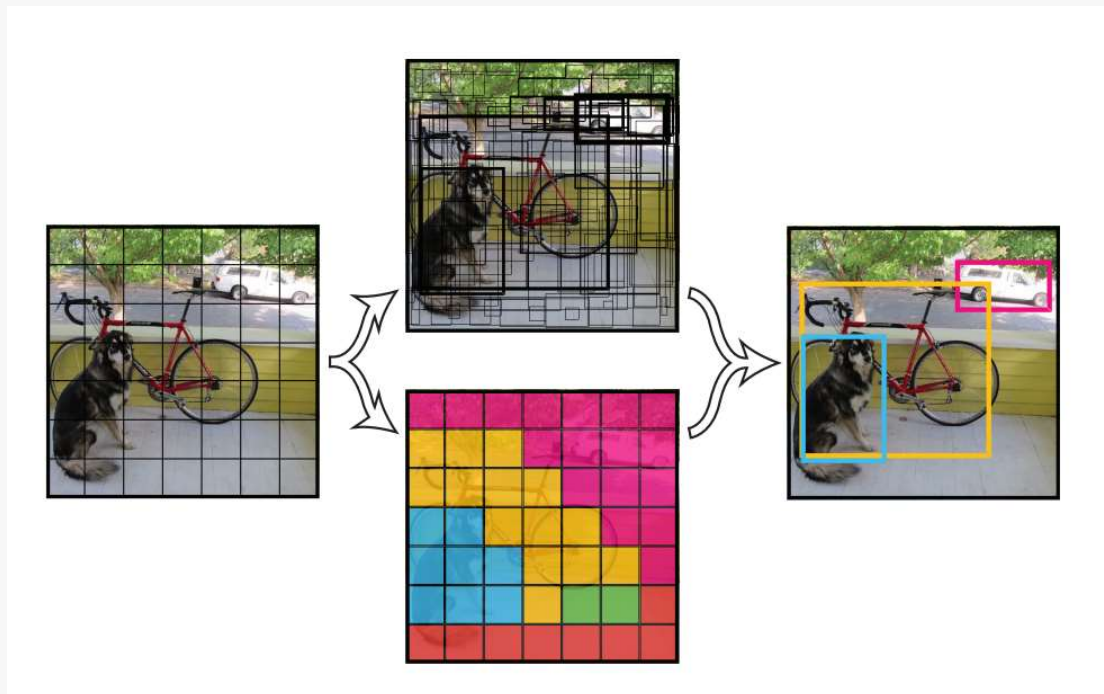
图像分类	将一个输入的图像分类为预先定义好的一类
目标检测	在图像中检测出特定物体的位置和类别
图像分割	将图像分割成若干个子区域，并对每个子区域进行分类
姿态估计	估计图像中物体的位置、方向和姿态
人脸识别	识别图像或视频中的人脸，并进行身份识别
运动跟踪	追踪视频中的物体或人员，并确定它们的运动轨迹
图像生成	生成新的图像或视频，例如通过GAN（生成对抗网络）生成写实的人脸、场景等



大模型下视觉任务

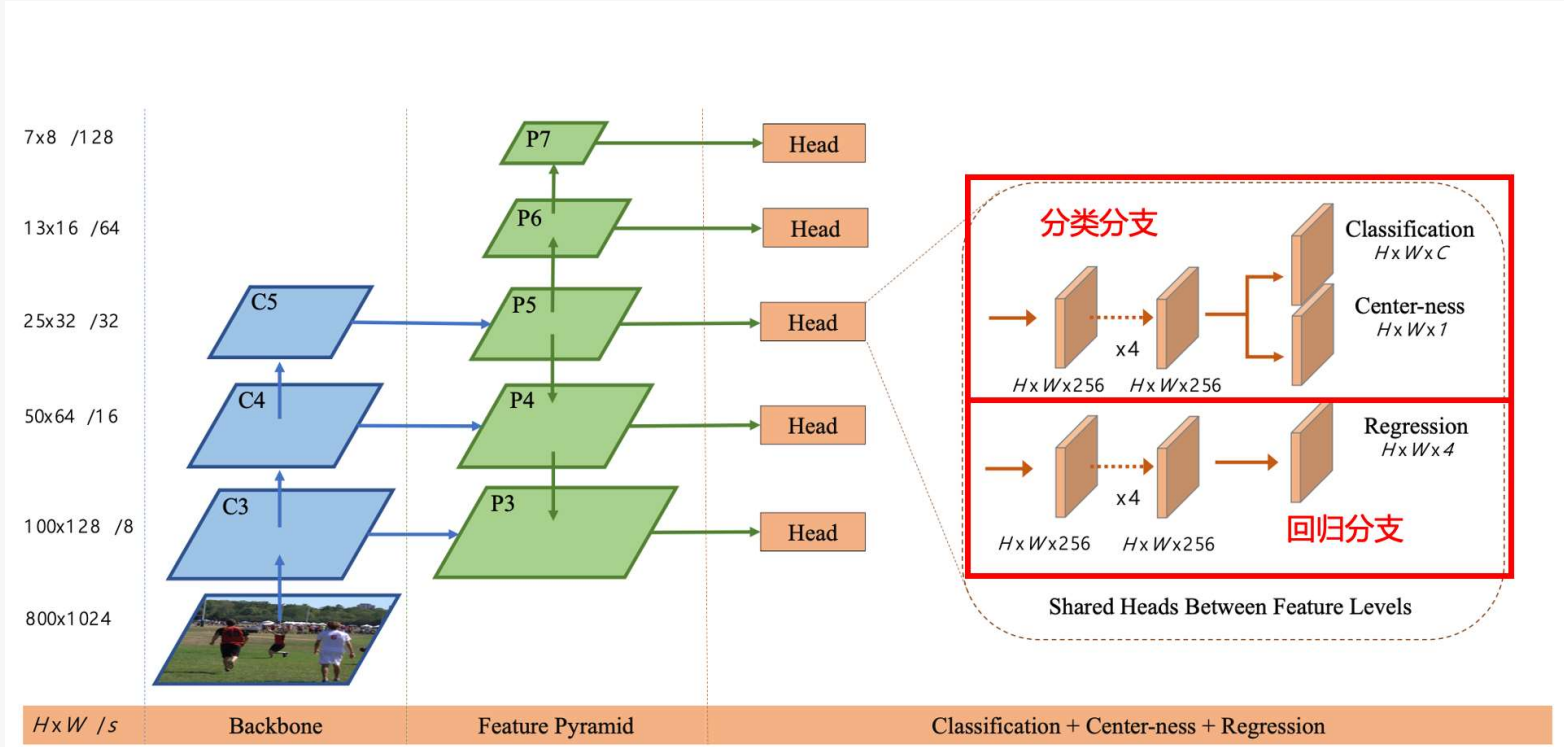


基于Anchor的目标检测



anchor boxes是指预定义的框集合，其宽度和高度被选择以匹配数据集中目标的宽度和高度
anchor boxes包含可以在数据集中找到的目标大小的可能组合，其中包括数据中存在的不同纵横比和比例。

Fcos算法架构



目录

contents

1.0

嵌入式智能开发板介绍

2.0

视觉感知原理

3.0

算法模型量化

4.0

计算机视觉应用实践

摩尔定律

强调传统性能指标

Power

+

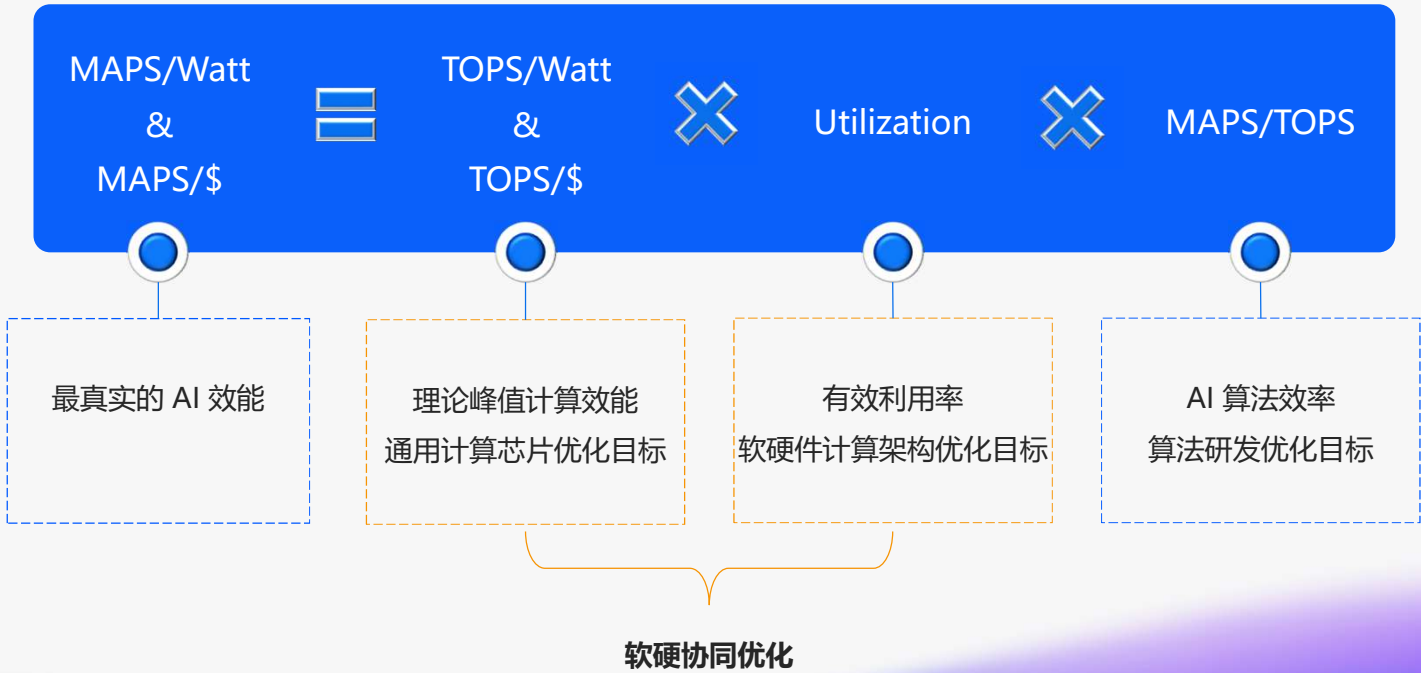
Performance

+

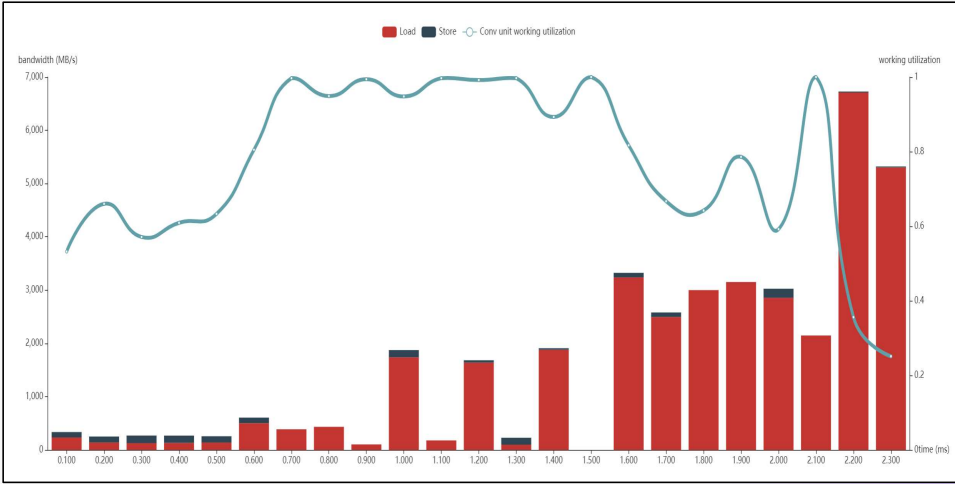
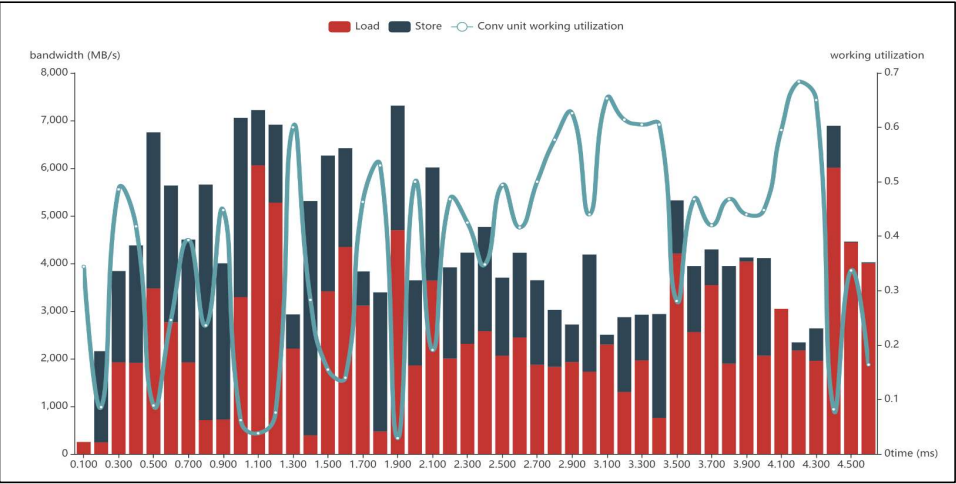
Area

AI 芯片的“新” 摩尔定律

聚焦 AI 计算的能效



	编译器优化前	编译器优化后
BPU Core 1 / Core 2 (fps)	218 / 437	439 / 878 (2x improvement)
Latency	4.6ms	2.3ms (50% reduction)
BPU working/effective utilization	39% / 33%	77% / 67% (2x improvement)
DDR	4.5 GB/s	1.8 GB/s (60% reduction)



全局层
拆分和融合

利用
片上存储器

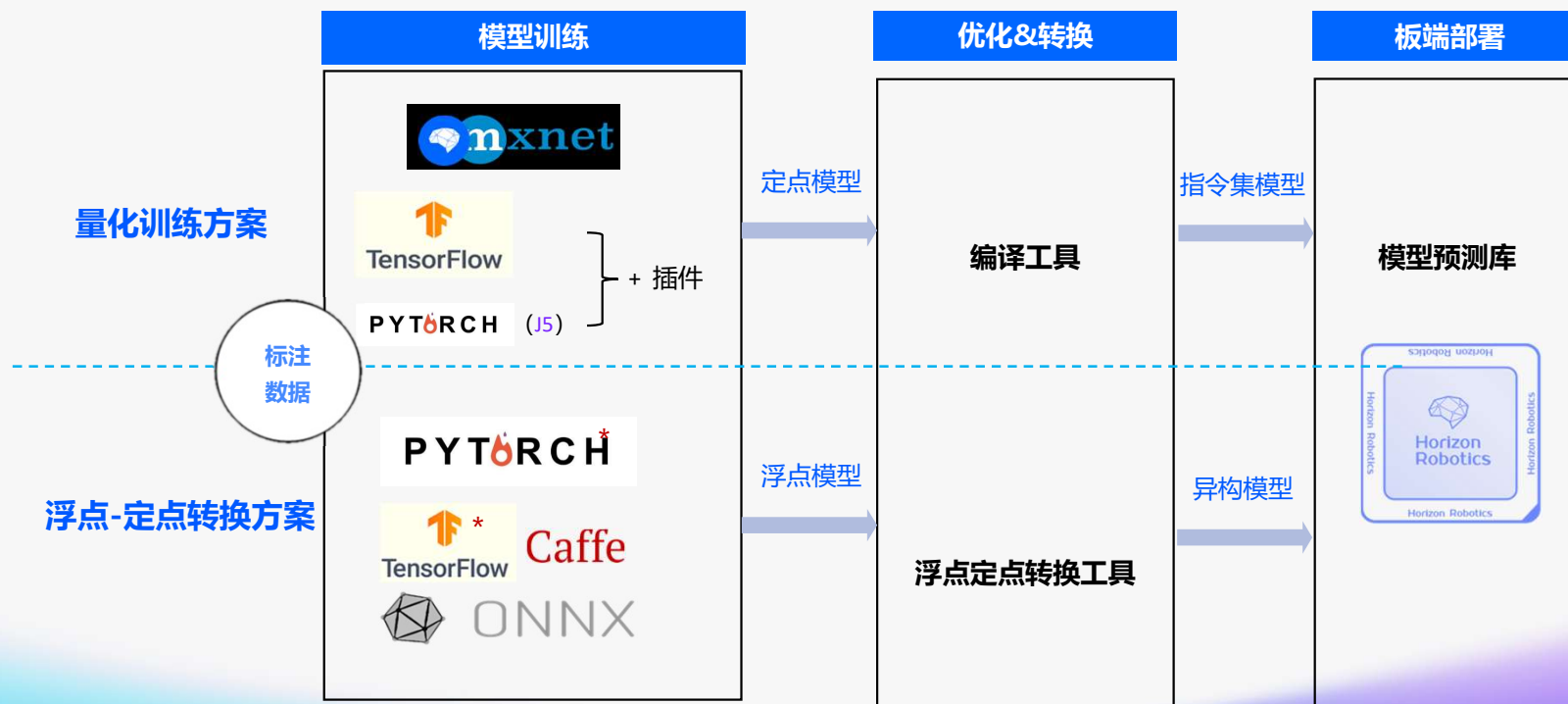
DDR 访问和计算的
更多并行

ISA/架构支持编译
器优化

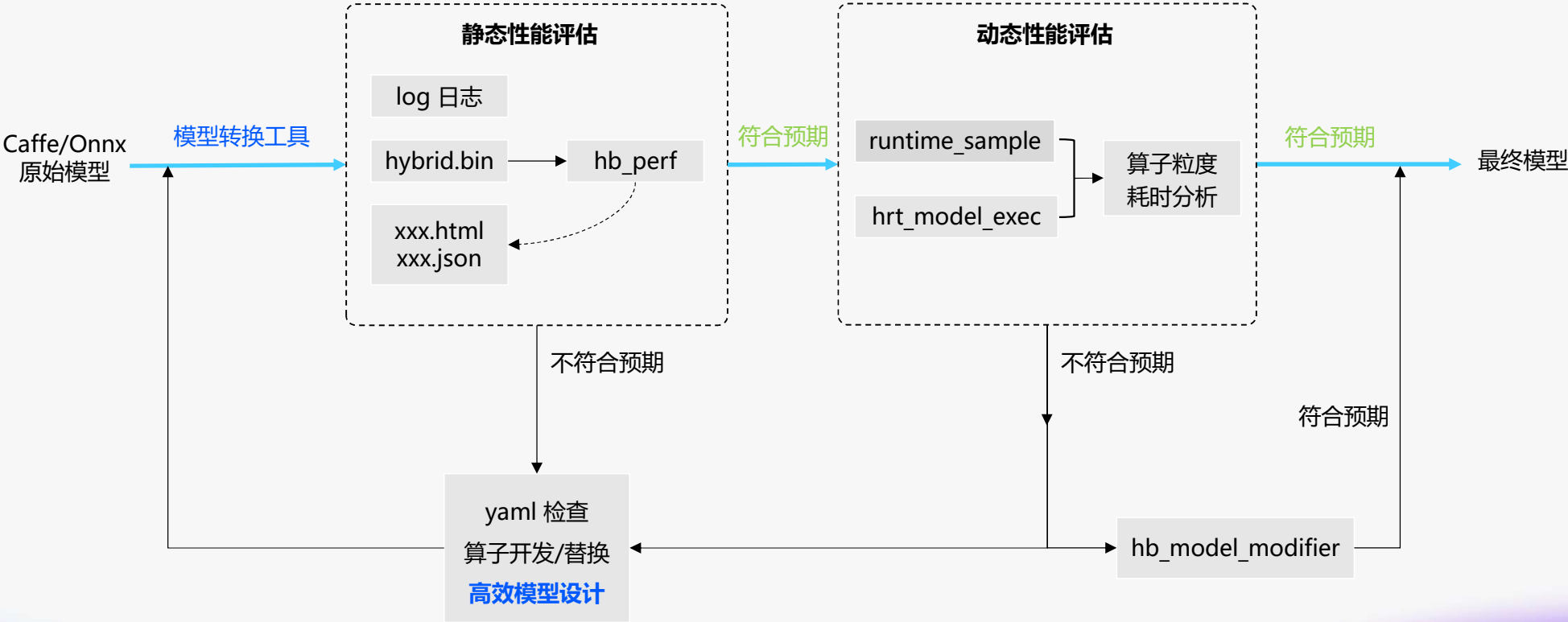
算法工具链

模型量化：是一种深度神经网络的性能优化技术，通过降低权重和激活的精度来减小存储&计算量，进而实现芯片级别的计算加速。常见的有 16bit, 8bit, 4bit 等量化方案，目前地平线 X2/J2、X3/J3 系列均采用 8bit 量化方案

地平线量化目标： 8-bit 量化精度 > 浮点模型精度 * 99%



浮点定点转换工具链工具链



目录

contents

1.0

嵌入式智能开发板介绍

2.0

视觉感知原理

3.0

算法模型量化

4.0

计算机视觉应用实践

全平台硬件方案：XJ3常见视频通路



目标检测效果展示

Step1

在使用 web 展示之前
确保板子与个人PC机处于同一网段
并且PC可以ping通板子的IP

Step2

进入文件夹并启动程序:

```
cd /app/pydev_demo/05_web_display_camera_sample  
python3 mipi_camera.py
```

Step3

浏览器输入ip地址, 观测实施效果
最终呈现效果如图所示
在画面中识别到了tv, cat 和 cup



人体检测与跟踪 环境准备



类别	说明	数据类型
body	人体框	Roi
head	人头框	Roi
face	人脸框	Roi
hand	人手框	Roi
body_kps	人体关键点	Point

板端：

```
# 配置tros.b环境
source /opt/tros/setup.bash

# 从tros.b的安装路径中拷贝出运行示例需要的配置文件。

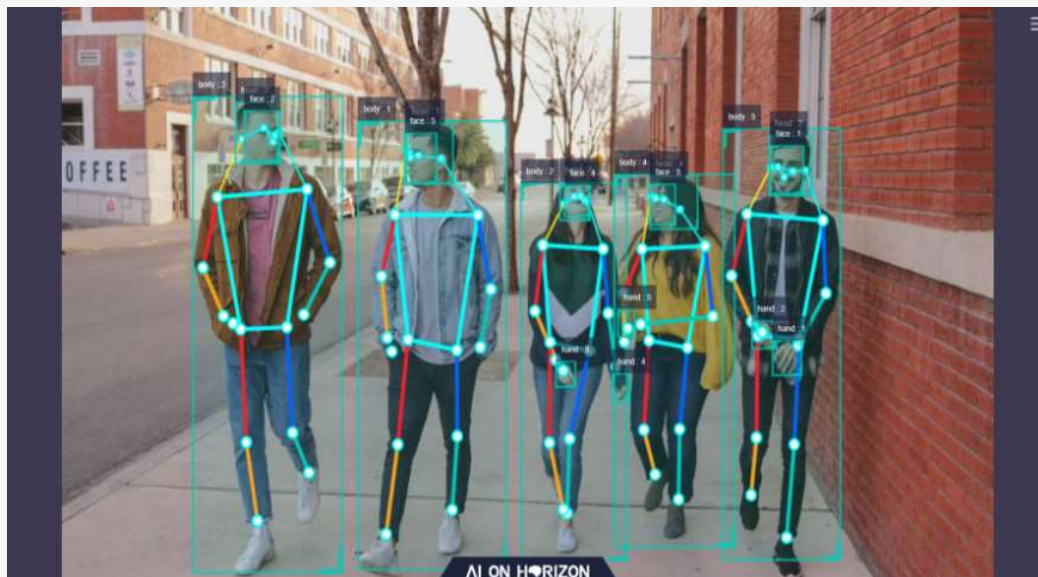
cp -r /opt/tros/lib/mono2d_body_detection/config/ .

# 配置MIPI摄像头
export CAM_TYPE=mipi

# 启动launch文件
ros2 launch mono2d_body_detection mono2d_body_detection.launch.py

#设备上已经有运行http的服务，需要找到并结束掉占用80端口的进程，执行命令lssof -i:80，得到占用端口的进程信息和它的PID，并用kill -9 PID结束掉进程即可
```

人体检测与跟踪 效果展示



输出log显示:
推理时算法输入和输出帧率为30fps
每秒钟刷新一次统计帧率

在PC端的浏览器输入<http://IP:8000>
IP为RDK X3/X86设备的IP地址

即可查看图像和算法渲染效果
人体、人头、人脸、人手检测框，检测框类型和目标跟踪ID，
人体关键点

```
[mono2d_body_detection-3] [WARN] [1680452863.050139756] [mono2d_body_det]: input fps: 31.40, out fps: 31.43, infer time ms: 28, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452864.065081108] [mono2d_body_det]: input fps: 29.67, out fps: 29.62, infer time ms: 29, post process time ms: 2
[mono2d_body_detection-3] [WARN] [1680452865.067722086] [mono2d_body_det]: input fps: 29.79, out fps: 29.94, infer time ms: 25, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452866.079732477] [mono2d_body_det]: input fps: 29.70, out fps: 29.67, infer time ms: 26, post process time ms: 2
[mono2d_body_detection-3] [WARN] [1680452867.083715155] [mono2d_body_det]: input fps: 29.79, out fps: 29.85, infer time ms: 23, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452868.099770944] [mono2d_body_det]: input fps: 29.73, out fps: 29.53, infer time ms: 30, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452869.107019183] [mono2d_body_det]: input fps: 29.73, out fps: 29.79, infer time ms: 28, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452870.117825668] [mono2d_body_det]: input fps: 29.73, out fps: 29.70, infer time ms: 29, post process time ms: 1
[mono2d_body_detection-3] [WARN] [1680452871.123122228] [mono2d_body_det]: input fps: 29.73, out fps: 29.85, infer time ms: 25, post process time ms: 2
[mono2d_body_detection-3] [WARN] [1680452872.134676942] [mono2d_body_det]: input fps: 29.76, out fps: 29.67, infer time ms: 28, post process time ms: 1
```

手势识别 环境准备

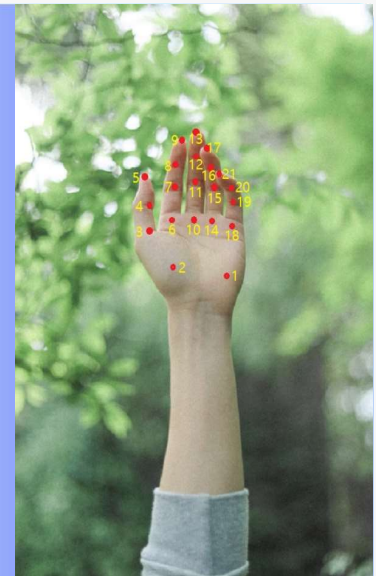
手势	说明	数值
ThumbUp	竖起大拇指	2
Victory	“V” 手势	3
Mute	“嘘” 手势	4
Palm	手掌	5
Okay	OK手势	11
ThumbLeft	大拇指向左	12
ThumbRight	大拇指向右	13
Awesome	666手势	14

```
# 配置tros.b环境  
source /opt/tros/setup.bash
```

```
# 从tros.b的安装路径中拷贝出运行示例需要的配置文件。  
cp -r /opt/tros/lib/mono2d_body_detection/config/ .  
cp -r /opt/tros/lib/hand_lm_k_detection/config/ .  
cp -r /opt/tros/lib/hand_gesture_detection/config/ .
```

```
# 配置MIPI摄像头  
export CAM_TYPE=mipi
```

```
# 启动launch文件  
ros2 launch hand_gesture_detection hand_gesture_detection.launch.py
```



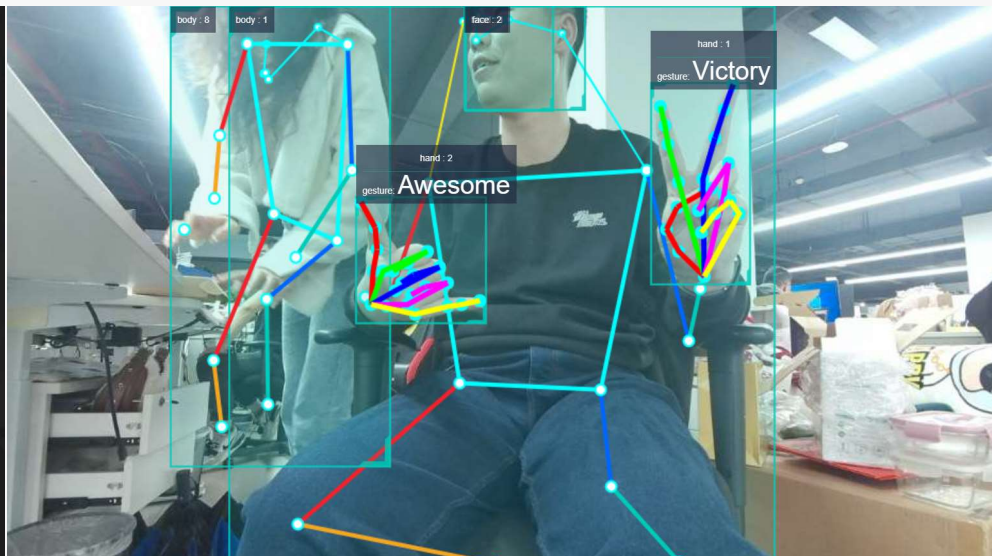
手势识别 效果展示

输出log显示:

订阅到的AI msg中有一个hand（包含人手框和人手关键点检测结果）
手势识别算法输出的手势分类结果是“Awesome”和“Victory”手势

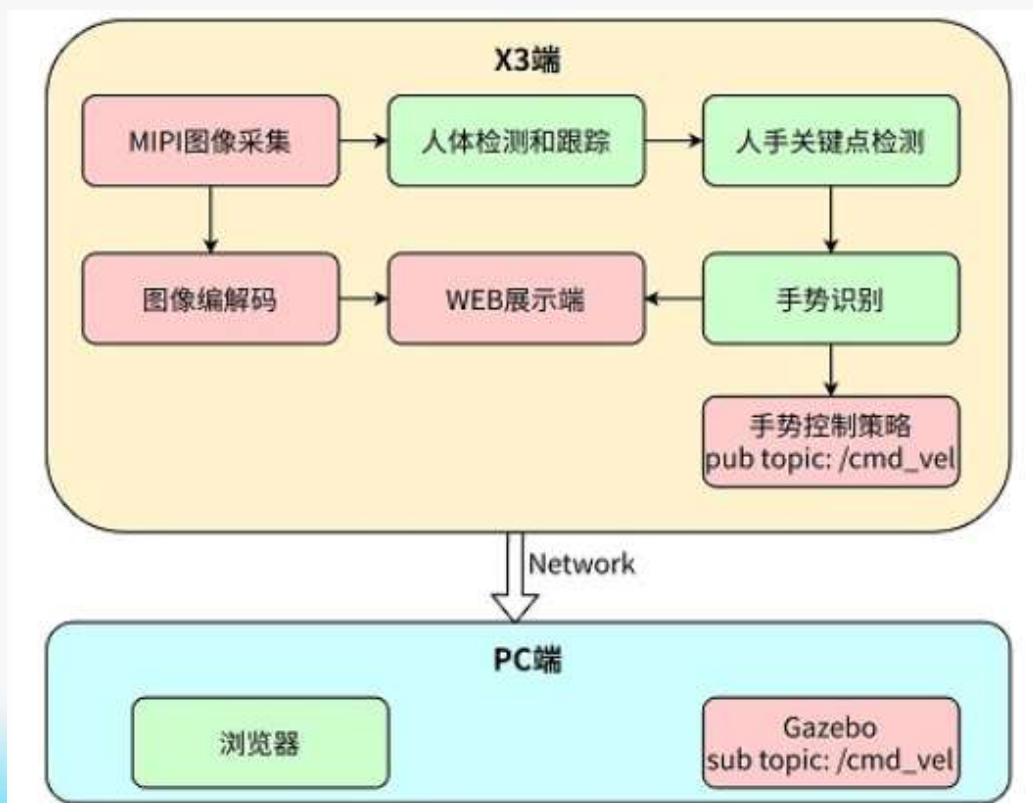
在PC端的浏览器输入<http://IP:8000>
即可查看图像和算法渲染效果
(IP为RDK X3的IP地址)

```
[hand_gesture_detection-5] [WARN] [1680452769.75083364] [hand_gesture_det]: Pub smart fps 29.35
[hand_gesture_detection-5] [WARN] [1680452769.941819523] [hand_gesture_det]: Sub smart fps 29.67
[hand_gesture_detection-5] [WARN] [1680452770.755938841] [hand_gesture_det]: Pub smart fps 29.85
[hand_gesture_detection-5] [WARN] [1680452770.971898212] [hand_gesture_det]: Sub smart fps 29.15
[hand_gesture_detection-5] [WARN] [1680452771.760707229] [hand_gesture_det]: Pub smart fps 29.88
[hand_gesture_detection-5] [WARN] [1680452771.995558484] [hand_gesture_det]: Sub smart fps 30.30
[hand_gesture_detection-5] [WARN] [1680452772.780138824] [hand_gesture_det]: Pub smart fps 29.44
[hand_gesture_detection-5] [WARN] [1680452773.008837894] [hand_gesture_det]: Sub smart fps 29.64
[hand_gesture_detection-5] [WARN] [1680452773.787307574] [hand_gesture_det]: Pub smart fps 29.82
[hand_gesture_detection-5] [WARN] [1680452774.030596726] [hand_gesture_det]: Sub smart fps 29.41
[hand_gesture_detection-5] [WARN] [1680452774.789626527] [hand_gesture_det]: Pub smart fps 29.94
[hand_gesture_detection-5] [WARN] [1680452775.075454142] [hand_gesture_det]: Sub smart fps 29.69
[hand_gesture_detection-5] [WARN] [1680452775.799673656] [hand_gesture_det]: Pub smart fps 29.73
[hand_gesture_detection-5] [WARN] [1680452776.099897370] [hand_gesture_det]: Sub smart fps 30.27
[hand_gesture_detection-5] [WARN] [1680452776.832475550] [hand_gesture_det]: Pub smart fps 29.07
[hand_gesture_detection-5] [WARN] [1680452777.127179545] [hand_gesture_det]: Sub smart fps 29.24
[hand_gesture_detection-5] [WARN] [1680452777.736322720] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.752082854] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.801274816] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.834296875] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.838348027] [hand_gesture_det]: Pub smart fps 29.85
[hand_gesture_detection-5] [WARN] [1680452777.844347989] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.882182326] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452777.913790080] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452778.083261275] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452778.111932908] [preprocess]: target id: 35 rois.size: 1 is unmatched with lmkses.size: 0
[hand_gesture_detection-5] [WARN] [1680452778.153598328] [hand_gesture_det]: Sub smart fps 30.21
```



手势识别 控制仿真机器人

小车手势控制APP功能为通过手势控制机器人小车运动，包括左右旋转和前后平移运动
APP由MIPI图像采集、人体检测和跟踪、人手关键点检测、手势识别、手势控制策略、图像编解码、WEB展示端组成，流程如下图：



666手势/Awesome —— 前进：



大拇指向右/ThumbRight —— 右转：



yeah/Victory —— 后退：



大拇指向左/ThumbLeft —— 左转：



手势识别控制仿真机器人

手势识别 控制仿真机器人

VNC:

```
sudo su
```

```
# 配置TogetherROS环境
```

```
source /opt/tros/setup.bash
```

```
# 启动小乌龟环境
```

```
ros2 run turtlesim turtlesim_node
```

板端:

```
# 配置tros.b环境
```

```
source /opt/tros/setup.bash
```

```
# 从tros.b的安装路径中拷贝出运行示例需要的配置文件。
```

```
cp -r /opt/tros/lib/mono2d_body_detection/config/ .
```

```
cp -r /opt/tros/lib/hand_lm_k_detection/config/ .
```

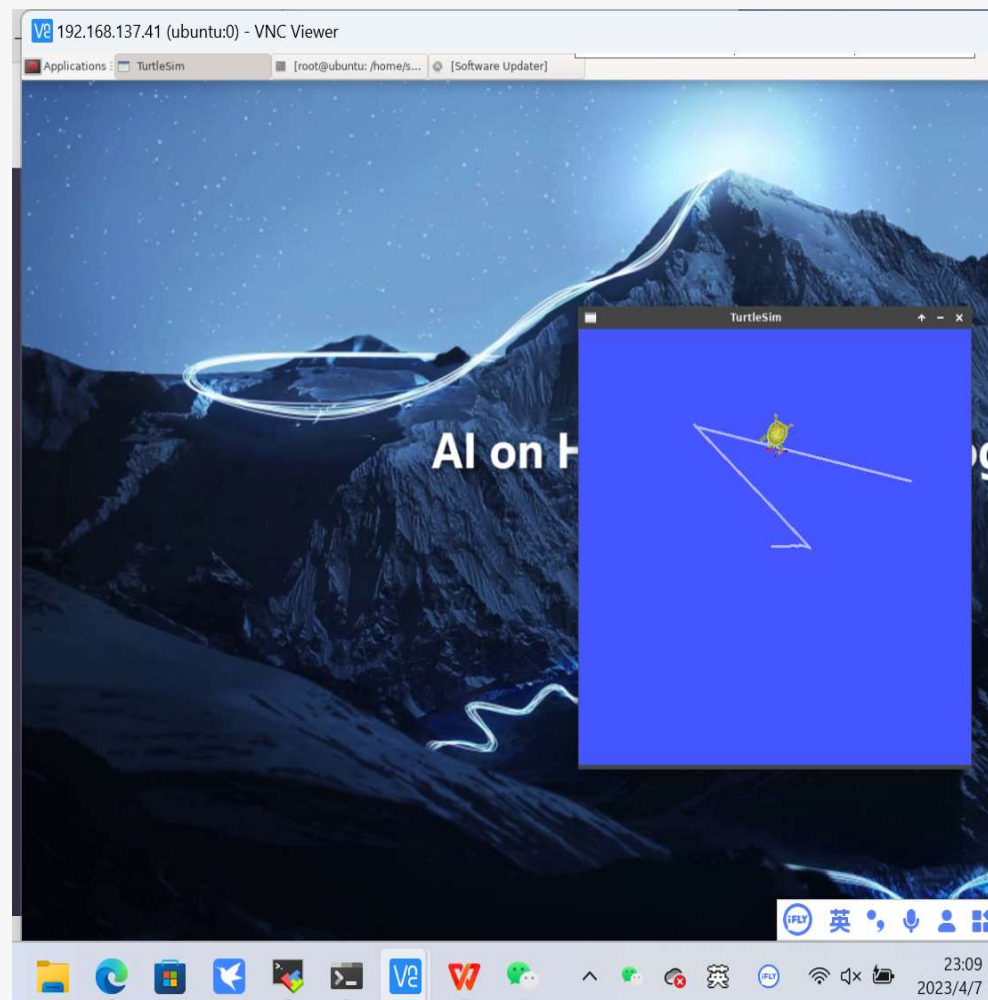
```
cp -r /opt/tros/lib/hand_gesture_detection/config/ .
```

```
# 配置MIPI摄像头
```

```
export CAM_TYPE=mipi
```

```
# 运行launch文件
```

```
ros2 launch gesture_control gesture_control.launch.py
```

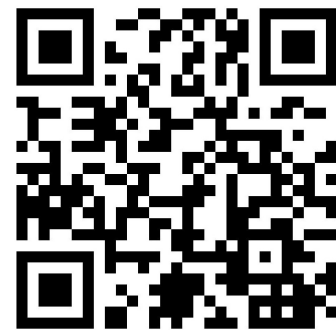




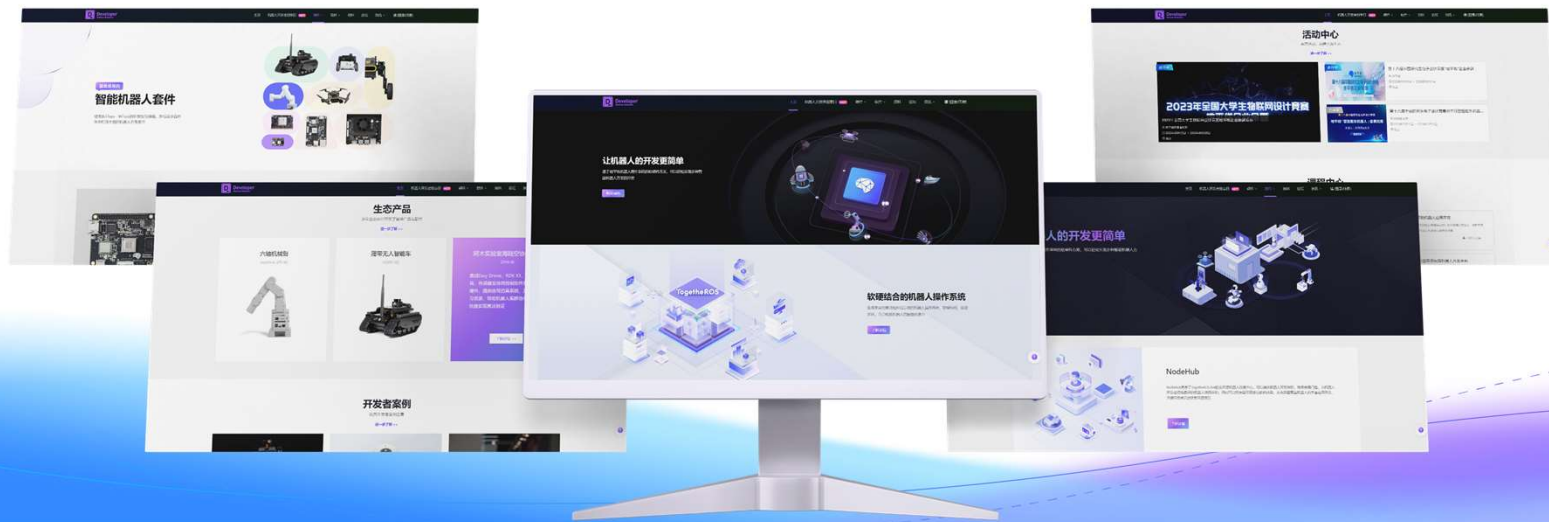
地平线开发者号



地平线开发者社区



大学生问卷调查





地平线
Horizon Robotics