



手把手教你玩转MP4人脸识别

2025年11月

提升社会运行效率 改善人类生活方式

Transforming Tomorrow - Smarter Efficiency Better Lives



一、硬件环境



- 1、CanMV K230 开发板
- 2、电脑 (Windows or MacOS or Ubuntu)
- 3、SD 卡 (建议京东自营或立创商城购买)





二、软件环境



1、CanMV IDE

下载地址：<https://www.kendryte.com/zh/resource/ide,k230>

根据你的电脑使用的系统下载对应版本

资料分类

硬件

芯片手册

参考设计

硬件配置工具

杂项

软件

系统镜像

工具链

杂项

工具

IDE工具

烧录工具

NNCASE

AI Cube

云训练平台数据集

杂项

IDE工具

CanMV IDE K230 for macOS(15.0+)

版本: 4.0.9
2025-07-14

下载

CanMV IDE K230 for Ubuntu(22.04+)

版本: 4.0.9
2025-07-14

下载

CanMV IDE K230 for windows

版本: 4.0.9
2025-07-14

下载



2、最新官网firmware

下载地址：<https://www.kendryte.com/zh/resource/images,k230>

资料分类

硬件

芯片手册

参考设计

硬件配置工具

杂项

软件

系统镜像

工具链

杂项

工具

IDE工具

烧录工具

NNCASE

AI Cube

云训练平台数据集

杂项

系统镜像

版本：

选择版本

Linux

RTOS

Micropython

Linux+RTOS

Micropython

Micropython Daily Build

2033-09-23

查看

Linux+RTOS

EVB-K230_sdcard_v2.0_nncase_v2.10.0.img.gz

版本: 2.0
2025-09-15

下载

Linux+RTOS

CanMV-K230_sdcard_v2.0_nncase_v2.10.0.img.gz

版本: 2.0
2025-09-15

下载

Micropython

CanMV_K230_V3P0_micropython_v1.4-0-g6cce59c_nncase_v2.9.0.img.gz

版本: 1.4
2025-09-09

下载

Micropython

CanMV_K230_V1P0_P1_micropython_v1.4-0-g6cce59c_nncase_v2.9.0.img.gz

版本: 1.4
2025-09-09

下载

Micropython

CanMV_K230_LCKFB_micropython_v1.4-0-g6cce59c_nncase_v2.9.0.img.gz

版本: 1.4
2025-09-09

下载

Micropython

CanMV_K230_01Studio_micropython_v1.4-0-g6cce59c_nncase_v2.9.0.img.gz

版本: 1.4
2025-09-09

下载

Micropython

CanMV-K230_RTT_EVB_BOARD_micropython_v1.4-0-g6cce59c_nncase_v2.9.0.img.gz

版本: 1.4
2025-09-09

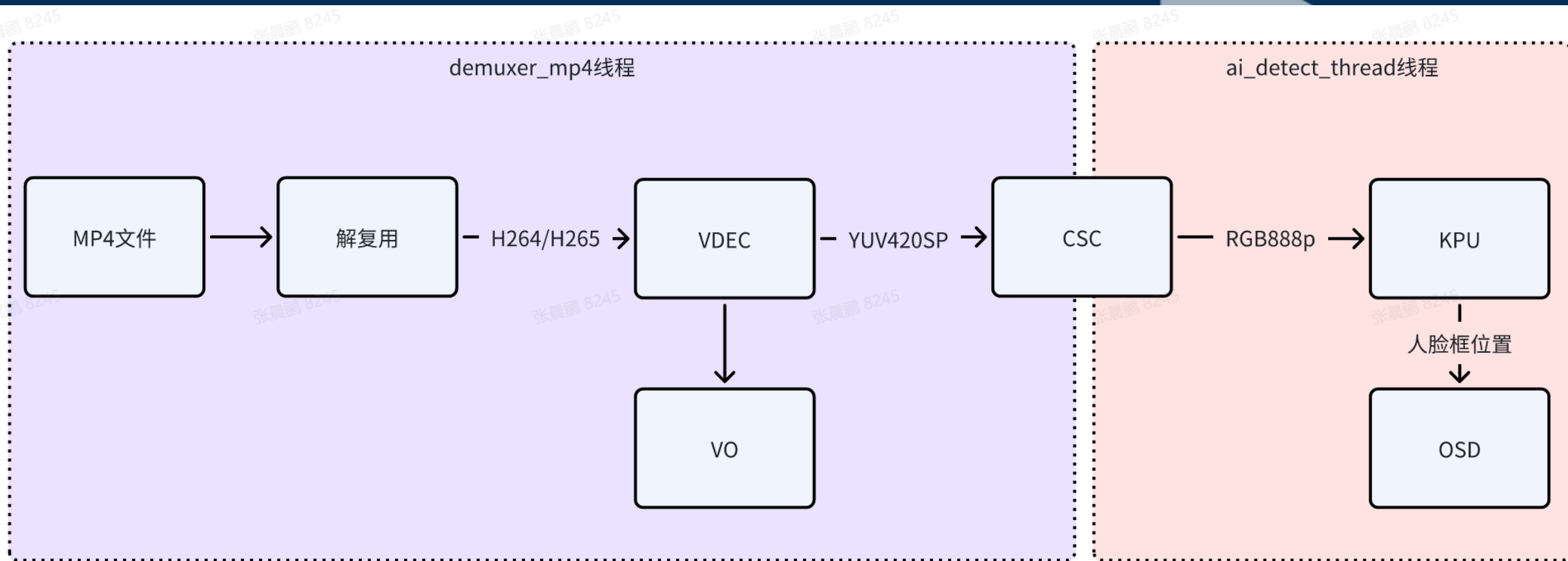
下载



三、代码讲解

整体流程图

Demo路径: \CanMV\sdcard\examples\21-AI-With-Others\face_detect_yunet_from_mp4.py





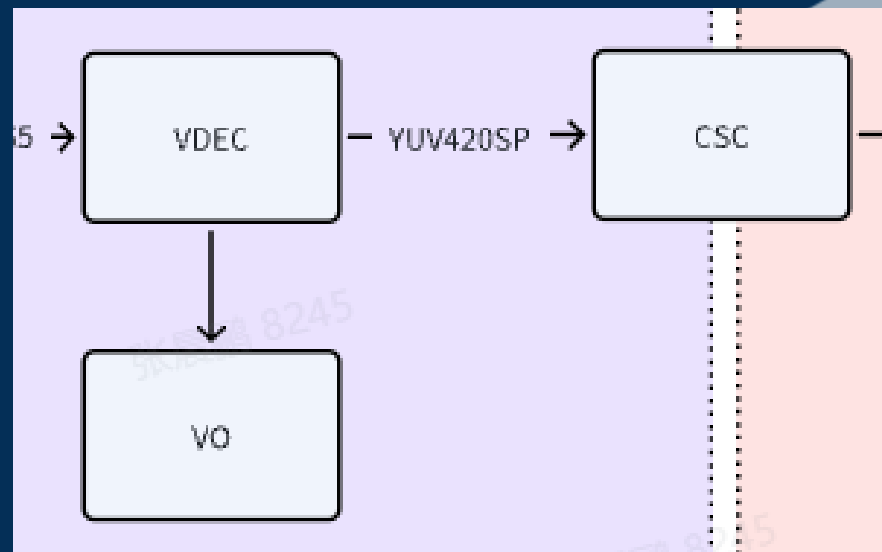
2、各个模块接口参考文档

https://www.kendryte.com/k230_canmv/zh/main/zh/api/index.html

K230 提供 2 层视频图层支持和 4 层 OSD 图层支持。分列如下：

显示层	说明	备注
LAYER_VIDEO1		仅可在 bind_layer 中使用,支持硬件旋转
LAYER_VIDEO2		仅可在 bind_layer 中使用,不支持硬件旋转
LAYER_OSD0		支持 show_image 和 bind_layer 使用
LAYER_OSD1		支持 show_image 和 bind_layer 使用
LAYER_OSD2		支持 show_image 和 bind_layer 使用
LAYER_OSD3		支持 show_image 和 bind_layer 使用

3. 各个模块之间连接最好使用bind模式



```
bind_info = vdec.bind_info(width=video_info.width, height=video_info.height, chn=vdec.get_vdec_channel())
Display.bind_layer(**bind_info, layer = Display.LAYER_VIDEO1)
```

```
vdec_link = MediaManager.link((VIDEO_DECODE_MOD_ID, VDEC_DEV_ID, vdec.get_vdec_channel()),
(NONAI_2D_CSC_MOD_ID, 0, 2))
```

4.注意各个模块需要的数据格式

```
vf = vf_info.v_frame  
img = vf.to_image()  
img_np_hwc = img.to_numpy_ref()  
shape = img_np_hwc.shape  
img_np_nhwc=img_np_hwc.reshape((1,shape[0],shape[1],shape[2]))
```



5.如何优化速度

- 跳帧检测，两帧数据检测一帧
- 输入码流变小
- 模型优化



6.资源的释放和线程退出

```
while (main_thread_flag):
    try:
        . . .
        if (frame_data.eof):
            sub_thread_flag = False
            main_thread_flag = False
            raise OSError("get frame data failed")
        . . . .
    except KeyboardInterrupt as e:
        print("user stop: ", e)
    except BaseException as e:
        print(f"Exception {e}")
        sub_thread_flag = False
        main_thread_flag = False
        time.sleep_ms(100)
        face_det.deinit()
        kd_mp4_destroy(mp4_handle.value)
        vdec.stop()
        vdec.destroy()
        Display.deinit()
        csc.destroy()
        # 释放vb buffer
        time.sleep_ms(200)
        MediaManager.deinit()
        print("release end")
```

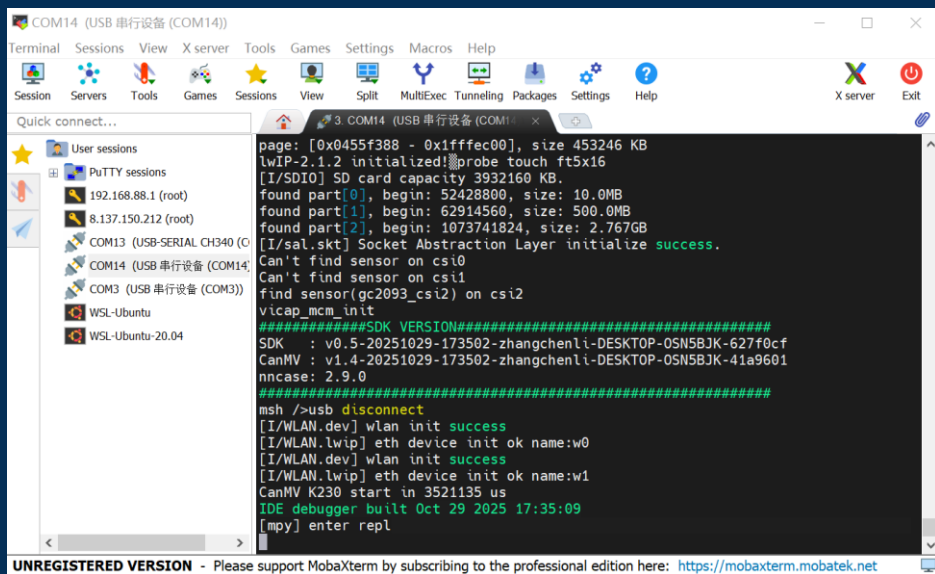


四、debug 方法

1、IDE log输出

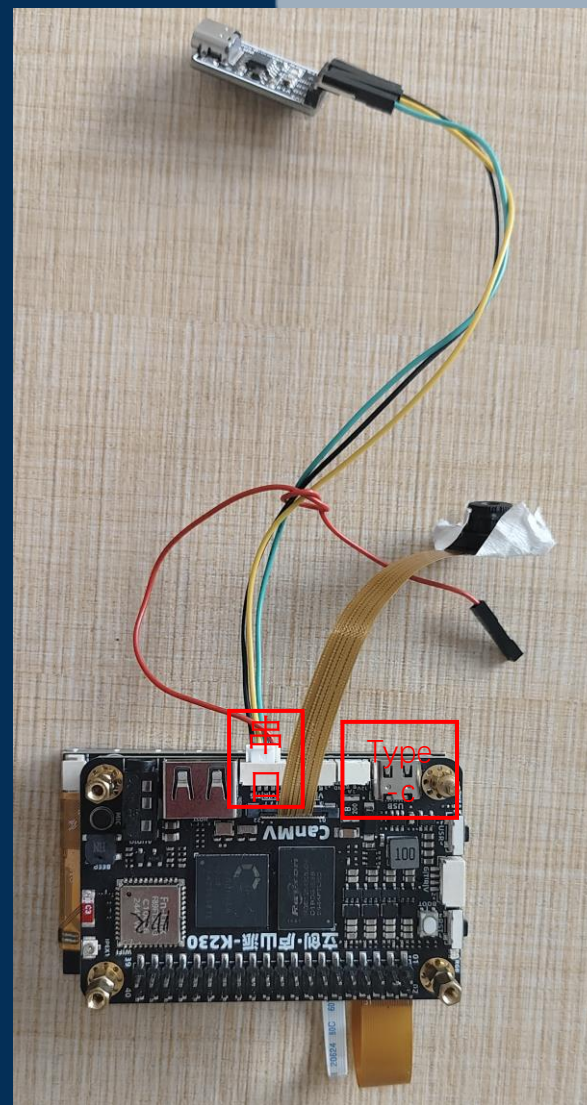
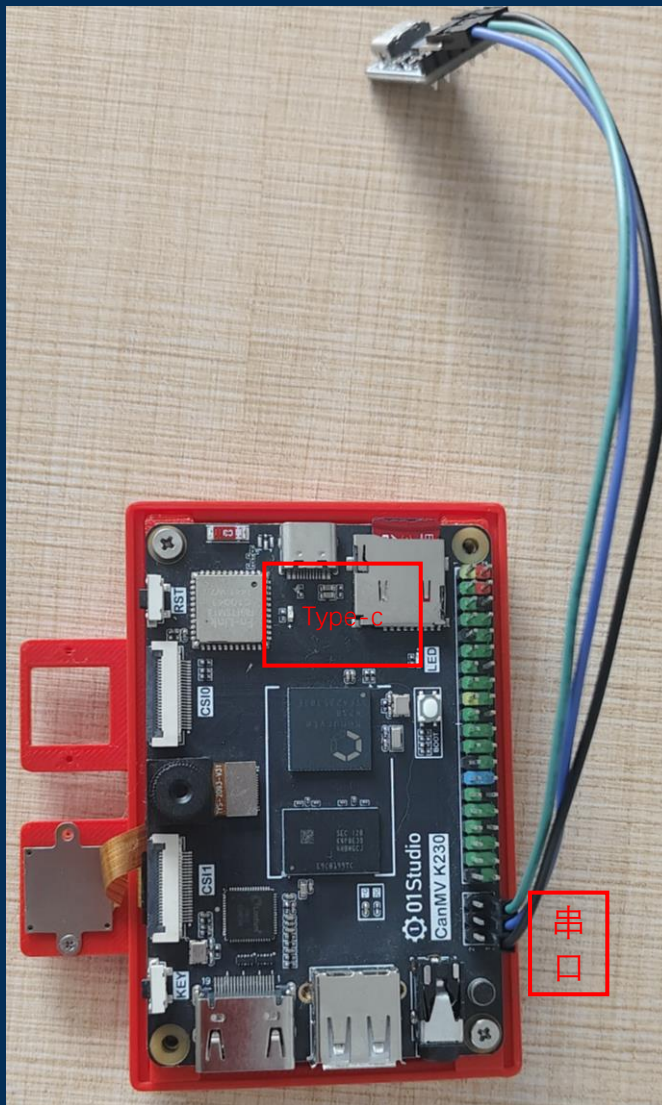
```
串行终端 |    
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
CSC.getframe get_frame ret 16  
Exception IDE interrupt  
destory_pool input 4  
destory_pool output 5
```

2、连接串口，查看串口输出log



```
COM14 (USB 串行设备 (COM14))
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
User sessions
PuTTY sessions
192.168.88.1 (root)
8.137.150.212 (root)
COM13 (USB-SERIAL CH340 (COM13))
COM14 (USB 串行设备 (COM14))
COM3 (USB 串行设备 (COM3))
WSL-Ubuntu
WSL-Ubuntu-20.04

page: [0x0455f388 - 0x1ffec00], size 453246 KB
lwIP-2.1.2 initialized! probe touch ft5x16
[I/SDIO] SD card capacity 3932160 KB.
found part[0], begin: 52428800, size: 10.0MB
found part[1], begin: 62914560, size: 500.0MB
found part[2], begin: 1073741824, size: 2.767GB
[I/sal.skt] Socket Abstraction Layer initialize success.
Can't find sensor on csi0
Can't find sensor on csi1
find sensor(gc2093_csi2) on csi2
vicap_mcm_init
#####SDK VERSION#####
SDK : v0.5-20251029-173502-zhangchenli-DESKTOP-OSNSBJK-627f0cf
CanMV : v1.4-20251029-173502-zhangchenli-DESKTOP-OSNSBJK-41a9601
nncase: 2.9.0
#####
msh />usb disconnect
[I/WLAN.dev] wlan init success
[I/WLAN.lwip] eth device init ok name:w0
[I/WLAN.dev] wlan init success
[I/WLAN.lwip] eth device init ok name:w1
CanMV K230 start in 3521135 us
IDE debugger built Oct 29 2025 17:35:09
[mpy] enter repl
```





3、后台运行micropython，使用命令行debug

代码块

```
1 #按ctrl+C退出micropython
2 [mpy] enter repl
3
4 ^C[mpy] exit, reset
5 <3>[4] [Func]:vb_do_exit [Line]:1569 [Info]:vb already exited!
6
7 #后台运行micropython
8 msh />cd /sdcard/
9 msh /sdcard>micropython &
10 msh /sdcard>CanMV K230 start in 349970133 us
11 IDE debugger built Oct 27 2025 09:55:21
12 [mpy] enter repl
13
14 #可以在运行的时候输入各种命令查看后台信息
15 msh /sdcard>cat /proc/umap/vb
16 -----VB PUB CONFIG-----
17 MaxPoolCnt
18      0
19 -----VB SUPPLEMENT ATTR-----
20 Config  Size  VbCnt
21 0        0      0
```



The End

提升社会运行效率 改善人类生活方式

Transforming Tomorrow - Smarter Efficiency Better Lives

