

Debug and run the routine

Debug and run the routine

Preface

Steps to run the routine

1. Open CanMV IDE
2. Connect K230 to the computer via USB cable
3. Run the example code

Offline operation

Preface

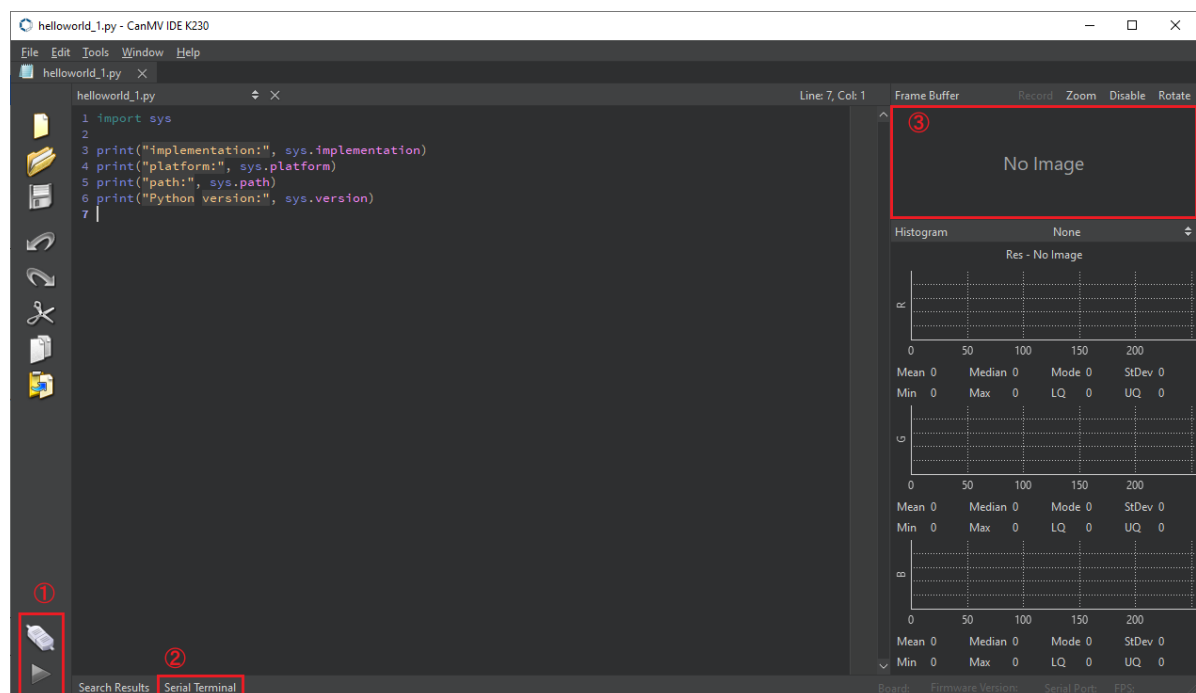
In this tutorial, we will learn how to use CanMV IDE to debug and run the routine code

Before starting this tutorial, please make sure that you have completed the installation of CanMV IDE and the burning of firmware mentioned in the previous chapter

Steps to run the routine

1. Open CanMV IDE

Double-click to run the installed CanMV IDE

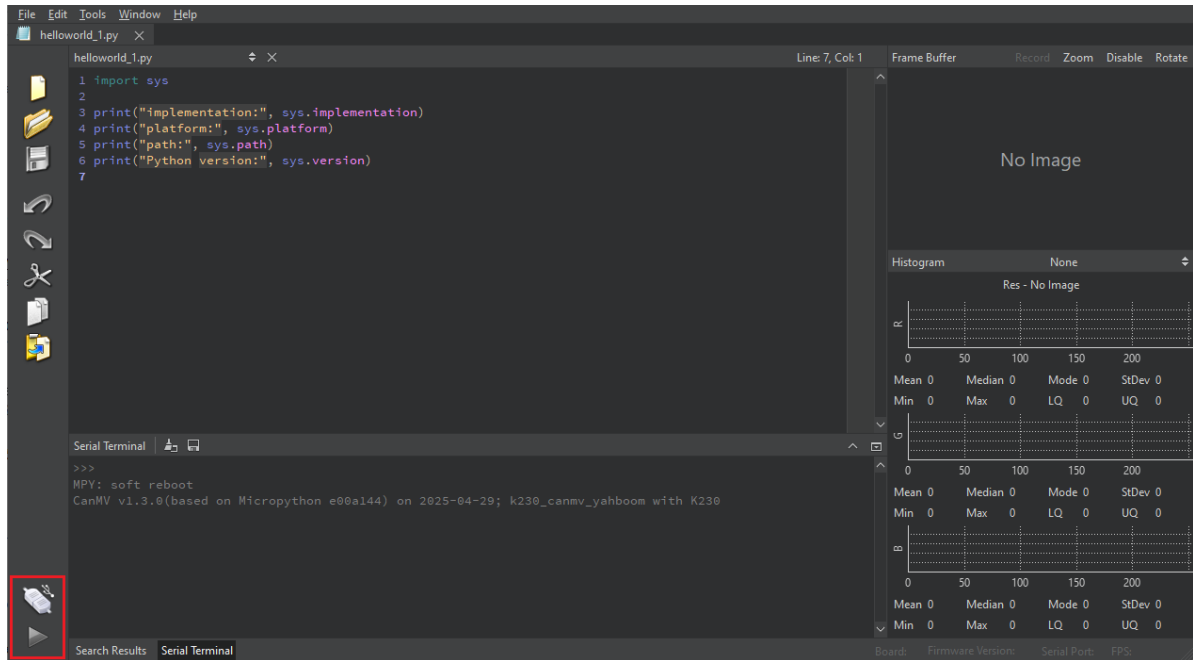


Click area ② to open the serial terminal

The purpose of the serial terminal is to facilitate us to view the debugging output during the program running process. Opening or closing the serial terminal will not affect the running effect of the program

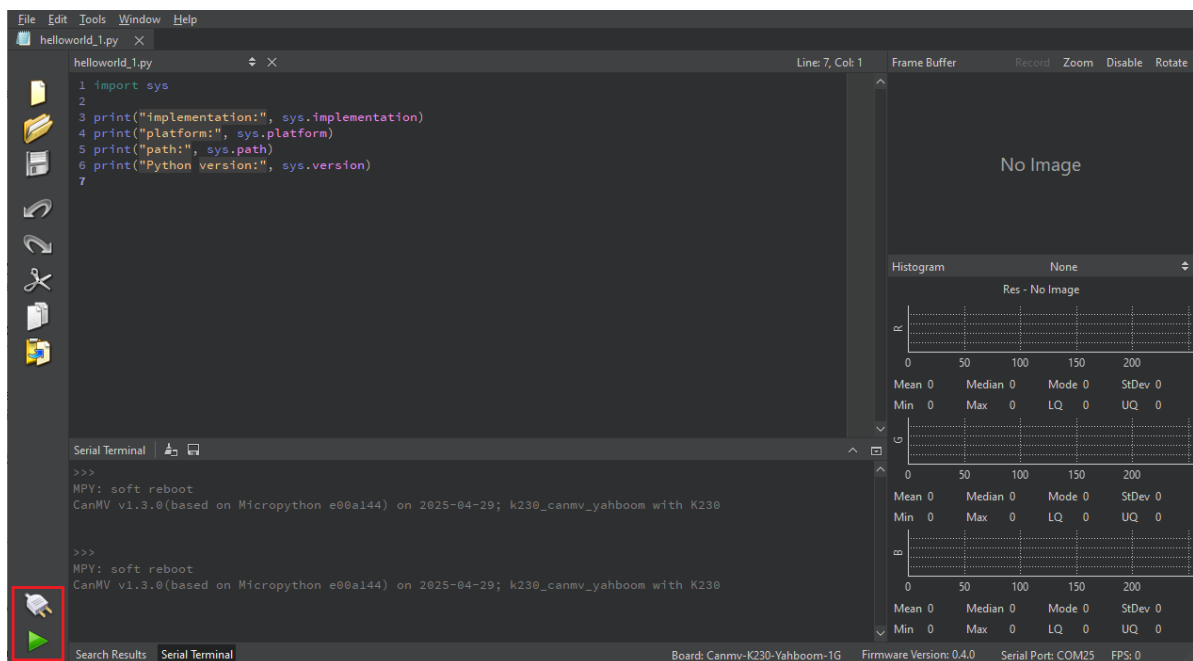
2. Connect K230 to the computer via USB cable

After the connection is successful, the icon in the lower left corner of CanMV IDE will change



If you do not turn on **【Auto Reconnect to CanMV】**, you need to manually click the button above and wait for CanMV IDE to establish a connection with K230

If automatic connection is turned on, wait for a few seconds, CanMV IDE can successfully connect to K230, and the lower left corner will become as shown in the figure



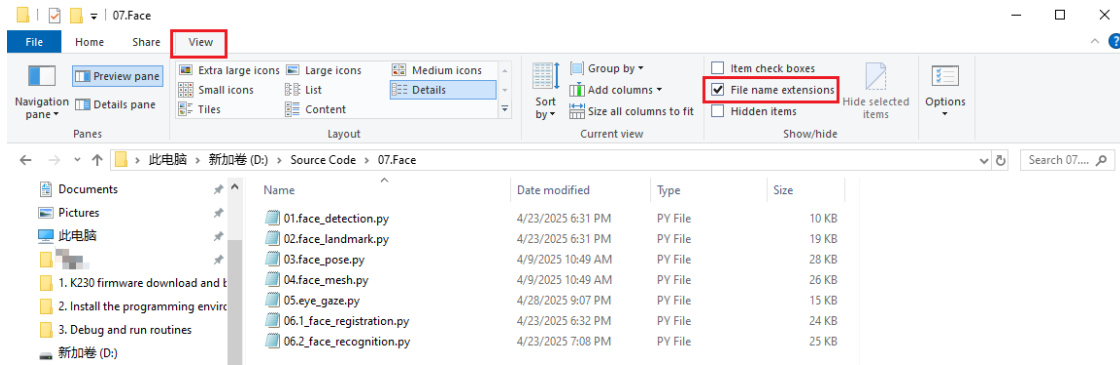
3. Run the example code

Let's take face detection as an example. Open the **【Source Code\07.Face】** directory in the downloaded data and find 01.face_detection.py

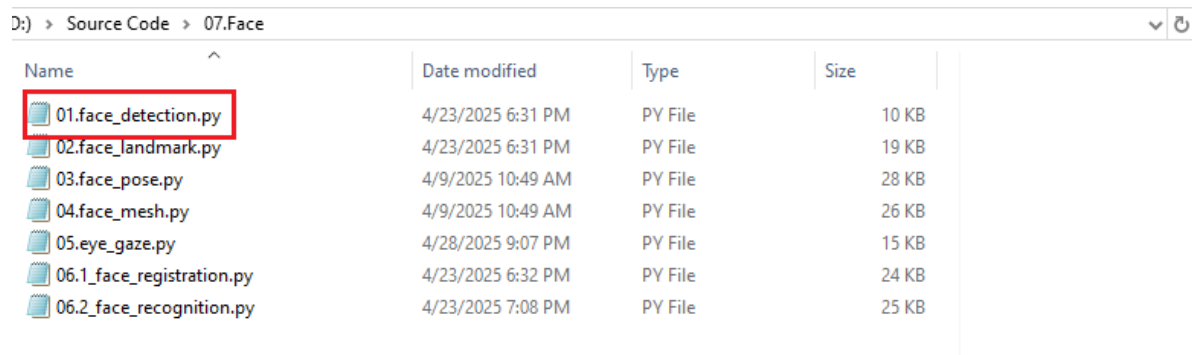
We double-click to open the file

1. The file name you may see is 01.face_detection, (no .py) because your computer system is not set to display file extensions

- It is strongly recommended to turn on this option during development. The methods for turning it on are different for different operating systems. Here we take win10 as an example



- If you have not set the default open file with the py suffix, you can choose to use the system's built-in Notepad software to open the code



```

01.face_detection.py - Notepad
File Edit Format View Help
from libs.PipeLine import PipeLine, ScopedTiming
from libs.AIBase import AIBase
from libs.AI2D import Ai2d
import os
import ujson
from media.media import *
from time import *
import nncase_runtime as nn
import ulab.numpy as np
import time
import utime
import image
import random
import gc
import sys
import aidemo
import _thread

face_det = None

from libs.YbProtocol import YbProtocol
from ybUtils.YbUart import YbUart
# uart = None
uart = YbUart(baudrate=115200)
pto = YbProtocol()

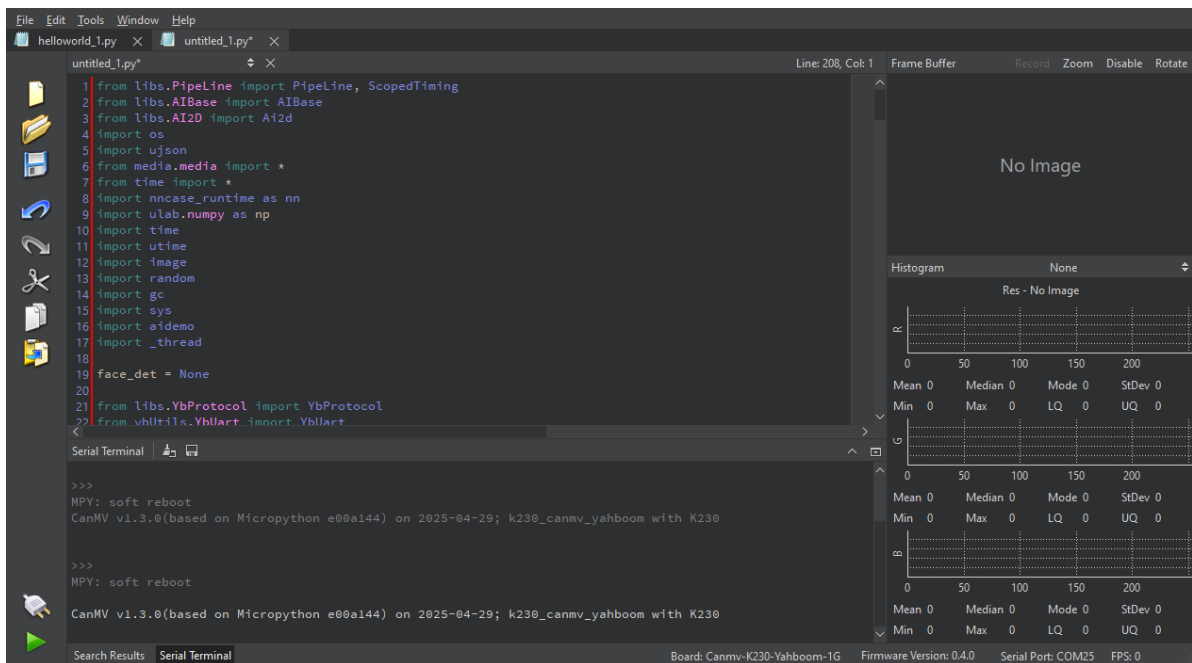
# 自定义人脸检测类，继承自AIBase基类 / Custom face detection class, inherited from AIBase base class
class FaceDetectionApp(AIBase):

```

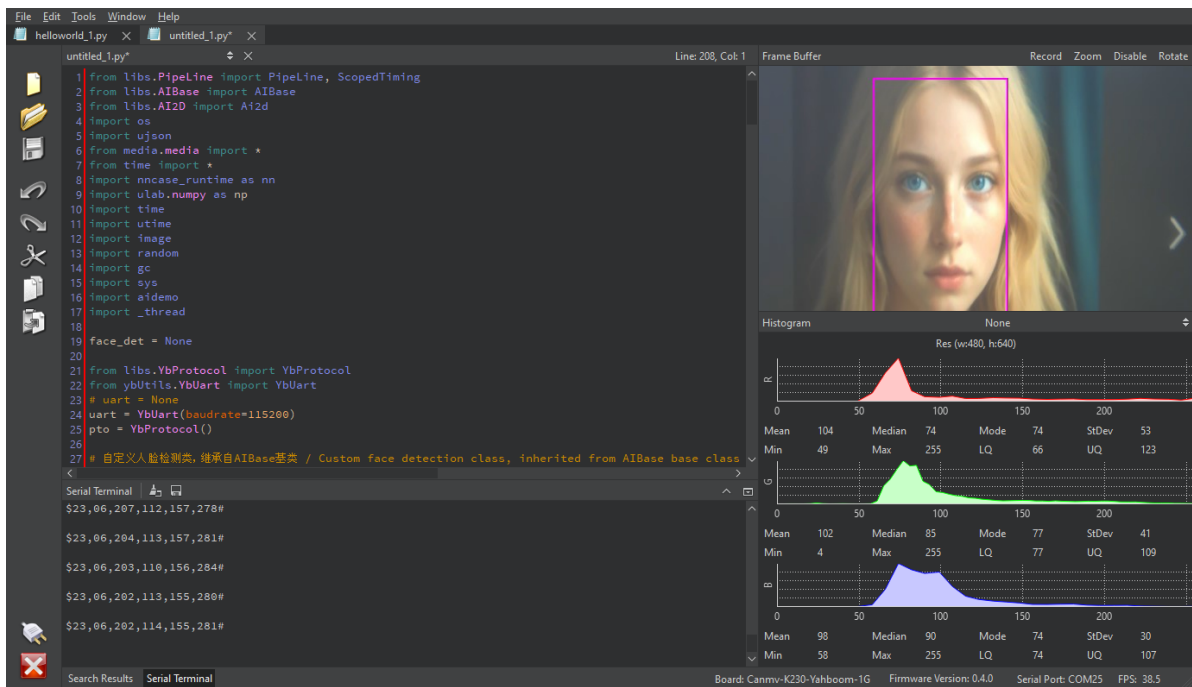
We select all the code, press Ctrl + C on the keyboard, and copy all the code

Then return to CanMV IDE, **clear the default content**, press Ctrl + V, and paste the code we just copied

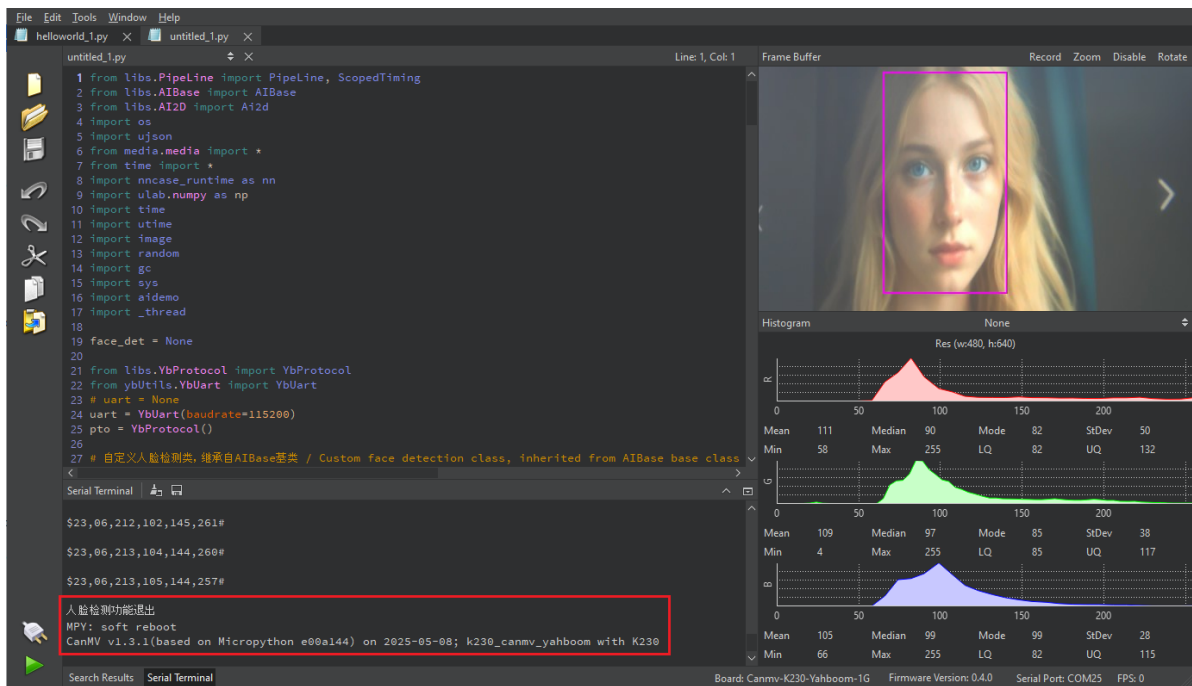
CanMV A new file in the IDE will have some default code content. Please select and delete all the default code before pasting.



After pasting successfully, click the **green run button** in the lower left corner, and K230 will execute the face detection function.



Click the red button in the lower left corner to exit the face detection function.



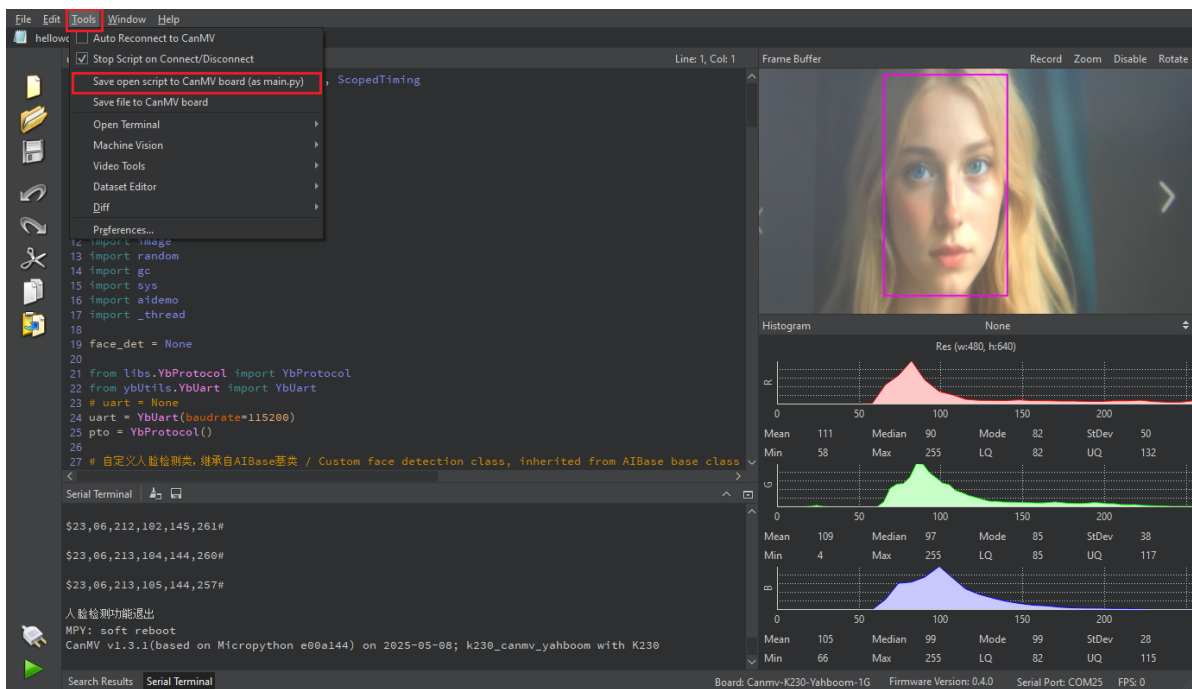
So far, we have successfully used K230 **online** to run a program.

Offline operation

Through the method mentioned above, we successfully let K230 run a face detection code in an online state (connected to CanMV IDE). At this time, the code is not saved on K230.

In actual scenarios, we often need to save the code offline in K230, and hope that K230 can automatically run the corresponding code after power on.

At this time, we only need to click **【Save open script to CanMV board (as main.py)】** in the toolbar in CanMV IDE



Notes

1. Please click this button when the code stops running, otherwise the save may fail
2. If you want to delete the automatically running program, just delete the main.py file in the /sdcard/ directory of K230

