1. Software Configuration

If you are using the latest Raspberry Pi Camera Module 3 or Raspberry Pi Global Shutter Camera, you need to run the following command to update the system (network connection is required).

```
sudo apt-get update -y

sudo apt-get upgrade -y
```

If only one camera is invoked, connect the camera to the CAM1 port.

If you do not use an official Raspberry Pi camera, you need to configure the "config.txt " file. If you use the latest Bookworm system, you need to configure /boot/firmware/config.txt.

```
sudo nano /boot/config.txt

#If using the bookworm system

sudo nano /boot/firmware/config.txt
```

Find "camera-auto-detect=1" and modify it to "camera_auto_detect=0".

At the end of the file, add the following setting statements according to the camera model.

Model	**Set Statement**
OV9281	dtoverlay=ov9281
IMX378	dtoverlay=imx378
IMX219	dtoverlay=imx219
IMX477	dtoverlay=imx477
IMX708	dtoverlay=imx708

2. Binocular Camera Configuration

Currently, both the CM4 and Raspberry Pi 5 support the connection of two cameras.

If you want to simultaneously connect to two cameras, you can designate the cameras by adding 'cam0' and 'cam1' after the corresponding camera configuration statements. For example, the imx219 is connected to the cam0 interface and the imx378 camera is connected to the cam1 interface.

```
dtoverlay=imx219,cam0

dtoverlay=imx378,cam1
```

3. Test Camera Commands

- Enter the Raspberry Pi terminal and enable the camera to preview:

```
sudo libcamera-hello -t 0
```

If you want to close the preview window, you can directly press the keys "Alt-F4", or click "x" to close. Also, you can return to the terminal interface and press "Ctrl-c" to end the demo.

Note: if using "Camera Module 3", the auto-focus function is enabled.

- Test Binocular Camera

When testing the binocular camera, you need to add "--camera" to specify the camera. If you do not add this parameter, "cam0" is specified by default.

```
sudo libcamera-hello -t 0 --camera 0

sudo libcamera-hello -t 0 --camera 1
```

4. Call Camera

- libcamera-hello

The libcamera software stack provides six commands for users to preview and test the camera interface.

```
libcamera-hello
```

This command will preview the camera on the screen for about 5 seconds. The user can use the "-t <duration>" parameter to set the preview time, where the unit of <duration> is milliseconds. If it is set to 0, it will keep previewing all the time. For example:

```
libcamerahello -t 0
```

- libcamera-jpeg

libcamera-jpeg is a simple static picture shooting program, different from the complex functions of libcamera-still, libcamera-jpeg code is more concise and has many of the same functions to complete picture shooting.

This shooting command will display a preview serial port for about 5 seconds, and then shoot a full-pixel JPEG image and save it as test.jpg.

```
libcamera-jpeg -o test.jpg
```

Users can set the preview time through the -t parameter and can set the resolution of the captured image through --width and --height. E.g:

```
libcamera-jpeg -o test.jpg -t 2000 --width 640 --height 480
```

Exposure control (global camera needs to use this command for exposure control)

All libcamera commands allow the user to set the shutter time and gain themselves, such as:

```
libcamera-jpeg -o test.jpg -t 2000 --shutter 20000 --gain 1.5
```

This command will capture an image with 20ms exposure and camera gain set to 1.5x.

The gain parameter set will first set the analog gain parameter inside the

photosensitive chip. If the set gain exceeds the maximum built-in analog gain value of the driver, the maximum analog gain of the chip will be set first, and then the remaining gain multiples will be used as digital gains to take effect.

Remarks: The digital gain is realized by ISP (image signal processing), not directly adjusting the built-in register of the chip. Under normal circumstances, the default digital gain is close to 1.0, unless there are the following three situations.

1) Overall gain parameter requirements, that is, when the analog gain cannot meet the set gain parameter requirements, the digital gain will be needed for compensation.

2) One of the color gains is less than 1 (the color gain is achieved by digital gain), in this case, the final gain is stabilized at 1/min(red_gain, blue_gain), that is, a uniform digital gain is applied, and is the gain value for one of the color channels (not the green channel).

3) AEC/AGC was modified. If there is a change in AEC/AGC, the digital gain will also change to a certain extent to eliminate any fluctuations, this change will be quickly restored to the "normal" value.

- The Raspberry Pi's AEC/AGX algorithm allows the program to specify exposure compensation, which is to adjust the brightness of the image by setting the aperture value, for example:

```
libcamera-jpeg --ev -0.5 -o darker.jpg
```

```
libcamera-jpeg --ev 0 -o normal.jpg

libcamera-jpeg --ev 0.5 -o brighter.jpg
```

- libcamera-vid

  libcamera-vid is a video recording demo,After the program runs, a preview window will be displayed on the screen, and simultaneously the bitstream encoding will be output to the specified file. For example, record a 10s video.

```
libcamera-vid -t 10000 -o test.mp4
```

For more detailed instructions on how to use the camera, refer to the official Raspberry Pi tutorial:

https://www.raspberrypi.com/documentation/computers/camera_software.html