2. Camera internal parameter calibration

1. Preparation before calibration

- A large chessboard of known dimensions. This tutorial uses a 9x6 checkerboard and a 20mm square, which needs to be flattened during calibration. **The calibration uses the internal vertices of the checkerboard, so a "10x7" checkerboard uses the internal vertex parameters "9x6", as shown in the example below.** Calibration boards of any specifications are acceptable, just change the parameters. An open area without obstacles or calibration board patterns
- Monocular camera for publishing images via ROS

2. Start calibration

Install the calibration function package camera_calibration, take foxy as an example, enter in the terminal,

```
sudo apt install ros-foxy-camera-calibration*
```

Start the camera before calibration, and then turn off the camera until all calibrations are completed. Start the camera and enter in the terminal,

```
ros2 launch orbbec_camera gemini2.launch.py
```

Use the following command to view the topic input in the terminal,

```
ros2 topic list
```

The topic we need to use to calibrate RGB color images is /camera/color/image_raw.

Run the calibration program and enter in the terminal,

```
ros2 run camera_calibration cameracalibrator --size 9x6 --square 0.02 --ros-args
--remap /image:=/camera/color/image_raw
```

size: Calibrate the number of internal corner points of the checkerboard, for example, 9X6, with a total of six rows and nine columns of corner points.

square: The side length of the checkerboard, in meters.

Topic name: /camera/color/image_raw, if **usb_cam** is started, change it here to **/image_raw**

X: The left and right movement of the checkerboard in the camera field of view

Y: The checkerboard moves up and down in the camera field of view

Size: the movement of the checkerboard back and forth in the camera field of view

Skew: The tilt and rotation of the checkerboard in the camera's field of view

As shown in the picture above, you need to collect the image by flipping it up and down, front and back, left and right, so that the X, Y, Size and Skew on the right turn green, as shown in the picture below, then click CALIBRATE to start calibration.

After the calibration is completed, click SAVE, as shown in the figure below,

The calibration results are saved to [/tmp/calibrationdata.tar.gz], and the saving path is the terminal directory where the calibration program is started. After the calibration is completed, you can move out the [/tmp/calibrationdata.tar.gz] file to see the content.

```
sudo mv /tmp/calibrationdata.tar.gz ~
```

Terminal input,

```
cd ~
tar -xvf calibrationdata.tar.gz
```

You will get the calibrated png file, ost.yaml and ost.txt files in the terminal directory.

Since the gemini2 driver loads the calibrated built-in parameters in the code when it is started, there is no need to load the calibrated parameters. However, when starting the USB camera, the parameters need to be loaded. Therefore, after calibration, the parameters need to be replaced with the original built-in parameters. Parameters, rename the calibrated ost.yaml to camera_info.yaml, then replace the original camera_info.yaml, and enter it in the docker terminal.

```
#Copy the file to /opt/ros/foxy/share/usb_cam/config first
sudo cp ost.yaml /opt/ros/foxy/share/usb_cam/config
#Switch to the /opt/ros/foxy/share/usb_cam/config directory
cd /opt/ros/foxy/share/usb_cam/config
#Back up the original camera_info.yaml
sudo mv camera_info.yaml camera_info_BK.yaml
#Rename ost.yaml to camera_info.yaml
sudo mv ost.yaml camera_info.yaml
```