


Lesson 2 Camera Calibration



The camera calibration methods are generally divided into two types. Method 1 is to calibrate by obtaining the captured checkerboard from the camera in real time, and Method 2 is to calibrate with some checkerboard images obtained before. This lesson will introduce the first one.

Note: Please print out the checkerboard image saved under the same directory in advance. The checkerboard for calibration in this lesson is generated by the built-in file of system. Other checkerboard images are not applicable. If the image is deleted accidentally, it can be regenerated according to the operation steps of "1. Generate Checkerboard Image".

1. Generate Checkerboard Image

 It should be case sensitive when entering command, and the "Tab" key can be used to complete the keywords. If the checkerboard image has been printed out, you can skip steps 5-7.

1) Turn on ArmPi Pro and connect it to the system desktop via No Machine.

2) Click  and select  to open the terminal.

3) Enter command "sudo systemctl stop start_node.service" and press "Enter" to stop all the ros nodes.

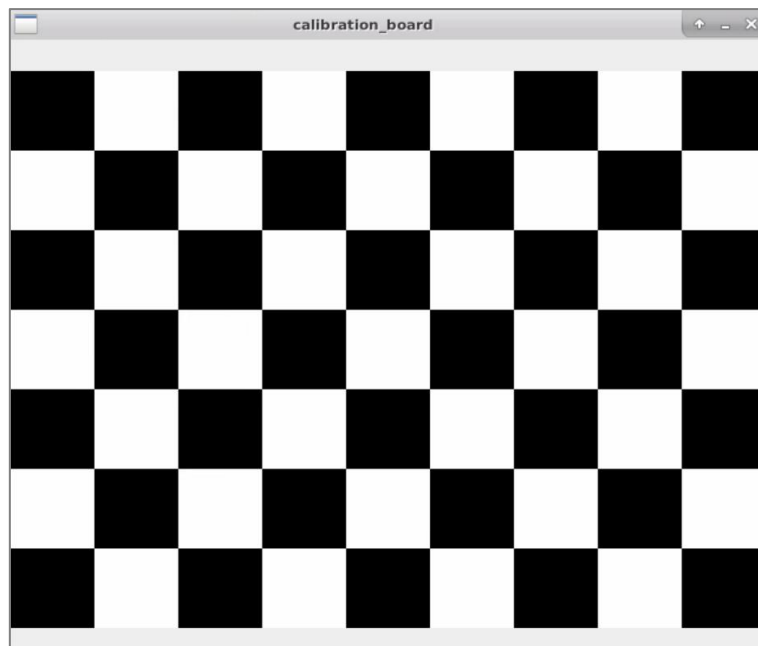
```
ubuntu@ubuntu:~$ sudo systemctl stop start_node.service
```

4) Enter `cd /home/ubuntu/armpi_pro/src/armpi_pro_demo/CameraCalibration/` command and press "Enter" to access to the program directory of monocular camera.

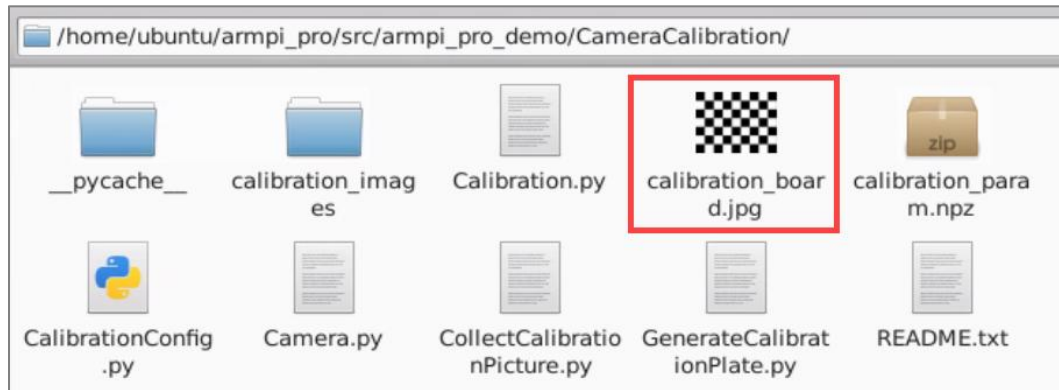
```
ubuntu@ubuntu:~$ cd /home/ubuntu/armpi_pro/src/armpi_pro_demo/CameraCalibration/
```

- 5) Enter command “sudo python3 GenerateCalibrationPlate.py” and press “Enter” to run the program for generating checkerboard. At this time, the generated checkerboard image will be displayed on screen.

```
ubuntu@ubuntu:~/armpi_pro/src/armpi_pro_demo/CameraCalibration$  
sudo python3 GenerateCalibrationPlate.py  
Press a random button to exit
```



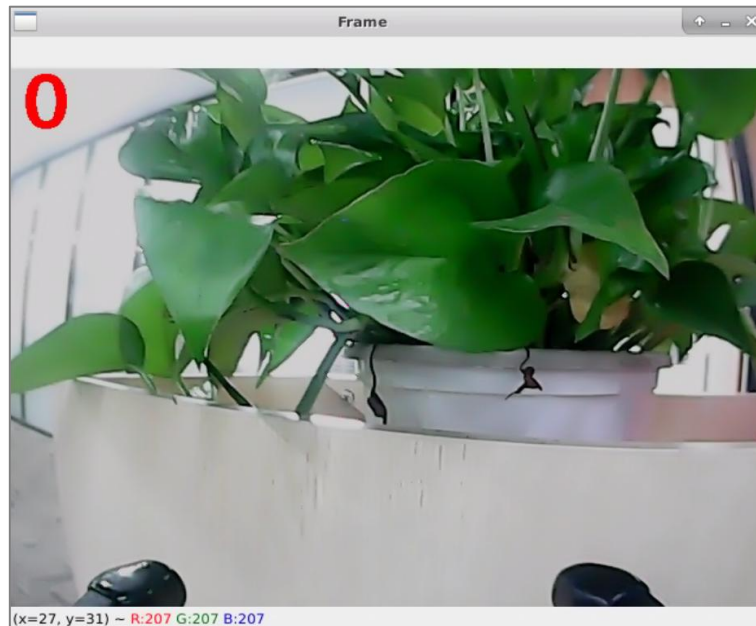
- 6) After generating the checker board, press a key arbitrarily to exit the program.
- 7) Next, find the generated checkerboard image in “/home/ubuntu/armpi_pro/src/armpi_pro_demo/CameraCalibration/calibration_board.jpg” and drag it to the local file. Then print the image.



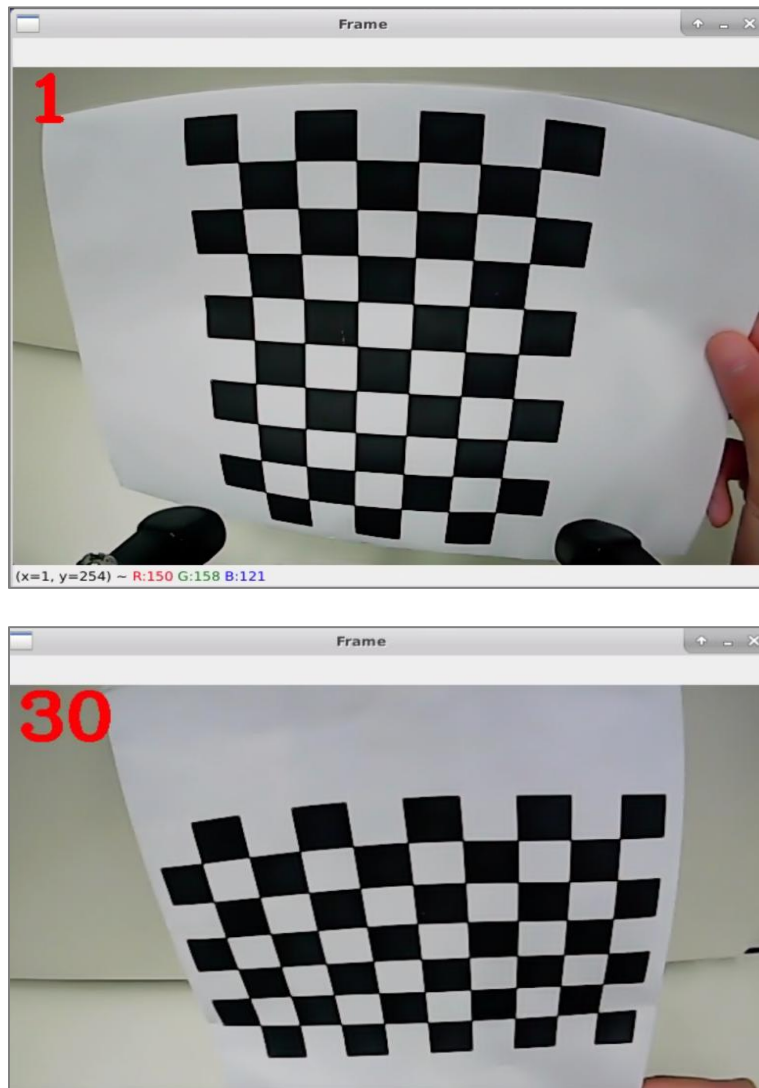
2. Collect the Information of Calibration Image

1) Enter command “sudo python3 CollectCalibrationPicture.py” and press “Enter” to run the program of calibration image information.

```
ubuntu@ubuntu:~/arduino_pro/src/arduino_pro_demo/CameraCalibration$  
sudo python3 CollectCalibrationPicture.py  
Press "space" on keyboard to save image and press "Esc" to exit.
```



2) Put the printed checkerboard image within the identified range of camera and press the space bar to save the collected images (collect 20-30 pieces of pictures in different positions).



3) After collecting, press “Esc” in the interface shown above to exit the program.

3. Calibration Image

1) Enter command “sudo python3 Calibration.py” and press “Enter” to run the program of calibration image. Then, camera starts calibrating the camera automatically according the collected image information.

```
ubuntu@ubuntu:~/armpi_pro/src/armpi_pro_demo/CameraCalibration$  
sudo python3 Calibration.py
```

2) After calibrating, appear the prompt “save successful” and exit the

calibration program automatically.

```
Found 5 valid images for calibration
resolution      : DIM=(640, 480)
intrinsic parameters : K=np.array([[537.9753510246653, 0.0, 3
05.2501555999097], [0.0, 538.1507964815162, 200.1928957921044],
[0.0, 0.0, 1.0]])
Distortion coefficients: D=np.array([[0.07267815066747105], [-5.
561320779681371], [41.57575009478954], [-97.7521585427949]])
param save successful
```

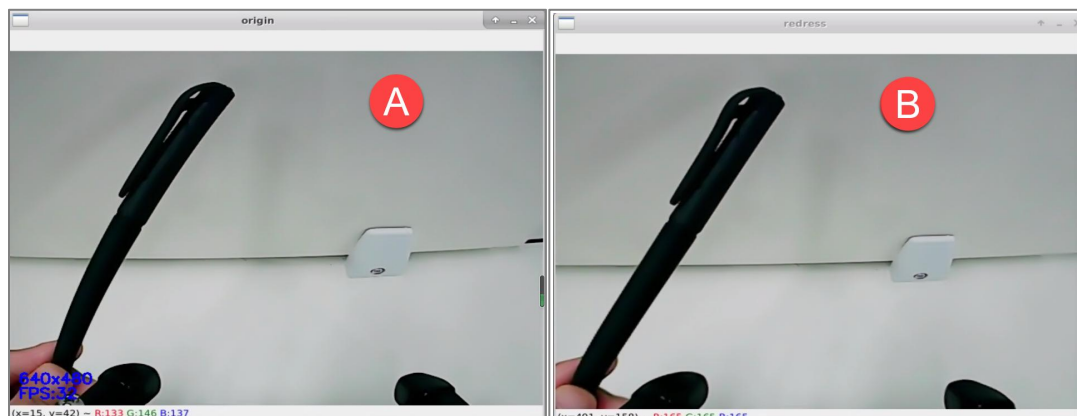
4. Calibration Test

1) Then enter command “**sudo python3 TestCalibration.py**” and press “Enter” to run the program of calibration test.

```
ubuntu@ubuntu:~/armpi_pro/src/armpi_pro_demo/CameraCalibration$
sudo python3 TestCalibration.py
```

```
dim:
(640, 480)
k:
[[658.91966905  0.  334.91108516]
 [  0.  663.38419636 263.07372599]
 [  0.  0.  1.  ]]
d:
[[-0.41171016]
 [ 1.06297639]
 [-8.0326887 ]
 [17.72447014]]
```

2) At this time, the screen will display two image windows transmitted by camera. Image A is the original one without calibration and it is obviously a distorted image. Image B is the calibrated one and it is normal image.



3) Press “Esc” to exit program. Now, camera calibration is completed.