CVGesture

Performance Report

2017-12-27

OPEN AI LAB

Revision Record

Date	Rev	Change Description	Author
2017-10-19	0.1.0	Initial version	
2017-12-27	0.2.0	New cascade classifier	Hao Han

catalog

1 Purpose

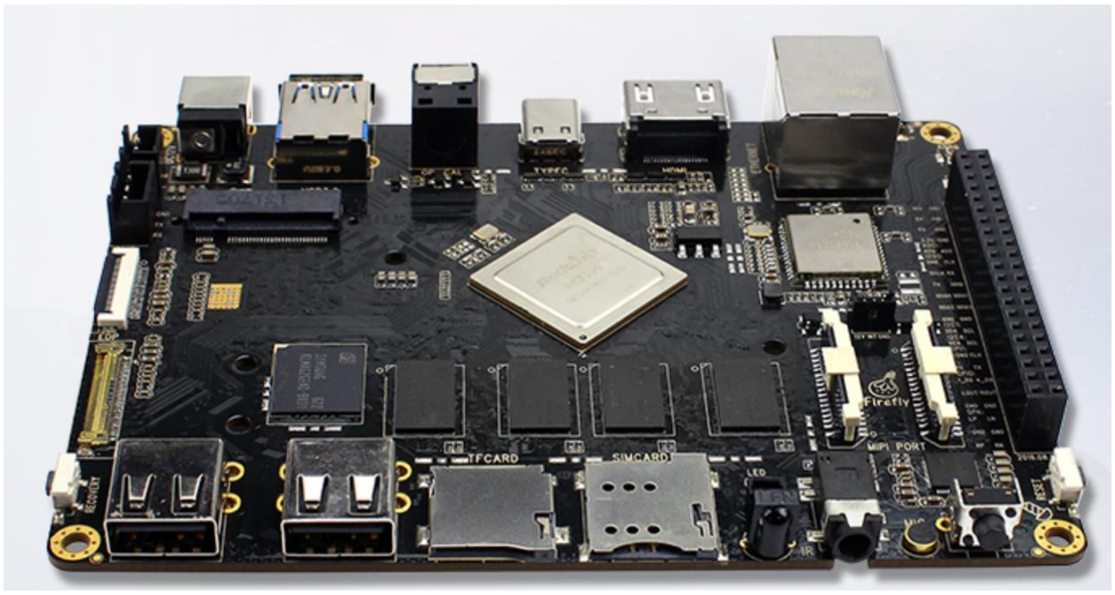
This Report is tested on RK3399 platform. The report includes CPU data.

2 Test Environment

Hardware SoC : Rockchip RK3399

- GPU: Mali T864 (800MHz)
- CPU: Dual-core Cortex-A72 up to 2.0GHz (real frequency is 1.8GHz); Quad-core Cortex-A53 up to 1.5GHz (real frequency is 1.4GHz)

Operating System : Ubuntu 16.04



Software : OpenCV 3.3.0

3 Definition of Gestures

In order to test the performance of the application, there should be some definitions and limitations to different gestures:

- Palm: with five fingers open (do not close five fingers together, as fig.3), facing directly to the camera (as fig.1). The angle of rotation of hand in the direction of front and back should not go beyond 30 degrees. The angle of rotation of hand in the direction of left and right should not go beyond 45 degrees. The standard is the same for both left and right hand.

- Fist: with fist clenched, facing directly to the camera (as fig.2). The angle of rotation of hand in the direction of front, back, left and right should not go beyond 30 degrees. The standard is the same for both left and right hand.

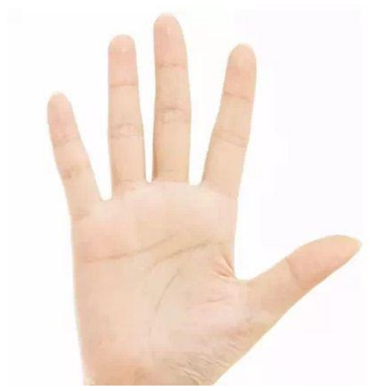


Figure 1. Palm facing directly to camera

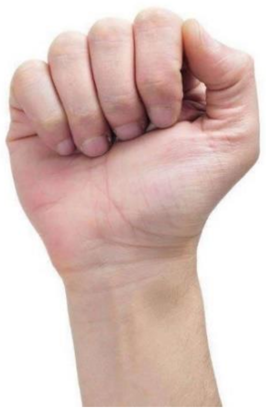


Figure 2. Fist facing directly to camera

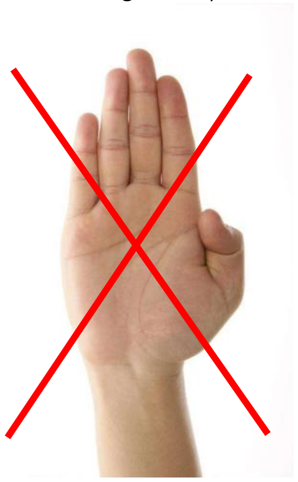


Figure 3. Wrong palm gesture

4 Performance on Different Cores

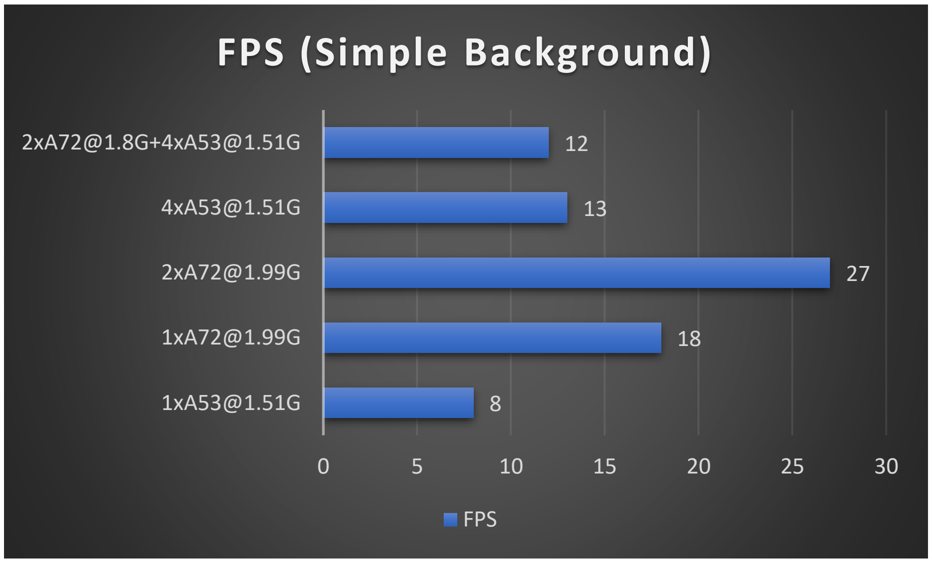
Calculate the FPS(Frame rate Per Second) in five seconds, and print the result in terminal. Skip the result of first five seconds. Recognize two gestures: palm and fist. The palm and fist detection time are also averaged in five seconds and 100 frames, skip the result of first five seconds. As the complexity of background will significantly affect the performance, the performance under different backgrounds will be given.

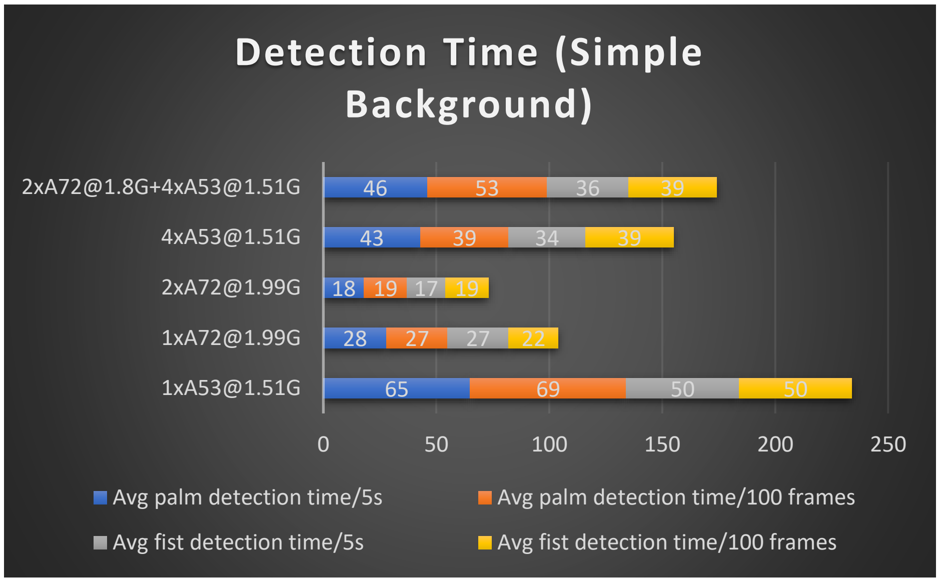
4.1 Simple background

Simple background means that the background does not contain object of irregular shape like a plain wall. Performance under simple background will be stronger.

FPS on different cores, 640x480 resolution

	FPS	Palm detection time (ms) in 5 seconds	Palm detection time (ms) in 100 frames	Fist detection time (ms) in 5 seconds	Fist detection time (ms) in 100 frames
1xA53@1.51G	8	65	69	50	50
1xA72@1.99G	18	28	27	27	22
2xA72@1.99G	27	18	19	17	19
4xA53@1.51G	13	43	39	34	39
2xA72@1.8G+4xA53@1.51G	12	46	53	36	39



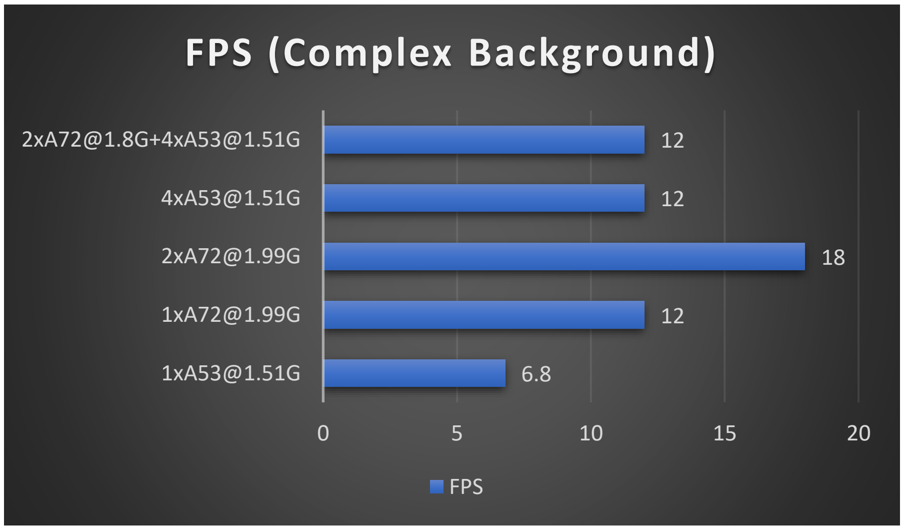


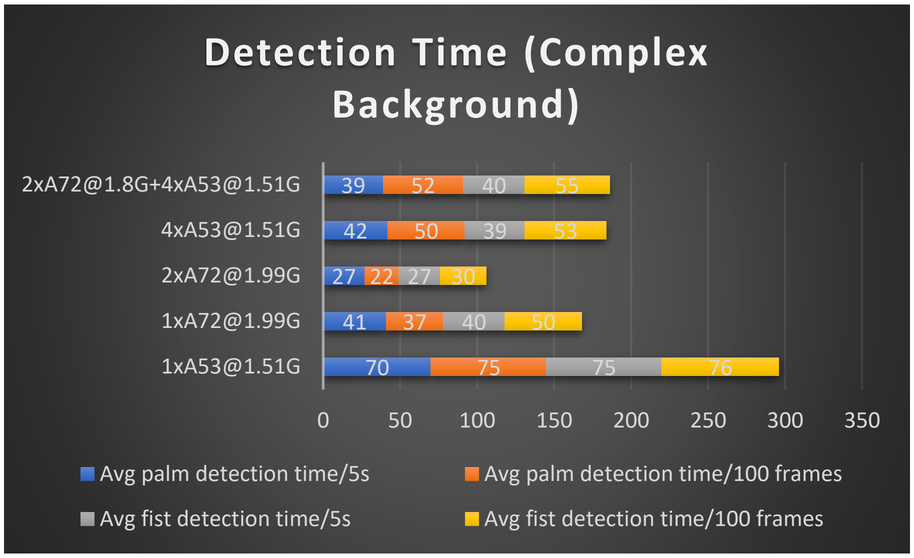
4.2 Complex background

Complex background means that the background contains many objects of irregular shape. Performance under complex background will be weaker.

FPS on different cores, 640x480 resolution

	FPS	Palm detection time (ms) in 5 seconds	Palm detection time (ms) in 100 frames	Fist detection time (ms) in 5 seconds	Fist detection time (ms) in 100 frames
1xA53@1.51G	6.8	70	75	75	76
1xA72@1.99G	12	41	37	40	50
2xA72@1.99G	18	27	22	27	30
4xA53@1.51G	12	42	50	39	53
2xA72@1.8G+4xA53@1.51G	12	39	52	40	55





5 Conclusion

From the above test cases, we can deduce that :

- the performance on 2xA72 is the best
- the performance on 4xA53+2xA72 is similar versus 4xA53

The algorithm should run on A72 or 4xA53, single A53 core cannot meet the performance requirement.

6 Testing Issues

There are many known factors that will significantly affect the performance of the application:

- The **version of OpenCV should be 3.3.0**, using OpenCV2 will slow down the detection
- The performance of the application depends on the complexity of background, running under **simple background like a plain wall** will be significantly faster