CaffeOnACL

Performance Report

2018-01-25

OPEN AI LAB

Revision Record

Date	Rev	Change Description	Author
2017-9-20	0.1.0	Initial	Huifang
2017-10-11	0.2.0	Validation Arm Compute Library v17.09	Huifang
2017-11-28	0.3.0	Validation Arm Compute Library v17.10	Huifang
2018-01-25	0.4.0	Validation Arm Compute Library v17.12	Huifang

Catalog

1 Purpose

This Report is tested on RK3399 platform and the Arm Compute Library is version 17.12. The report includes both CPU data and GPU data. We collected the data on AlexNet, GoogLeNet, SqueezeNet and MobileNet. And we found the mixed mode can improve performance 2.92X for the best case.

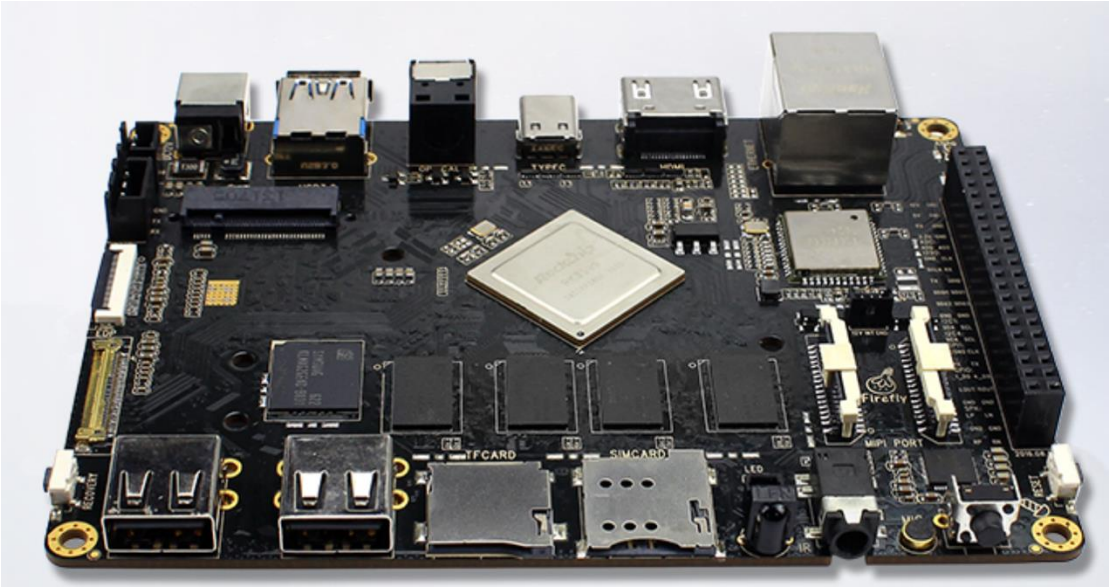
2 Test Environment

Hardware SoC: firefly

http://www.t-firefly.com/product/rk3399.html

- GPU: Mali T864 (800MHz)
- RAM: 4G
- 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



3 Performance Improvement Achievement

The ACL_NEON's LRN and POOLING are better, and ACL_CL(GPU) has the better performances on large FC while OpenBLAS has better on CONV. It's possible to gain better performance on mixing the calculation on different component, for example, using OpenBLAS layers (SoftMax, RELU, FC, CONV) and ACL_NEON layers (LRN, Pooling) in neural network.

After we mixed the layers calculation on OpenBLAS and ACL, it's very easy to mix the layers calculation by exporting environment variable BYPASSACL, details in User Guide 5.2.

For the total time spent per inference, we have achieved about 2.78X performance in the best case.

	Original Caffe(s)	Mixed Mode(s)	Performance Gain
AlexNet	0.959	0.5386	1.7X
GoogleNet	1.3864	0.4978	2.78X
SquezzeNet	0.1458	0.1517	0.096X
MobileNet	0.3123	0.2957	1.06X

4 Performance

For GPU, the OpenCL driver need compile CL kernel for the first time running, but after 2nd time, the CL kernel may not be compiled. This will impact performance. Here we list the 1st data separately. We tested total 10 times from 2nd to 11th and calculated the average time. The data in the below tables are in the unit of second.

The items (TPI, Allocate, Run, Config, Copy, FC, CONV, LRN, Pooling, RELU, SOFTMAX) in the below tables:

TPI: The total time for per inference
Avg. Time: tested total 10 times from 2nd to 11th and calculated the average time. The unit of all the data columns in tests below is second.

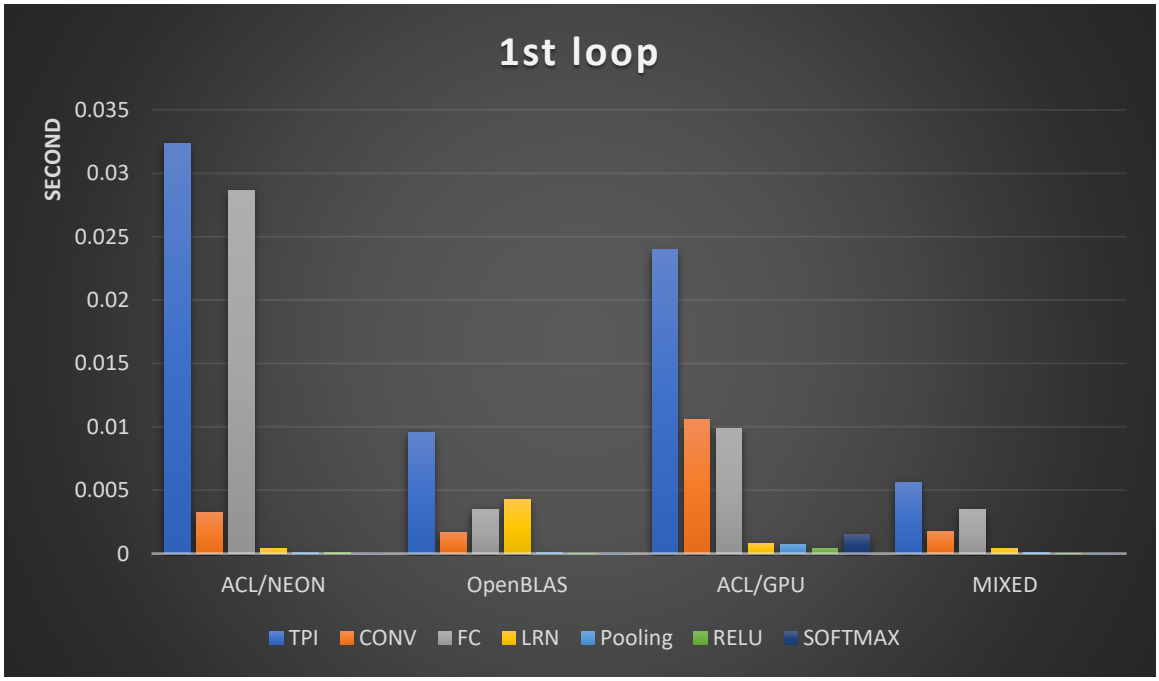
The details see user manual section “Use Cases”.

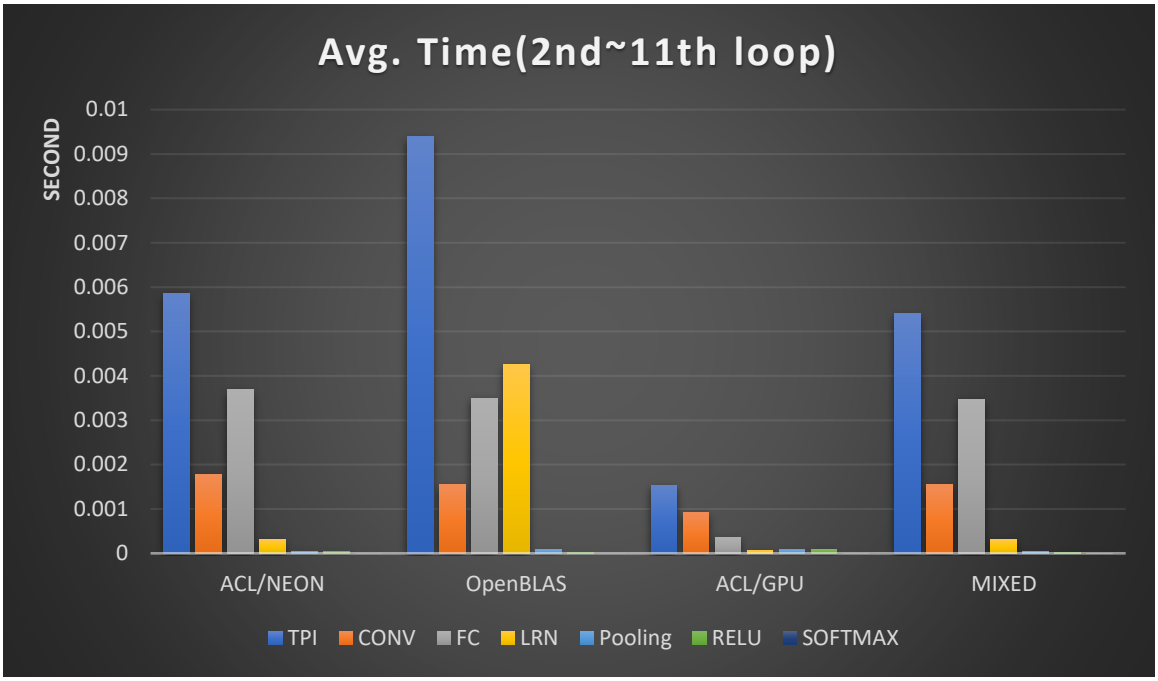
Note that the CPU data of this section is on a single A72 core.

4.1 AlexNet

	TPI(s)	Allocate(s)	Run(s)	Config(s)	Copy(s)
1st					
ACL/NEON	3.2382	0.1729	2.7081	0.2177	0.1365
OpenBLAS	0.959				
ACL/GPU	2.3964	0.1721	0.0646	1.3768	0.7785
MIXED	0.5641	0.0044	0.0326	0.0013	0.0055
Avg. Time					
ACL/NEON	0.586		0.5763		0.0091
OpenBLAS	0.9388				
ACL/GPU	0.1525		0.0134		0.138
MIXED	0.5398		0.0317		0.0043

	TPI(s)	CONV(s)	FC(s)	LRN(s)	Pooling(s)	RELU(s)	SOFTMAX(s)
1st							
ACL/NEON	3.2382	0.3223	2.8634	0.0374	0.0069	0.0079	0.0003
OpenBLAS	0.959	0.17	0.3491	0.429	0.0093	0.0014	0.0002
ACL/GPU	2.3964	1.0582	0.9893	0.079	0.074	0.043	0.1529
MIXED	0.5641	0.1705	0.3471	0.038	0.0069	0.0014	0.0002
Avg. Time							
ACL/NEON	0.586	0.177	0.3691	0.0315	0.0046	0.0037	0.0001
OpenBLAS	0.9388	0.1548	0.3493	0.4249	0.0082	0.0014	0.0001
ACL/GPU	0.1525	0.0913	0.0365	0.0069	0.0078	0.0095	0.0005
MIXED	0.5398	0.1549	0.3472	0.0318	0.0044	0.0014	0.0001

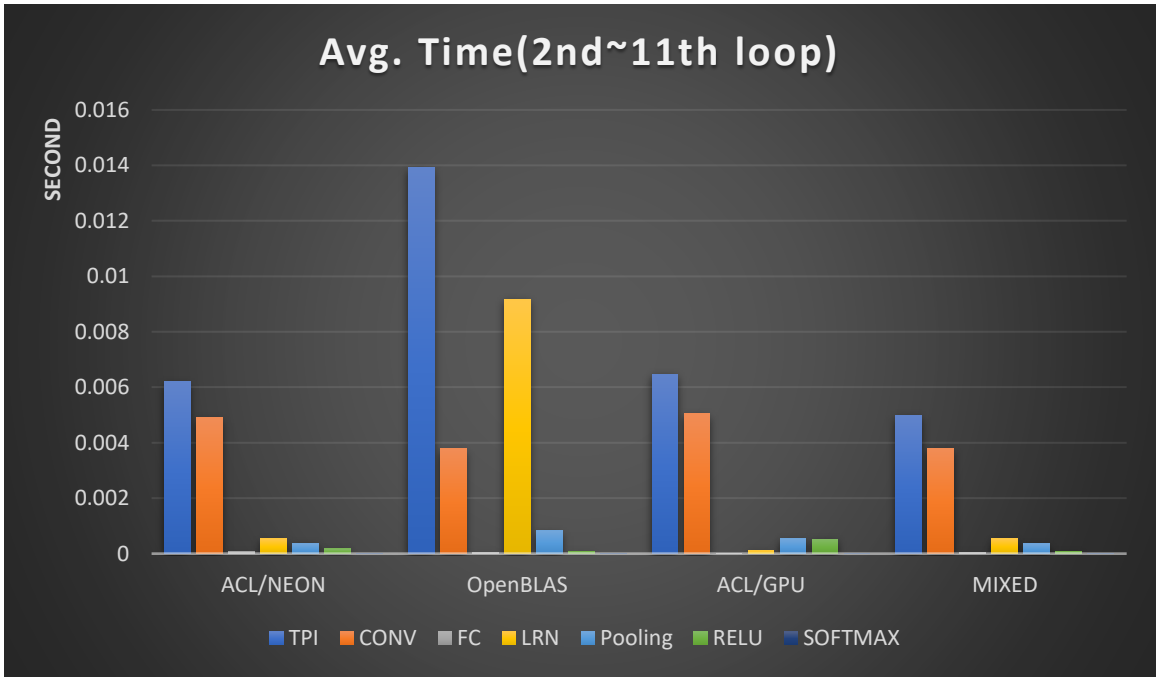




GoogleNet

	TPI(s)	Allocate(s)	Run(s)	Config(s)	Copy(s)
1st					
ACL/NEON	1.1663	0.0848	0.6274	0.2226	0.2146
OpenBLAS	1.443				
ACL/GPU	4.9772	0.1106	0.1237	3.5542	1.1656
MIXED	0.5703	0.0256	0.0837	0.0037	0.0307
Avg. Time					
ACL/NEON	0.6207		0.5588		0.0567
OpenBLAS	1.3918				
ACL/GPU	0.6475		0.0794		0.5593
MIXED	0.4994		0.0827		0.023

	TPI(s)	CONV(s)	FC(s)	LRN(s)	Pooling(s)	RELU(s)	SOFTMAX(s)
1st							
ACL/NEON	1.1663	0.9628	0.0194	0.0629	0.0541	0.0385	0.0002
OpenBLAS	1.443	0.406	0.0052	0.9322	0.0874	0.0069	0.0002
ACL/GPU	4.9772	4.1447	0.1484	0.0866	0.2326	0.1415	0.1529
MIXED	0.5703	0.4094	0.0052	0.064	0.0559	0.007	0.0002
Avg. Time							
ACL/NEON	0.6207	0.4892	0.0062	0.0536	0.0366	0.0177	0.0001
OpenBLAS	1.3918	0.3781	0.0052	0.9173	0.0819	0.0069	0.0001
ACL/GPU	0.6475	0.5047	0.0015	0.0116	0.0533	0.0522	0.0005
MIXED	0.4994	0.3805	0.0048	0.0542	0.0361	0.0068	0.0001

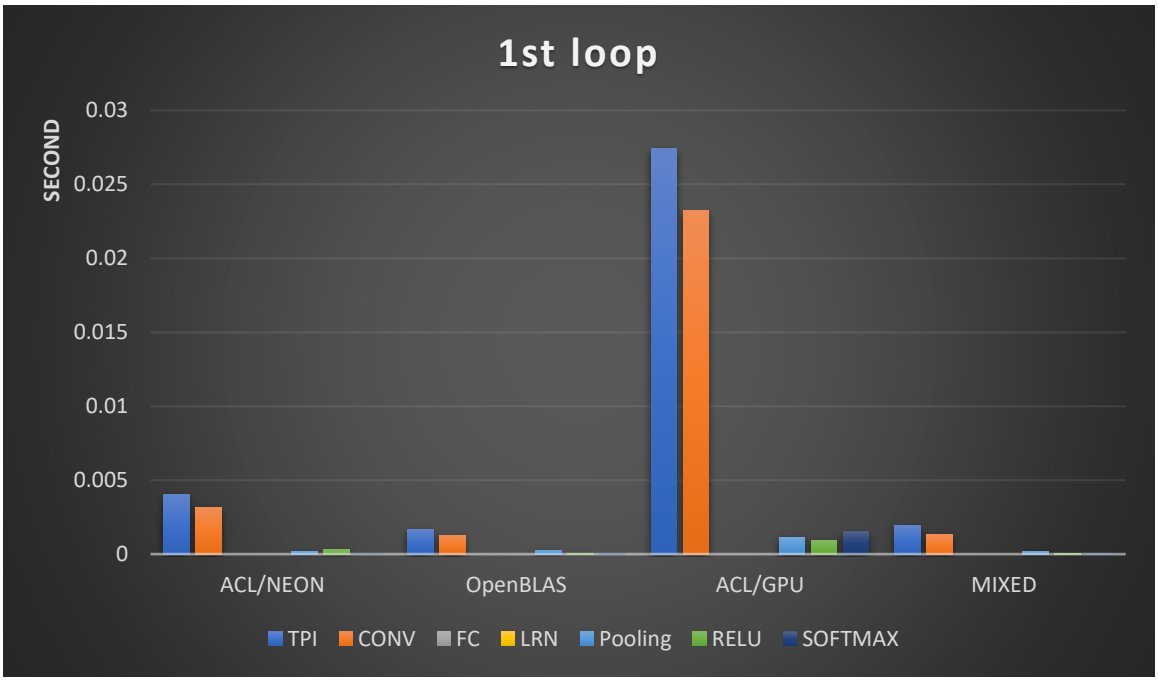


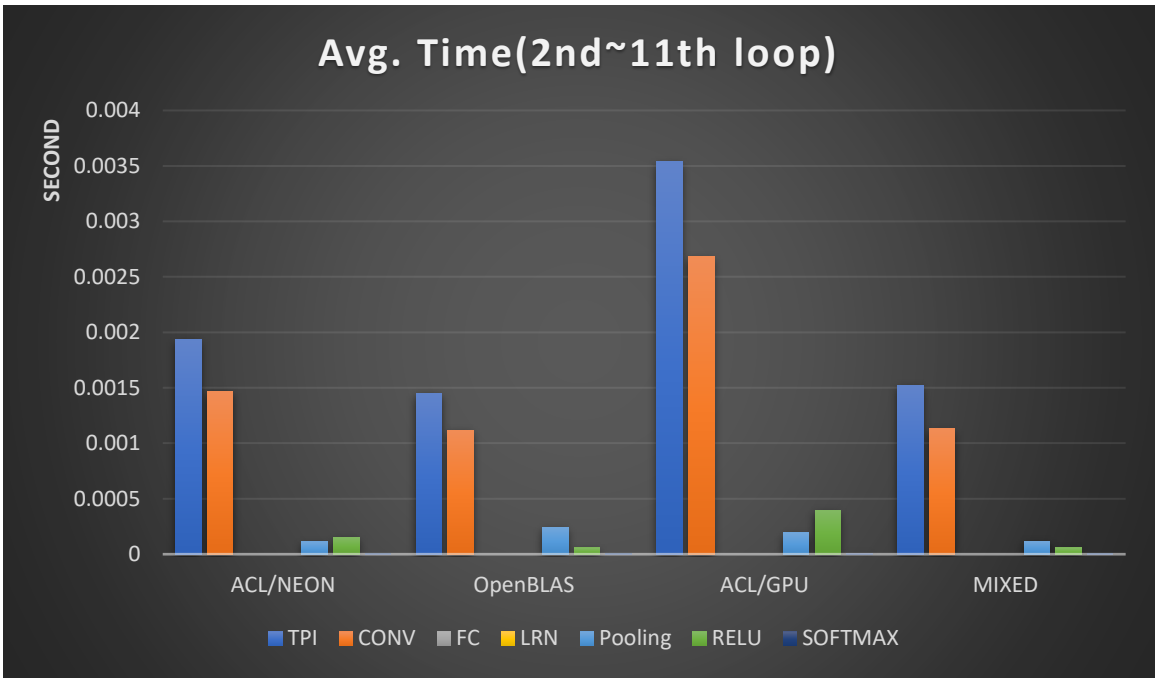
4.2 SqueezeNet

	TPI(s)	Allocate(s)	Run(s)	Config(s)	Copy(s)
1st					
ACL/NEON	0.4047	0.0462	0.1727	0.0911	0.0857

OpenBLAS	0.1664				
ACL/GPU	2.7437	0.0391	0.043	2.2475	0.4029
MIXED	0.1911	0.0142	0.019	0.0005	0.0189
Avg. Time					
ACL/NEON	0.1937		0.1587		0.0322
OpenBLAS	0.1453				
ACL/GPU	0.3536		0.0296		0.3201
MIXED	0.1522		0.0185		0.0135

	TPI(s)	CONV(s)	FC(s)	LRN(s)	Pooling(s)	RELU(s)	SOFTMAX(s)
1st							
ACL/NEON	0.4047	0.3185			0.0195	0.0324	0.0003
OpenBLAS	0.1664	0.1278			0.0259	0.0059	0.0002
ACL/GPU	2.7437	2.3222			0.1131	0.0941	0.1524
MIXED	0.1911	0.13			0.019	0.0061	0.0002
Avg. Time							
ACL/NEON	0.1937	0.1468			0.0115	0.0151	0.0001
OpenBLAS	0.1453	0.1119			0.0244	0.0059	0.0001
ACL/GPU	0.3536	0.268			0.0192	0.0396	0.0009
MIXED	0.1522	0.1129			0.0118	0.0059	0.0001



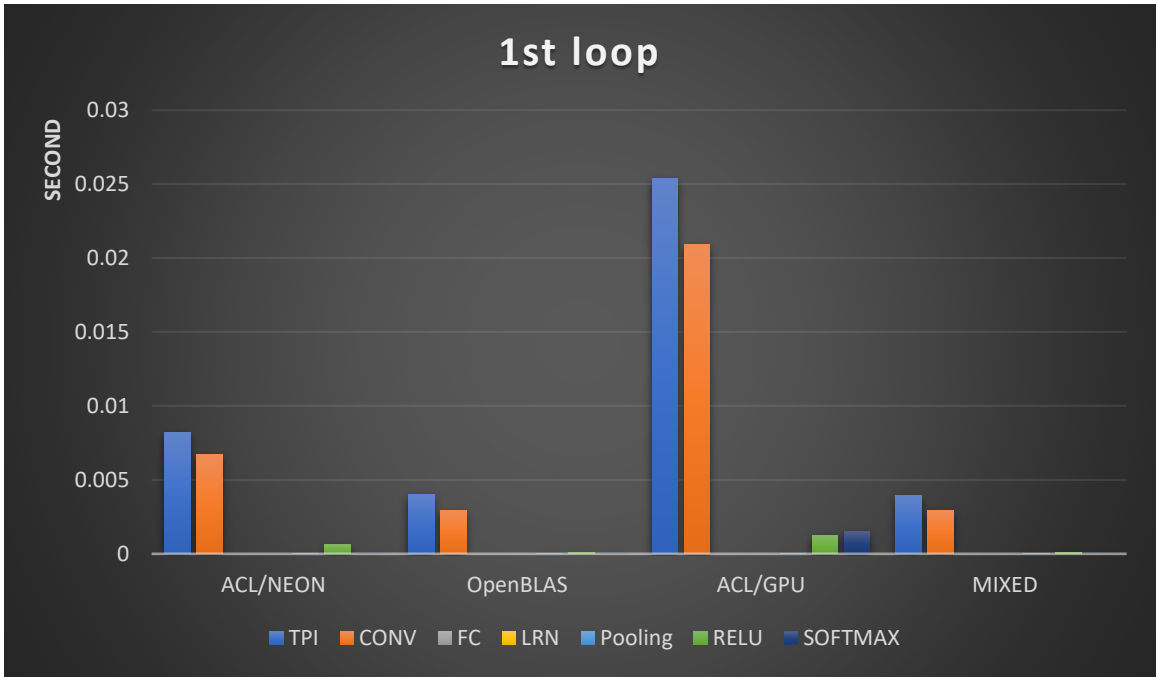


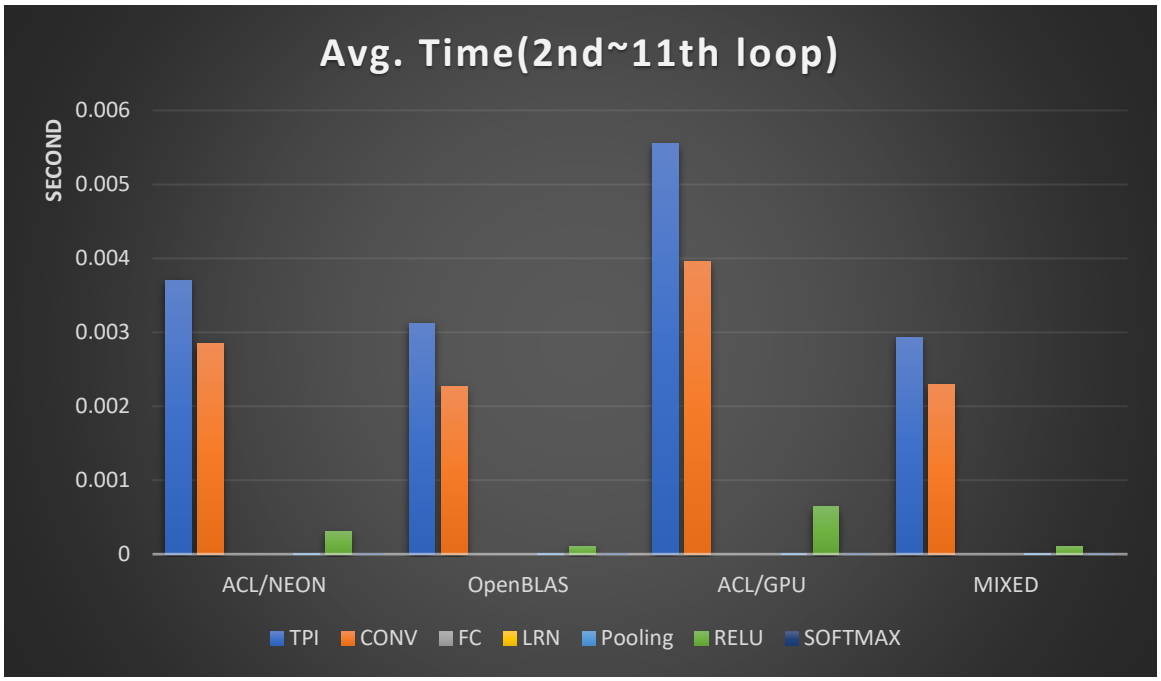
4.3 MobileNet

	TPI(s)	Allocate(s)	Run(s)	Config(s)	Copy(s)
1st					
ACL/NEON	0.8188	0.0824	0.2633	0.0777	0.2508
OpenBLAS	0.3992				
ACL/GPU	2.5371	0.0746	0.0404	1.5823	0.6937
MIXED	0.3917	0.0294	0.025	0.0008	0.0291
Avg. Time					
ACL/NEON	0.3705		0.2201		0.0595
OpenBLAS	0.3123				
ACL/GPU	0.5552		0.0398		0.424
MIXED	0.2936		0.024		0.0282

	TPI(s)	CONV(s)	FC(s)	LRN(s)	Pooling(s)	RELU(s)	SOFTMAX(s)
1st							
ACL/NEON	0.8188	0.6713			0.0005	0.0604	0.0003
OpenBLAS	0.3992	0.2919			0.0005	0.0111	0.0001

ACL/GPU	2.5371	2.0923			0.0005	0.1247	0.1501
MIXED	0.3917	0.2932			0.0006	0.0109	0.0003
Avg. Time							
ACL/NEON	0.3705	0.2845			0.0005	0.0305	0.0001
OpenBLAS	0.3123	0.2269			0.0005	0.0108	0.0001
ACL/GPU	0.5552	0.3964			0.0005	0.0641	0.0005
MIXED	0.2936	0.229			0.0005	0.0109	0.0001





5 Performance on Different Cores

The TPI is not very stable, it's in wide fluctuation. The data in the tables is lower limit of the range.

5.1 The TPI Data For ACL/NEON, OpenBLAS And Mixed Mode

AlexNet for ACL/NEON, OpenBLAS and mixed mode

	ACL/NEON(s)	OpenBLAS(s)	MIXED(s)
1xA53	1.9298	1.8578	0.937
1xA72	0.586	0.9388	0.5398
2xA72	0.3296	0.8891	0.4853
4xA53	0.6	1.6436	0.6795
2xA72+4xA53	0.4526	0.8996	0.5889

GoogleNet for ACL/NEON, OpenBLAS and mixed mode

	ACL/NEON(s)	OpenBLAS(s)	MIXED(s)
1xA53	1.2226	3.3659	1.2973
1xA72	0.6207	1.3918	0.4994
2xA72	0.4143	1.2366	0.3437

4xA53	0.8485	2.8061	0.6693
2xA72+4xA53*	0.6541	1.549	0.3604

SqueezeNet for ACL/NEON, OpenBLAS and mixed mode

	ACL/NEON(s)	OpenBLAS(s)	MIXED(s)
1xA53	0.4123	0.3677	0.3977
1xA72	0.1937	0.1453	0.1522
2xA72	0.1438	0.1015	0.1073
4xA53	0.3278	0.2244	0.2349
2xA72+4xA53*	0.1666	0.1034	0.2038

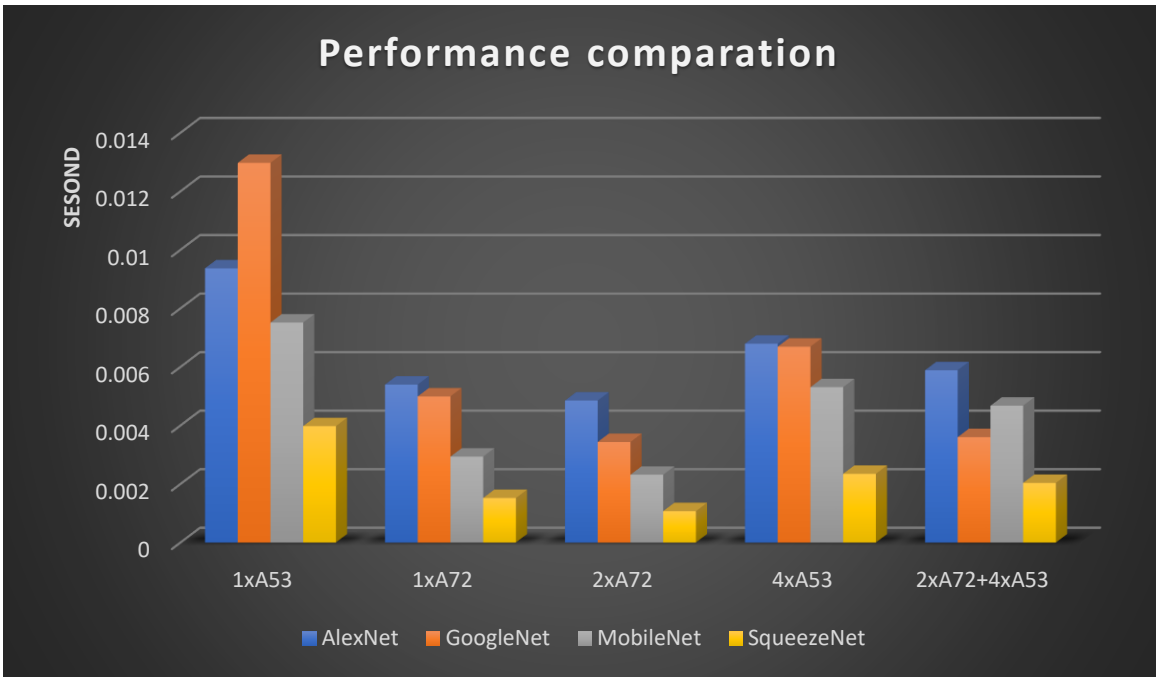
MobileNet for ACL/NEON, OpenBLAS and mixed mode

	ACL/NEON(s)	OpenBLAS(s)	MIXED(s)
1xA53	0.7918	0.8042	0.7517
1xA72	0.3705	0.3123	0.2936
2xA72	0.3202	0.2323	0.2318
4xA53	0.6156	0.5832	0.5309
2xA72+4xA53*	0.3871	0.2347	0.4677

5.2 The TPI In Mixed mode

The TPI data for different CPU cores in mixed mode:

	AlexNet	GoogleNet	MobileNet	SqueezeNet
1xA53	0.937	1.2973	0.7517	0.3977
1xA72	0.5398	0.4994	0.2936	0.1522
2xA72	0.4853	0.3437	0.2318	0.1073
4xA53	0.6795	0.6693	0.5309	0.2349
2xA72+4xA53	0.5889	0.3604	0.4677	0.2038



6 Conclusion

From the above test cases, we can deduce that:

- the performances of large FC are better under ACL_CL(GPU) than under NEON and OpenBLAS
- the performances of LRN are better under ACL_NEON than under OpenBLAS
- the performances of Pooling are better under ACL_NEON than under OpenBLAS

	AlexNet(s)	GoogleNet(s)	SquezzeNet(s)	MobileNet(s)
FC/ACL/NEON	0.3802	0.0063	0	0
FC/ACL/GPU	0.3447	0.0044	0	0
FC/OpenBLAS	0.0426	0.0017	0	0
LRN/ACL/NEON	0.0322	0.0547	0	0
LRN/OpenBLAS	0.4253	0.9151	0	0
Pooling/ACL/NEON	0.0046	0.0365	0.0118	0.0005
Pooling/OpenBLAS	0.0082	0.0820	0.0245	0.0005

However, for different cases, you may see different result for different layers by using ACL or OpenBLAS. Therefore, for applications, you can select best solution by combining ACL and OpenBLAS together.

7 Testing Issues

This section discusses some common issues when conducting the test.

7.1 Print Performance Log

In order to print performance log during testing, the value of USE_PROFILING should be set to 1 in ~/CaffeOnACL/makefile.config (the default value of USE_PROFILING is 0).

```
## Refer to http://caffe.berkeleyvision.org/installation.html
# Contributions simplifying and improving our build system ar

# cuDNN acceleration switch (uncomment to build with cuDNN).
# USE_CUDNN := 1

# CPU-only switch (uncomment to build without GPU support).
CPU_ONLY := 1

USE_PROFILING := 1

USE_ACL :=1
ACL_ROOT :=/home/firefly/ComputeLibrary
ACL_INCS :=$(ACL_ROOT)/include
ACL_INCS +=$(ACL_ROOT)
ACL_LIBS_DIR :=$(ACL_ROOT)/build
ACL_LIBS_DIR +=$(ACL_ROOT)/build/arm_compute
ACL_LIBS :=arm_compute arm_compute_core OpenCL

# uncomment to disable IO dependencies and corresponding data
# USE_OPENCV := 0
# USE_LEVELDB := 0
# USE_LMDB := 0
```

Set USE_PROFILING to 1 in makefile.config

After setting the value, please *make clean* in ~/CaffeOnACL and then *make all* and *make distribute* to recompile caffe. When compilation is complete, please copy libcaffe.so and libcaffe.so.1.0.0-rc5 to /usr/lib again (this step is significant, or some unreadable code may be outputted).

```
sudo cp ~/CaffeOnACL/distribute/lib/libcaffe.so /usr/lib
sudo cp ~/CaffeOnACL/distribute/lib/libcaffe.so.1.0.0-rc5 /usr/lib
```

Copy libcaffe.so and libcaffe.so.1.0.0-rc5 to /usr/lib again

By now some results should be printed when running a test. If all performance data is required, please set LOGACL to 0xffffffff by *export LOGACL=0xffffffff* (using 0xffff may lead to incomplete results in MobileNet).

7.2 Test on Different Cores

To conduct tests on different CPU cores, please first set OPENBLAS_NUM_THREADS to the number of CPU cores by *export OPENBLAS_NUM_THREADS=X* (X equals to the number of cores you want). Then *taskset -a mask* command can be used to specify cores to run test. The masks of cores are listed below.

Mask	Cores
0x10	1xA72
0x30	2xA72
0x1	1xA53
0xf	4xA53
0x3f	2xA72 + 4xA53

Mask of different CPU cores

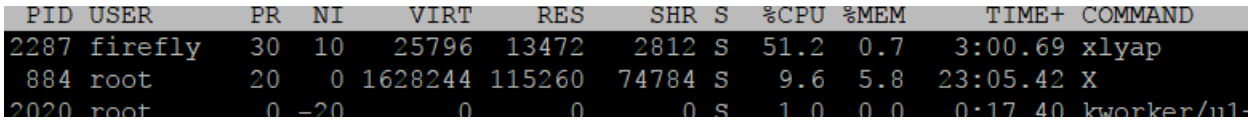
For example, the code below will run a test on AlexNet using Neon+OpenBlas on 2xA72 and 4xA53 CPU cores.

```
echo "AlexNet(Neon+OpenBlas_2xA72+4xA53)"
export OPENBLAS_NUM_THREADS=6
export BYPASSACL=0x14c
export LOGACL=0xffffffff
taskset -a
0x3f ./distribute/bin/classification_profiling.bin ./models/bvlc_alexnet/deploy.prototxt ./models/bvl
c_alexnet/bvlc_alexnet.caffemodel data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
examples/images/cat.jpg > ./log/Alexnet1_014c_2xA72+4xA53.log
```

7.3 Influencing Factors of Results

There are some factors that will make impacts on the performance of tests. Therefore, some settings should be checked before conducting tests.

First of all, lightdm is running on default on RK3399 which will slow down the system. To check if lightdm is running, please use *top* command to see if there is a process that has an “X” in COMMAND column.



Lightdm is running

Lightdm can be stopped by *sudo service lightdm stop* if it is running.

Secondly, different CPU frequencies may lead to different test results. To check CPU frequency, one possible approach is given below:

```
1) Get root shell
sudo -s

2) Set CPU policy
echo performance > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
echo performance > /sys/devices/system/cpu/cpufreq/policy4/scaling_governor

3) Check CPU frequency
cat /sys/devices/system/cpu/cpufreq/policy0/scaling_cur_freq
cat /sys/devices/system/cpu/cpufreq/policy4/scaling_cur_freq

4) Check available CPU frequency
cat /sys/devices/system/cpu/cpufreq/policy0/scaling_available_frequencies
cat /sys/devices/system/cpu/cpufreq/policy4/scaling_available_frequencies
```

Finally, different POSIX clocks may also lead to a relatively small difference in the test results. Within ~/CaffeOnACL/src/caffe/layer.cpp, please set the POSIX clock from CLOCK_MONOTONIC_COARSE to CLOCK_MONOTONIC.