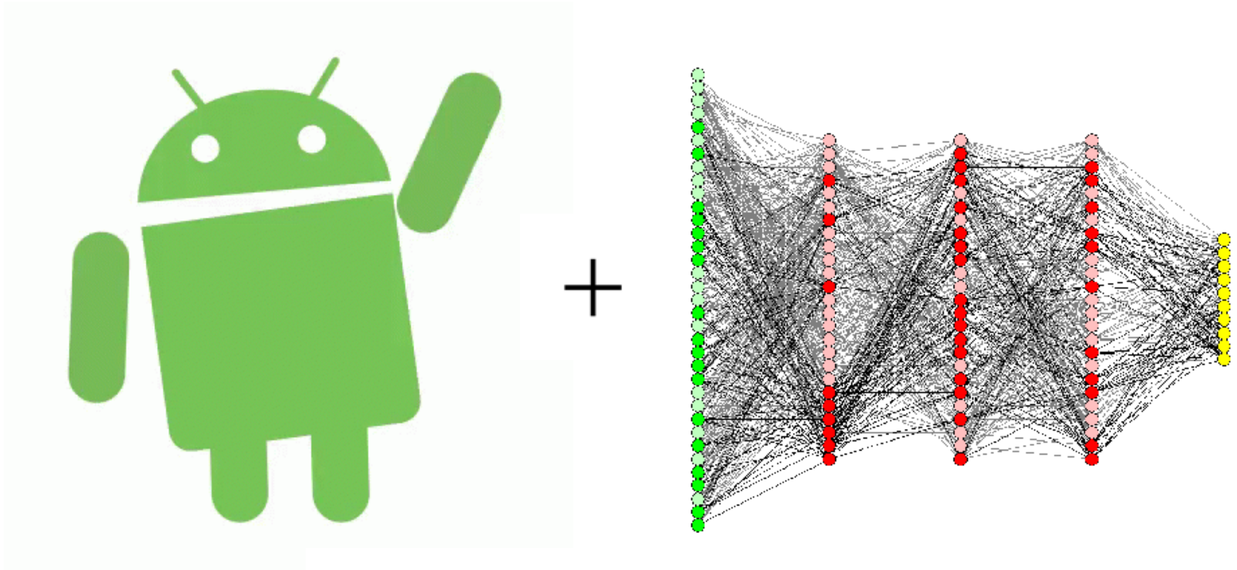
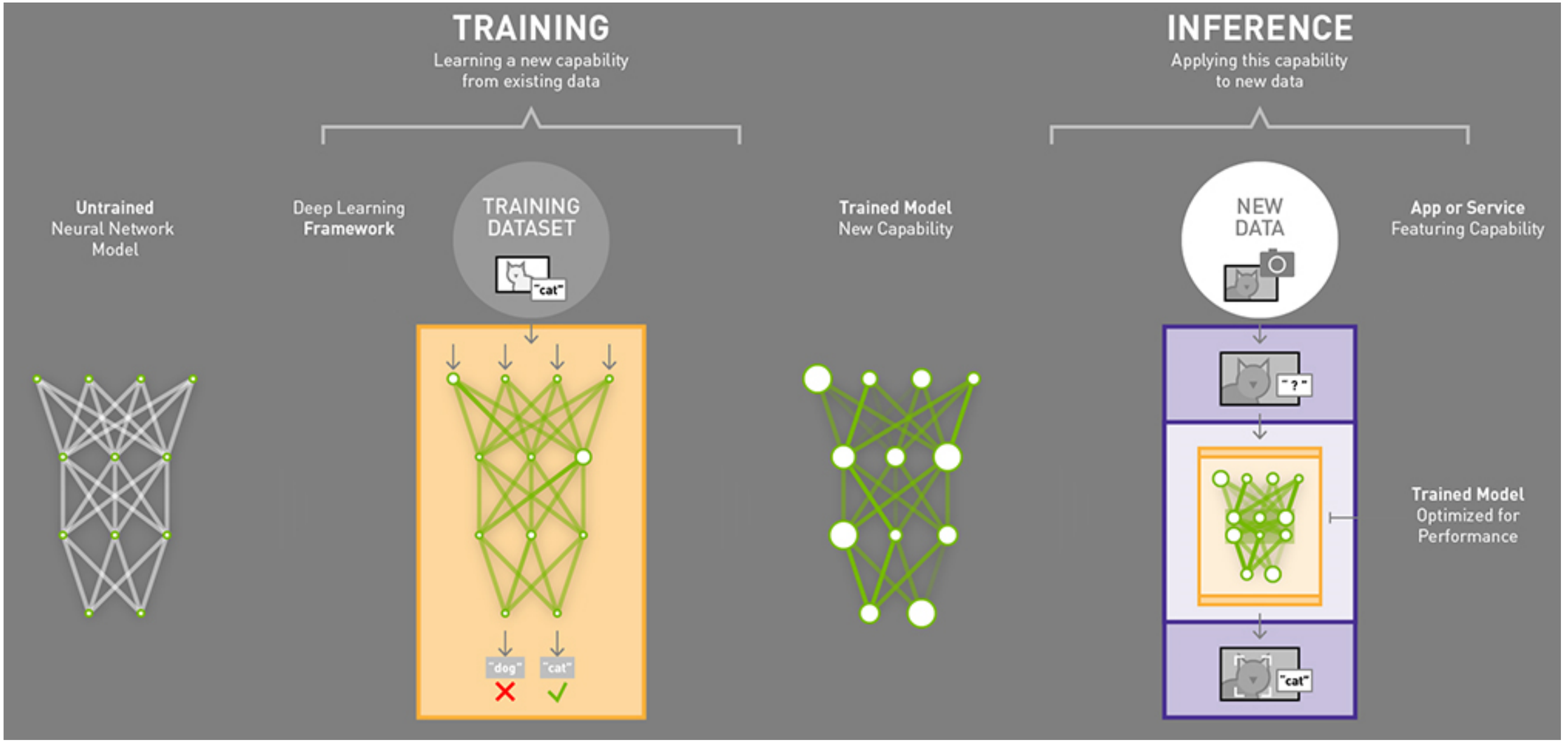
Мир и согласие: Android и нейросети

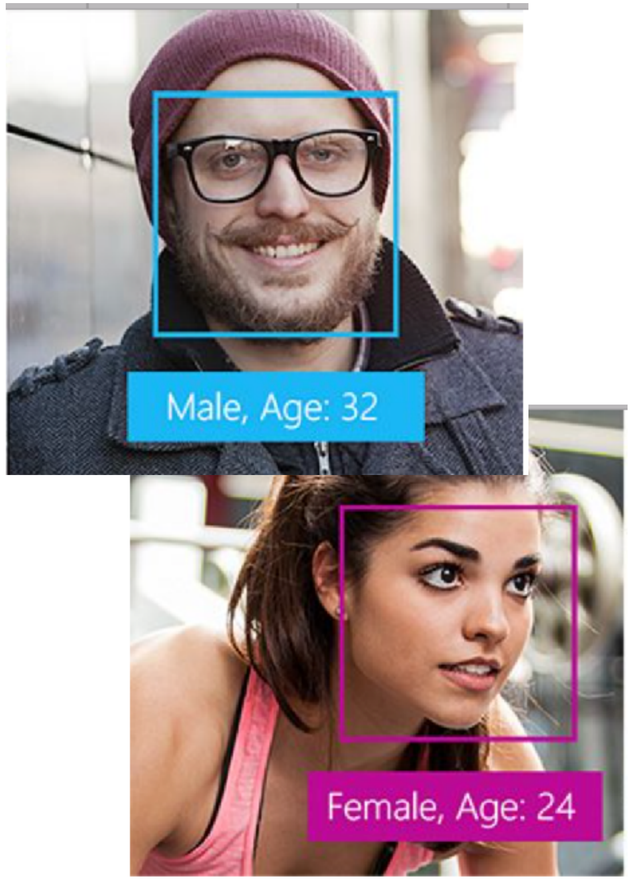


Тарасов Вячеслав
CTO Exploud
@vyacheslav_tarasov



Graphic source: [Nvidia Blog]

1. Classification



2. Segmentation + matting

1) Input

Takes an initial image and reduces it’s size to

2) Segmentation

Neural network trained on thousands of portrait photo predicts a rough mask on an image highlighting a human

3) Matting

Another Neural network is run on the edges of the segmentation mask on 2000x2000 image in order

4) Output

Applies the mask to the image of the original

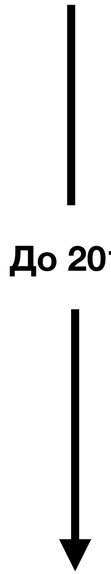
Основные критерии качества приложений под Android

- Графический дизайн и взаимодействие с пользователем
- Функциональность
- Производительность и стабильность
- Соответствие требованиям Google Play

Frameworks

	Название	Язык	ОС	FC NN	CNN	AE	RBM
1	DeepLearnToolbox	Matlab	Windows, Linux	+	+	+	+
2	Theano	Python	Windows, Linux, Mac	+	+	+	+
3	Pylearn2	Python	Linux, Vagrant	+	+	+	+
4	Deepnet	Python	Linux	+	+	+	+
5	Deepmat	Matlab	?	+	+	+	+
6	Torch	Lua, C	Linux, Mac OS X, iOS, Android	+	+	+	+
7	Darch	R	Windows, Linux	+	—	+	+
8	Caffe	C++, Python, Matlab	Linux, OS X	+	+	—	—
9	nnForge	C++	Linux	+	+	—	—
10	CXXNET	C++	Linux	+	+	—	—
11	Cuda-convnet	C++	Linux, Windows	+	+	—	—
12	Cuda CNN	Matlab	Linux, Windows	+	+	—	—
13	TensorFlow	C/C++, Python	Linux, Mac OS X, iOS, Android	+	+	+	+
14	PyTorch	Python	Linux, Mac OS X	+	+	+	+
15	Caffe2Go	Python	MacOS X Ubuntu CentOS Windows iOS Android Raspbian Tegra	+	+	+	+

Inference&Benchmarks



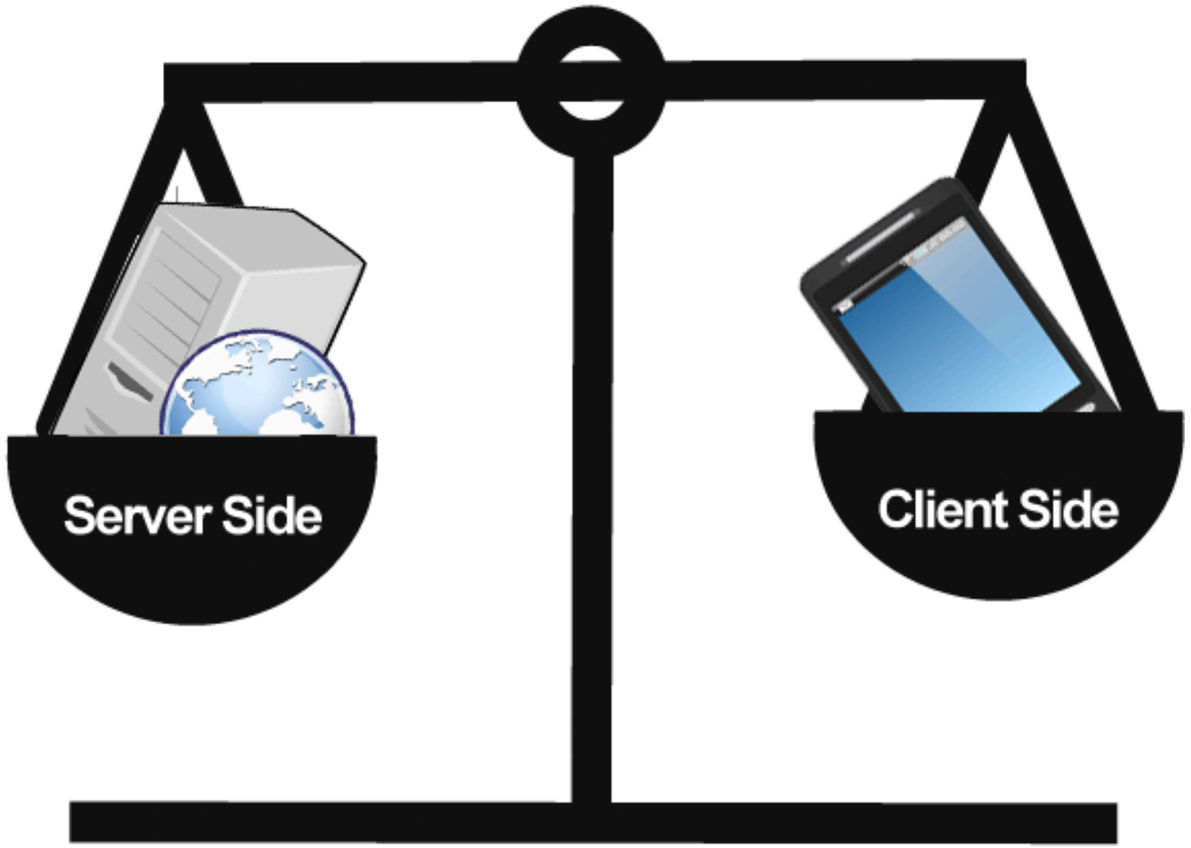
До 2016

- Only server side

Benchmarks

2016-2018

style transfer + segmentation 605*405						
model		min, sec	max, sec	avg, sec	android version	processor
mi5	Xiaomi	4.45	5.23	4.84	7.1	Snapdragon 820 MSM8996
one+	one	23.413	20.519	21.966	7.1	Qualcomm Snapdragon 801
a7 galaxy	samsung	39.17	40.358	39.764	6.0.1	Exynos 7880
s6 galaxy	samsung	4.328	5.146	4.737	7	Exynos 7420
expire z3	sony	22.709	23.138	22.9235	6.0.1	Qualcomm Snapdragon 801
galaxy a3	samsung	54.104	55.137	54.6205	5.0.2	Exynos 7870
redmi note 3	xiaomi	не запусился, система блочит		-		Qualcomm Snapdragon 650
s8	samsung	3.268	3.425	3.3465	7.1	Qualcomm Snapdragon 835
leon	lg	Failure [INSTALL_FAILED_INSUFFICIENT_STORAGE]			5.0.1	
a5	samsung	4.756	4.436	4.596	5.0.1	Exynos 7580
galaxy s4	samsung	38.874	49.471	44.1725	5.0.1	Exynos 5410
Zenphone 2 551kl	asus	5.453	5.84	5.6465	6.0.1	Intel Atom Z3580
Yota phone 2	yota	40.903	43.048	41.9755	4.4.3	Qualcomm Snapdragon 800
ht3	homtom	45.583	50.442	48.0125	5.0.1	MediaTek MT6580
galalaxy a5	samsung	26.241	26.576	26.4085	7	Exynos 7880
s3	samsung	-Didn't find class "com.qualDidn't find class "com.qual comm.qti.Performance" on path: DexPathList[[],nativeLibraryDirectori			5.1.1	Exynos 4412
Zenphone 2	asus	8.494	12.61	10.552	5.1.1	Intel Atom Z3580
honor 7	huawei	6.786	7.12	6.953	5.0.2	HiSilicon Kirin 935
m3 note	meizy	8.33	9.128	8.729	5.1.1	MediaTek Helio P10 (MT6755)
honor 4c	huawei	10.234	10.931	10.5825	5.1.1	HiSilicon Kirin 620
note 2	xiaomi	3.581	3.944	3.7625	7	Snapdragon 821 MSM 8996 Pro


Server Side
Client Side

On-device inferencing benefits

Latency: вам не нужно отправлять запрос по сетевому соединению и ждать ответа

Availability: приложение работает даже при отсутствии интернет соединения

Speed: современные телефоны обеспечивают значительно более быстрые вычисления

Privacy: Данные не покидают устройство

Cost: вы не платите за сервера

Trade-offs

System utilization: потребление энергии аккумулятора

Application size: минимизируем вес моделей в вашем приложении. Если не удалось, загружаем их после установки приложения

Server-side inference: “тяжелые” модели лучше запускать в облаке

Поддержка архитектур

- arm64-v8a

- armeabi-v7a

- x86

- x86_64

Inference&Benchmarks

End of
2018-2019

AnTuTu releases a benchmark to test the AI performance of smartphones

Show Prototypes / Dev Platform Scores

Model	CPU	RAM	Year	Android	Updated	Test 1a, ms (C)	Test 1b, ms (F)	Test 1c, ms (Q)	Test 2, ms (F)	Test 3, ms (Q)	Test 4, ms (F)	Test 5, ms (Q)	Test 6, ms (C)	Test 7, ms (C)	Test 8, ms (F)	Test 9, px (F)	AI-Score
Xiaomi Mi 9	Snapdragon 855	8GB	2019	9	3.19	97	21	8	70	44	58	257	1513	287	142	10	20964 [3]
Lenovo Z5 Pro GT	Snapdragon 855	6GB	2018	9	3.19	105	22	8	70	43	59	257	1633	290	142	10	20829 [3]
Samsung Galaxy S10+	Snapdragon 855	8GB	2019	9	3.19	59	23	8	72	43	63	258	1547	289	143	10	20606 [3]
Samsung Galaxy S10	Snapdragon 855	6GB	2019	9	3.19	56	25	8	73	43	64	257	1538	291	145	10	20439 [3]
Huawei Mate 20 Pro	HiSilicon Kirin 980 (NPU)	6GB	2018	9	3.19	53	20	94	51	749	54	662	1595	310	1151	20	17391 [1,10]
Huawei Mate 20 X	HiSilicon Kirin 980 (NPU)	6GB	2018	9	3.19	53	20	91	51	752	54	655	1574	314	1138	20	17346 [1,10]
Huawei Mate 20	HiSilicon Kirin 980 (NPU)	4GB	2018	9	3.19	53	20	94	51	738	54	665	1588	316	1160	20	17236 [1,10]
Samsung Galaxy S9+	Snapdragon 845	6GB	2018	9	1.19	121	26	10	83	90	69	349	2138	349	174	10	16808 [3]
Asus Zenfone 5z	Snapdragon 845	6GB	2018	9	3.19	79	15	8	148	88	157	420	2210	346	184	10	14229 [3]
OnePlus 6T	Snapdragon 845	8GB	2018	9	12.18	130	27	10	160	89	217	393	2109	348	218	10	12358 [3]
OnePlus 6	Snapdragon 845	8GB	2018	9	12.18	125	27	11	160	89	220	394	2087	347	218	10	12340 [3]
Huawei P20 Pro	HiSilicon Kirin 970 (NPU)	6GB	2018	9	12.18	187	46	194	84	916	88	1121	3173	582	1554	6	8921 [1]
Huawei P20	HiSilicon Kirin 970 (NPU)	4GB	2018	9	12.18	196	45	167	82	920	89	1164	3464	607	1531	6	8908 [1]

Processor	Cores	Year	CPU Q AI Score	CPU F AI Score	QUANT Score	QUANT Accuracy	FP16 Score	FP16 Accuracy	FP32 Score	FP PAR Score	Accuracy	AI-Score
Mediatek Helio P90	CPU (4xCortex-A75 + 4xCortex-A55) + DSP x 2 + APU	2018	1054	2012	6212	98	9910	95	158	46	96	19496 [1,4]
Snapdragon 855	CPU (8xKryo) + DSP (Hexagon 690) + GPU (Adreno 640)	2018	1988	3598	3695	55	7361	37	1158	831	43	18924 [1]
HiSilicon Kirin 980	CPU (4xCortex-A76 + 4xCortex-A55) + NPU x 2 / n.a.	2018	1817	3447	222	60	10750	85	139	64	76	16684 [2]
Snapdragon 845	CPU (8xKryo) + DSP (Hexagon 685) + GPU (Adreno 630)	2018	1580	2176	2028	58	6298	38	929	701	45	13868 [1]
Spreadtrum ud710	n.a.	2019	1371	1497	112	60	7540	59	67	30	59	10773 [2,4]
Exynos 9820 Octa	CPU (2xM4 & 2xA75 & 4xA55) + GPU (Mali-G76 MP12)	2018	1737	2021	960	99	2244	99	186	139	99	7288 [1]
HiSilicon Kirin 970	CPU (4xCortex-A73 & 4xCortex-A53) + NPU / n.a.	2017	1286	1976	166	60	3431	58	132	51	59	7147 [2]
Snapdragon 845	CPU (8xKryo 385 Gold&Silver) + DSP (Hexagon 685)	2018	1667	2216	1724	58	851	99	130	47	85	6752 [3]
Mediatek Helio P60	CPU (4xA73 + 4xA53) + GPU (Mali-G72 MP3) + APU	2018	1151	1765	1334	99	1266	70	113	152	79	5806 [1]
Snapdragon 675	CPU (8xKryo 460 Gold&Silver) + DSP (Hexagon 685)	2018	1015	2000	1761	88	739	99	108	44	95	5785 [3]
Snapdragon 710	CPU (8xKryo 360 Gold&Silver) + DSP (Hexagon 685)	2018	870	1508	1980	58	715	99	109	45	85	5329 [3]
Exynos 9810 Octa	8 (4x2.7 GHz Mongoose M3 & 4x1.8 GHz Cortex-A55)	2018	1301	1843	148	60	1160	99	30	46	86	4530
Exynos 9610	CPU (4xA73 & 4xA53) + GPU (Mali-G72 MP3)	2018	1173	1423	279	99	1212	99	197	82	99	4385 [2]
Snapdragon 835	8 (4x2.45 GHz Kryo 280 & 4x1.9 GHz Kryo 280)	2017	1199	1697	164	60	772	99	126	46	86	4129
Snapdragon 820	4 (2x2.15 GHz Kryo & 2x1.6 GHz Kryo)	2016	957	1820	133	60	699	99	96	65	86	3874

The benchmark consists of 21 Computer Vision and AI Tests performed by Neural Networks running on your smartphone. It is measuring over 50 different aspects of AI performance, including the speed, accuracy, initialization time, etc. Considered Neural Networks comprise a comprehensive range of architectures allowing to assess the performance and limits of various approaches used to solve different AI tasks. A detailed description of 11 Benchmark Sections is provided below.

Task for tests

- Section 1. Classification, MobileNet-V2: CPU (FP16) + NPU / GPU (FP16) + NPU / GPU / DSP (INT8)
- Section 2. Classification, Inception-V3: CPU (FP16) + NPU / GPU (FP16) + NPU / GPU / DSP (INT8)
- Section 3. Face Recognition, Inception-ResNet-V1: CPU (INT8) + NPU / GPU (FP16) + NPU / GPU / DSP (INT8)
- Section 4. Playing Atari Games, LSTM: CPU (FP16)
- Section 5. Deblurring, SRCNN: NPU / GPU (FP16) + NPU / GPU / DSP (INT8)
- Section 6. Super-Resolution, VGG19: NPU / GPU (FP16) + NPU / GPU / DSP (INT8)
- Section 7. Super-Resolution, SRGAN: CPU (FP32) + CPU (INT8)
- Section 8. Bokeh Simulation, U-Net: CPU (FP32)
- Section 9. Semantic Segmentation, ICNet: NPU / GPU (FP32) x 2 - two CNNs running in parallel
- Section 10. Image Enhancement, DPED ResNet: NPU / GPU (FP16) + NPU / GPU (FP32)
- Section 11. Memory limits, SRCNN: NPU / GPU (FP16)

End of 2018

Huawei P20 Pro	HiSilicon Kirin 970 (NPU)	6GB RAM	6519
Huawei Honor 10	HiSilicon Kirin 970 (NPU)	4GB RAM	6496
HTC U12 Plus	Snapdragon 845	6GB RAM	1708
Asus Zenfone 5z	Snapdragon 845	6GB RAM	1698
Samsung Galaxy S9 Plus	Exynos 9810 Octa	6GB RAM	1628
Samsung Galaxy Note8	Exynos 8895 Octa	6GB RAM	1512
Sony Xperia XZ2	Snapdragon 845	4GB RAM	1499
LG G7 ThinQ	Snapdragon 845	4GB RAM	1464
Xiaomi Mi Mix 2S	Snapdragon 845	6GB RAM	1426
Samsung Galaxy S8	Exynos 8895 Octa	4GB RAM	1413
Motorola Z2 Force	Snapdragon 835	6GB RAM	1384
HTC U11 Plus	Snapdragon 835	6GB RAM	1352
Essential Phone	Snapdragon 835	4GB RAM	1337
OnePlus 3T	Snapdragon 821	6GB RAM	1302
Lenovo ZUK Z2 Pro	Snapdragon 820	6GB RAM	1300
Google Pixel 2	Snapdragon 835	4GB RAM	1293

Beginning of 2018

Huawei P20 Pro	HiSilicon Kirin 970 (NPU)	6GB RAM	8921
Huawei P20	HiSilicon Kirin 970 (NPU)	4GB RAM	8908

Processor	Cores	Year	Test 1a, ms (C)	Test 1b, ms (F)	Test 1c, ms (Q)	Test 2, ms (F)	Test 3, ms (Q)	Test 4, ms (F)	Test 5, ms (Q)	Test 6, ms (C)	Test 7, ms (C)	Test 8, ms (F)	AI-Score
Snapdragon 855	CPU (8xKryo) + DSP (Hexagon 690) + GPU (Adreno 640)	2018	100	20	8	70	42	60	30	1493	284	142	19769 [3,5]
Mediatek Helio P90	CPU (4xCortex-A75 + 4xCortex-A55) + DSP x 2 + APU	2018	82	8	3	103	37	46	82	2402	376	991	18231 [3,4]
Snapdragon 855	CPU (8xKryo) + DSP (Hexagon 690) + GPU (Adreno 640)	2018	77	23	8	71	42	59	255	1638	292	143	15453 [3]
HiSilicon Kirin 980	CPU (4xA76 + 4xA55) + NPU x 2 + GPU (Mali-G76 MP10)	2018	54	20	19	51	113	56	140	1541	275	347	14646 [3,5]
Snapdragon 845	CPU (8xKryo) + DSP (Hexagon 685) + GPU (Adreno 630)	2018	121	25	10	82	90	69	349	2130	349	174	12508 [3]
HiSilicon Kirin 980	CPU (4xCortex-A76 + 4xCortex-A55) + NPU (x2 / n.a.	2018	51	10	86	50	741	58	664	1682	300	1155	11843 [1]

1 - This SoC provides native acceleration support for float AI models (via NNAPI)

2 - This SoC provides native acceleration support for quantized AI models (via NNAPI)

3 - This SoC provides acceleration support for both float and quantized AI models (via NNAPI)

4 - This SoC might be using unofficial / prototype hardware or drivers

5 - These are the results of an early prototype. The results of the commercial SoC might be different

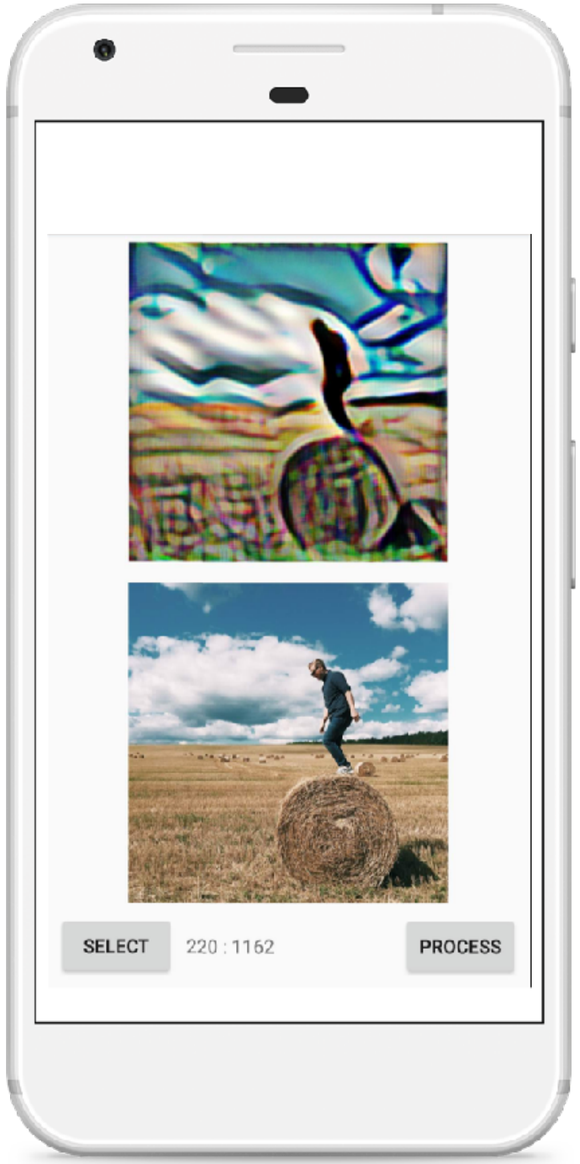
Name			GPU	Частота GPU	Время теста
Snapdragon 410	4/1.2 ГГц		Adreno 306	450 МГц	47.5825
Snapdragon 650	2/1.8 ГГц	4/1.2 ГГц	Adreno 510	600 МГц	-------
Snapdragon 800	4/2.3 ГГц		Adreno 330	450 МГц	41.9755
Snapdragon 801	4/2.5 ГГц		Adreno 330	578 МГц	21.966
Snapdragon 820	2/2.15 ГГц	2/1.6 ГГц	Adreno 530	624 МГц	4.84
Snapdragon 821	4/2.35 ГГц		Adreno 530	624 МГц	
Snapdragon 835	4/2.45 ГГц	4/1.9 ГГц	Adreno 540	710 МГц	
Intel Atom Z3580	4/2.3 ГГц		PowerVR GX6430	450 МГц	
HiSilicon Kirin 620	8/1.2 ГГц		Mali-450 MP4	700 МГц	
HiSilicon Kirin 935	8/1.6 ГГц		Mali-T628	600 МГц	
MediaTek MT6580	4/1.3 ГГц		Mali-400 MP2	500 МГц	
MediaTek MT6737	4/1.25 ГГц		Mali-T720 MP2	650 МГц	
MediaTek Helio P10 (MT6755)	8/2 ГГц		Mali-T860 MP2	700 МГц	
MediaTek Helio X20	10/2.1 ГГц		Mali-T880 MP4	700 МГц	5.3265
Exynos 4412	4/1.4 ГГц		Mali-400 MP4	500 МГц	-------
Exynos 5410	8/1.6 ГГц		PowerVR SGX544MP3	533 МГц	44.1725
Exynos 7420	4/2.1 ГГц	4/1.5 ГГц	Mali-T760 MP8	772 МГц	4.737
Exynos 7580	8/1.6 ГГц		Mali-T720 MP2	600 МГц	4.596
Exynos 7870	8/1.6 ГГц		Mali-T830 MP3	950 МГц	54.6205
Exynos 7880	8/1.9 ГГц		Mali-T830 MP3	950 МГц	39.764

Architecture tuning

models		7	22_1	23	24a	a1	m7	qq6
htc 630	6	13.066	14.136	15.854	16.014	14.353	18.769	14.372
expire z3	5.0.2	3.796	3.796	3.755	3.755	3.755	3.755	3.755
lenovo p70	4.4	5.268	5.268	5.268	5.268	5.268	5.34	5.34
galaxy a5 (2017)	7.0.1	33.646	3.339	3.393	3.306	3.447	3.155	14.209
mi5	7.1	3.13	3.28	3.361	3.206	3.395	3.129	3.39
samsung s8	7.1	2.883	2.974	2.691	2.386	2.162	3.225	2.834
htc 1S	Failure [INSTALL_FAILED_INSUFFICIENT_STORAGE]							
meizu pro5	5.1	25.754	27.718	27.683	25.973	25.426	26.418	25.538
dexp 245	5.1	26.522	214.578	195.713	171.663	161.192	176.598	184.341
sone expire x	7.1.1	14.552	12.537	13.973	12.075	14.151	13.167	13.158
asus z601	6	24.208	28.514	23.535	24.915	25.974	28.988	28.622
asus z010	6.0.1	58.374	67.54	63.058	61.872	64.844	55.539	60.884

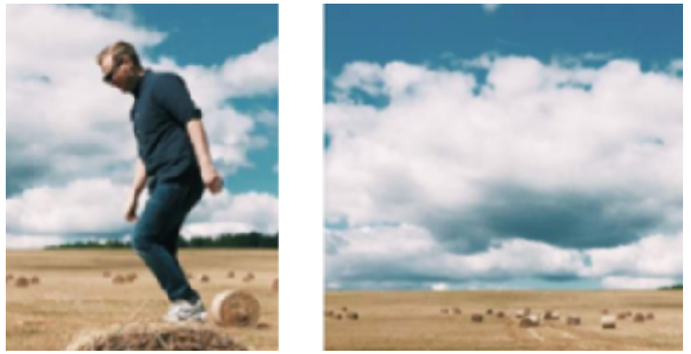
Memory limits.
Tiling

Input 256x256

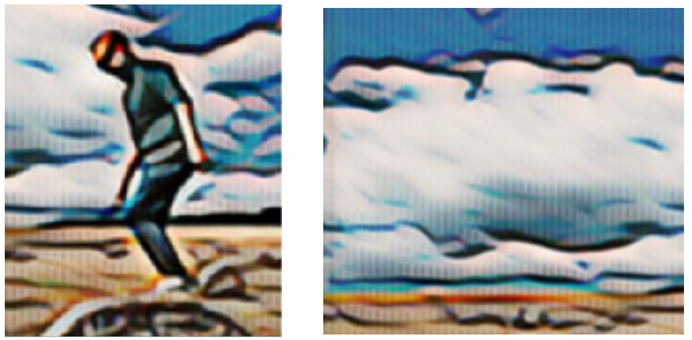


Input 1200x1200

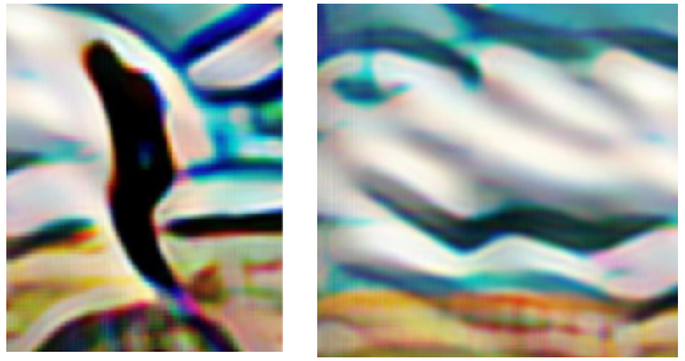
Original



1200x1200



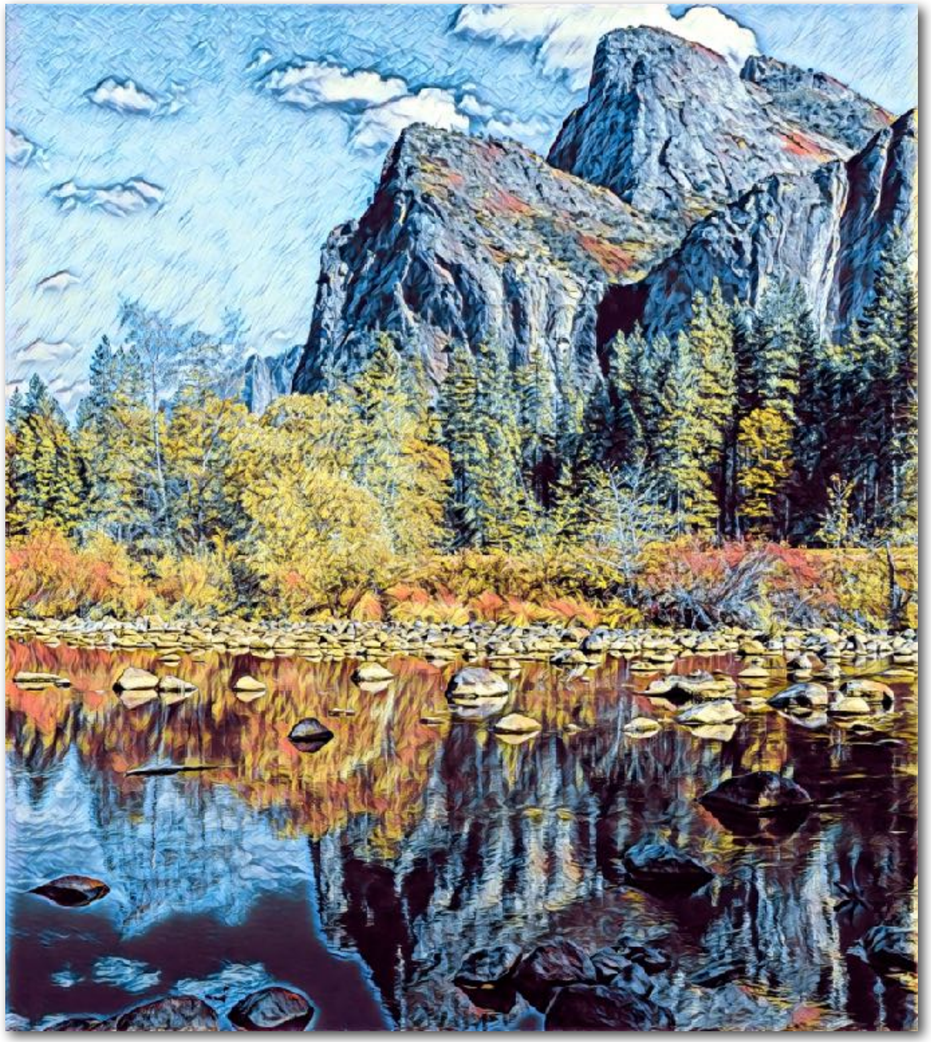
256x256



Benchmarks with different input resolution

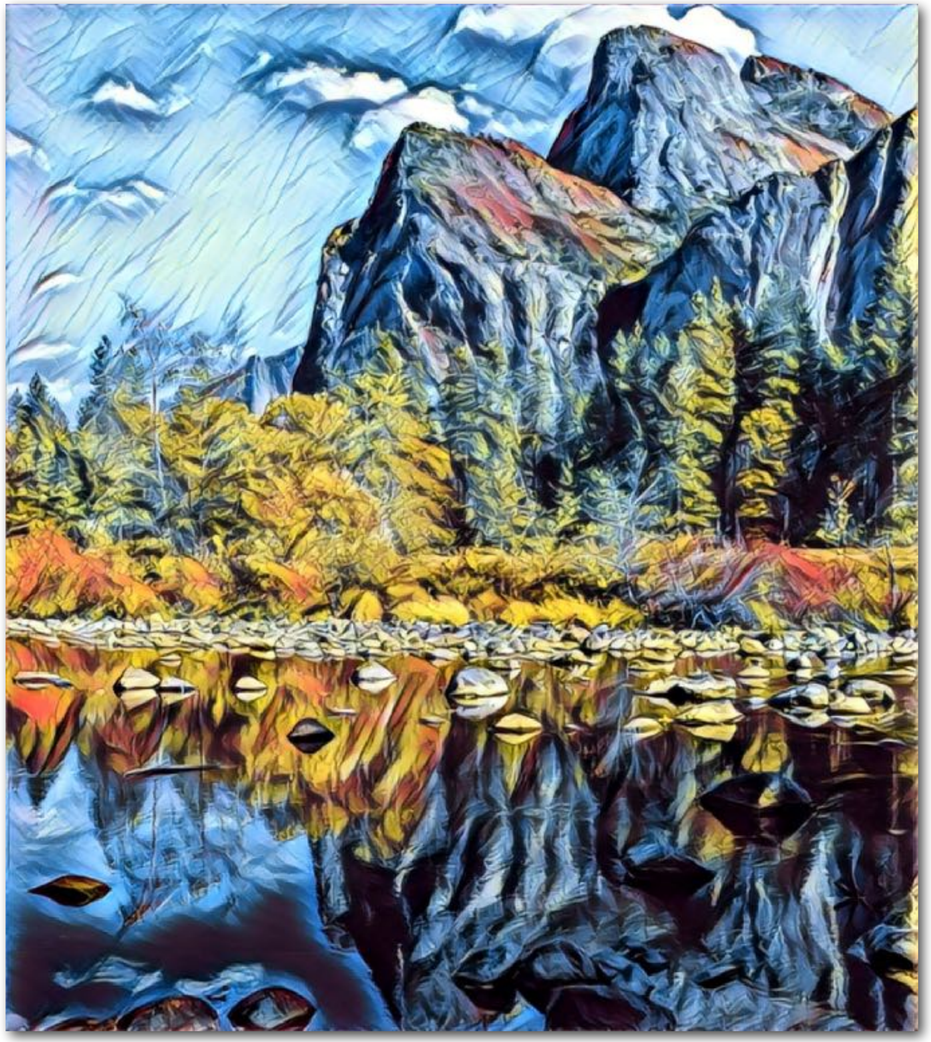
style transfer				
Processor	**GPU**	605x405, sec	1000x1000, sec	3200x2000, sec
Snapdragon 210 MSM8909	Adreno 304	26.522	186.484	489.958
Snapdragon 400	Adreno 305	15.223	101.382	324.802
Snapdragon 410	Adreno 306	12.780	68.382	273.528
Snapdragon 616 MSM8939v2	Adreno 405	8.970	26.394	118.772
Snapdragon 650	Adreno 510	5.410	21.730	91.266
Snapdragon 801	Adreno 330	3.767	11.238	56.190
Snapdragon 820	Adreno 530	3.270	9.889	39.556
Snapdragon 835	Adreno 540	2.736	6.087	31.044
MediaTek MT6752	Mali-T760 MP2	5.289	8.905	38.292
Exynos 7880	Mali-T830 MP3	9.214	18.250	109.497
Exynos 7420	Mali-T760 MP8	11.756	39.764	186.891

Input 2400x2100

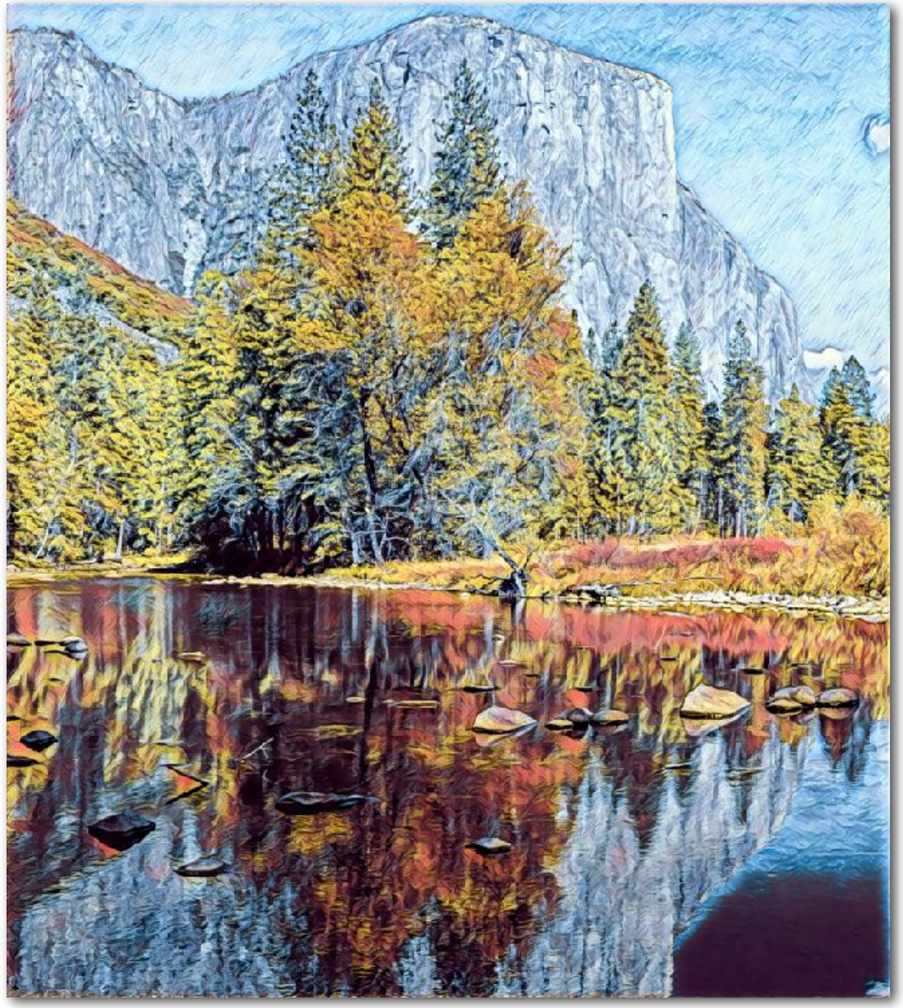


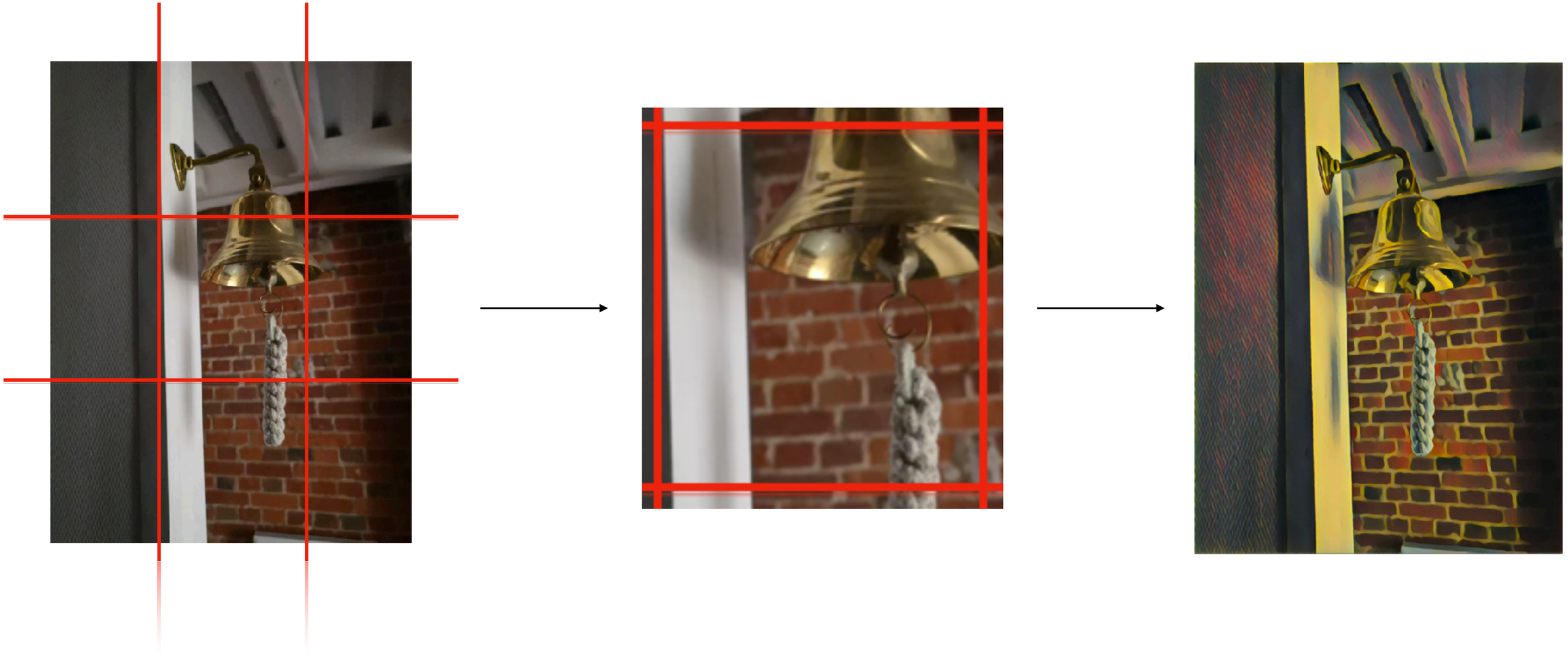
- P20 pro Huawei - 2.32 sec
- Pixel - 8.95 sec

Input 1080*950

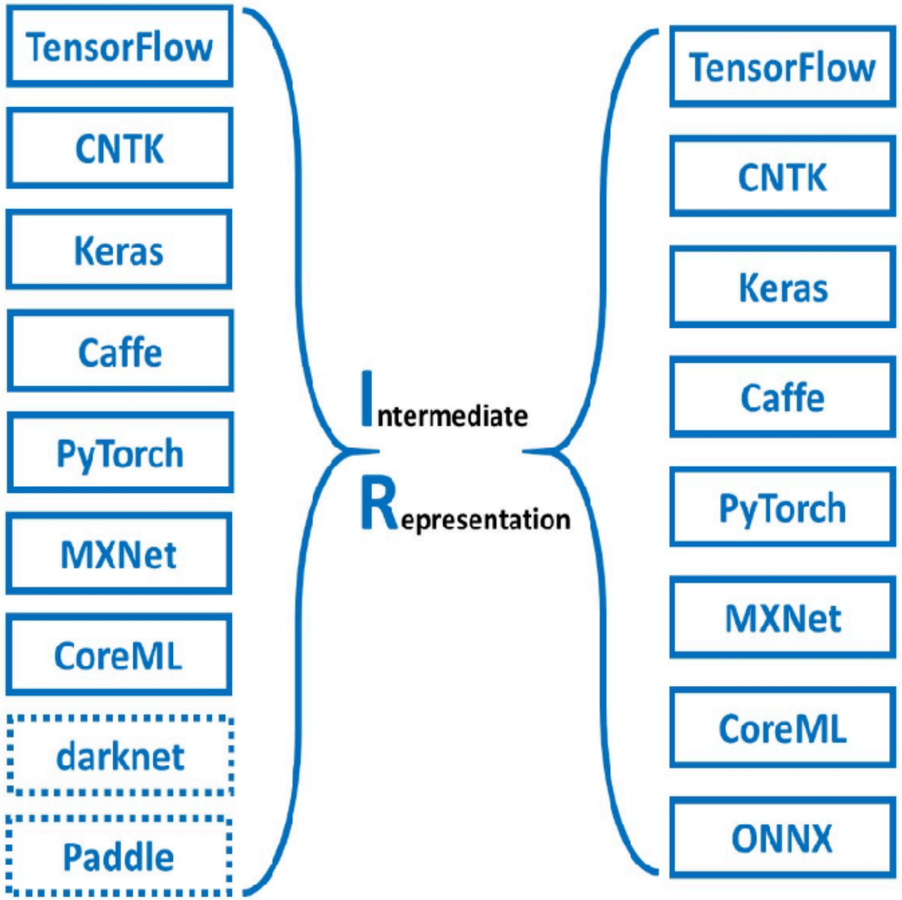


- P20 pro Huawei - 0.97 sec
- Pixel - 2.84 sec





Model convertor



https://github.com/ysh329/deep-learning-model-convertor

Pipeline нашей модели

- TF - torch - Caffe - HiAi
- TF - TF quantize - pytorch - snpe
- TF -- pytorch - snpe
- TF - TF quantize - pytorch - mace
- TF -- pytorch - mace
- TF - tflite

On-device CPU frameworks

- Tensorflow
- NCNN (Tencent)
- CNNdroid *

On-device GPU frameworks

- Snapdragon Neural Processing Engine SDK (SNPE SDK)
- Android Neural Networks API (NNAPI) - Tensorflow lite
- HUAWEI HiAI
- MACE XiaoMi
- MNN

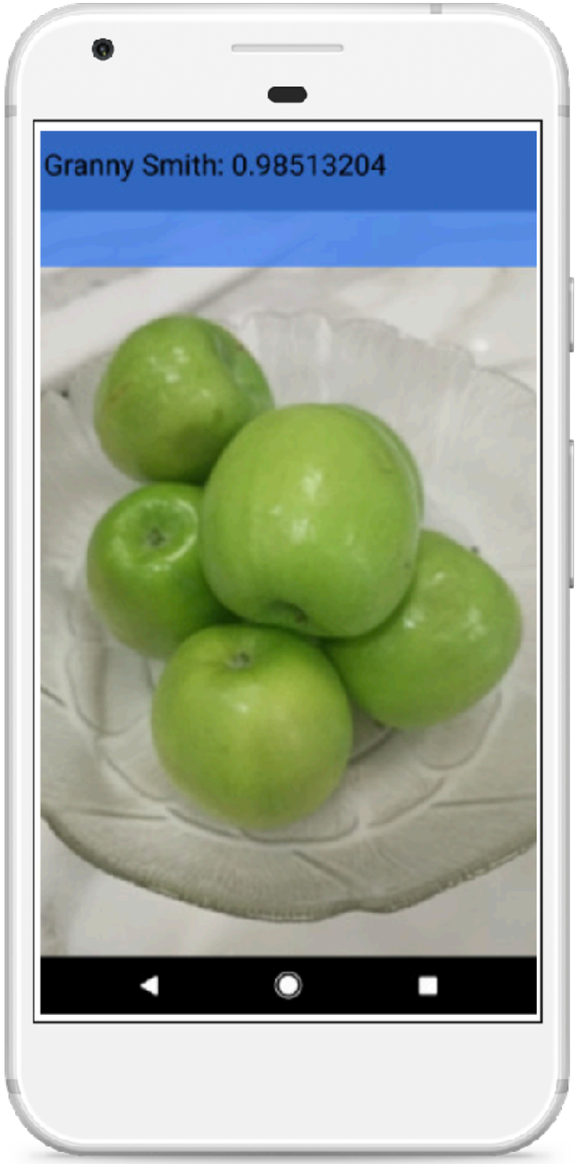
Tensorflow

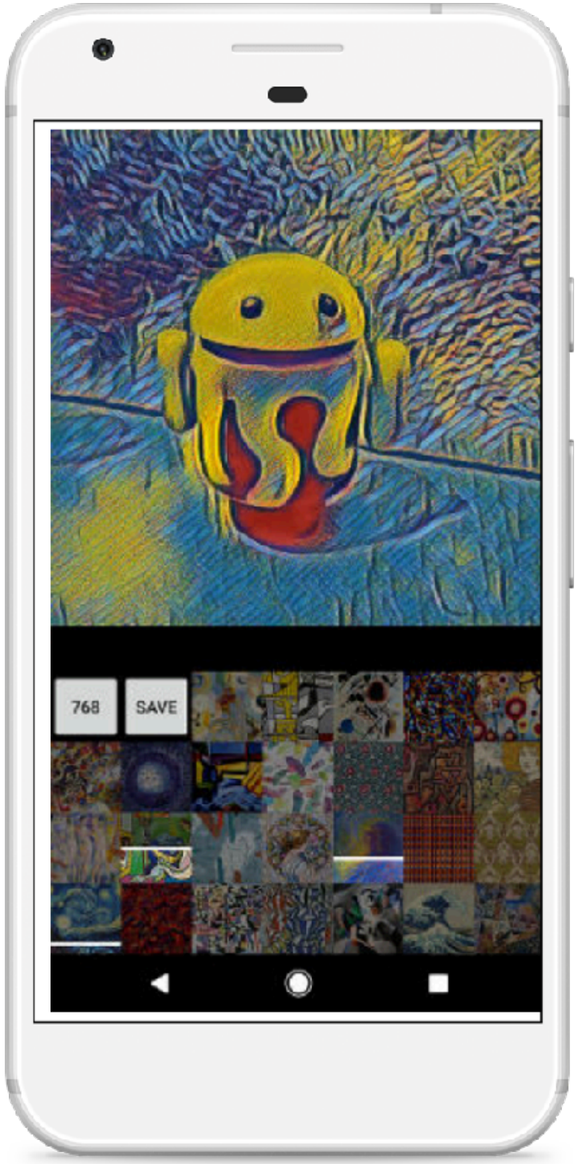
- Запускается всегда и везде
- Хорошая документация
- Доступны почти все слои
- Доступен из Jcenter:

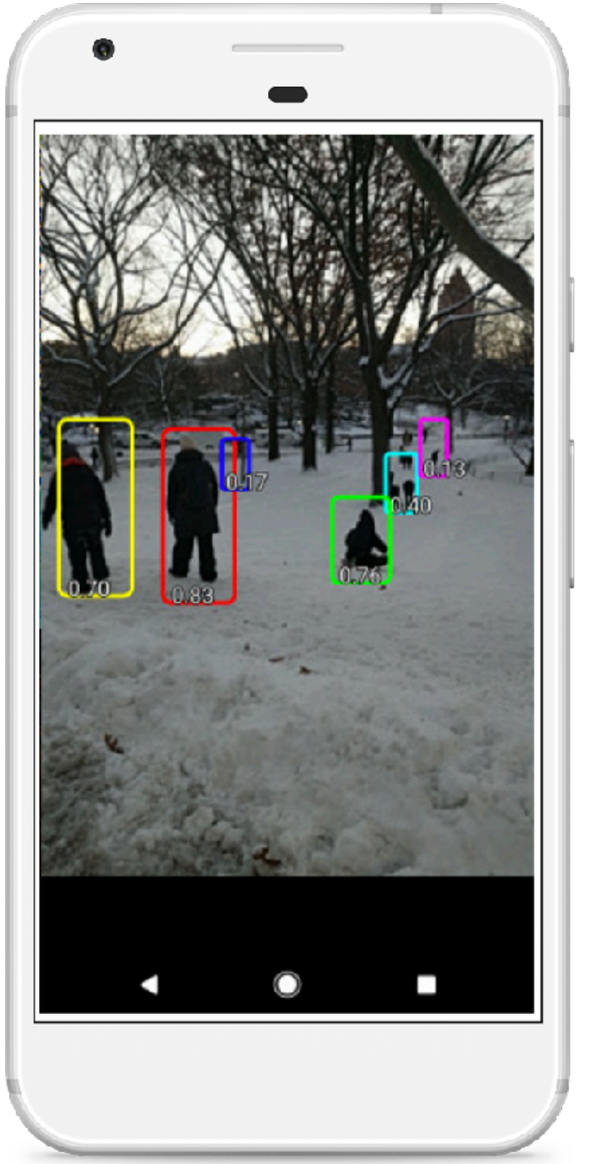
```
allprojects {
    repositories {
        jcenter()
        }
}
 dependencies {
     compile 'org.tensorflow:tensorflow-android:+'
}
```

- Есть сложность если собирать библиотеку руками
- YUV -> RGB однопоточный на CPU
- При запуске на устройствах Android 5.0 (API 21) или выше, требует подключение camera2 API.
- Самое стабильное
- Самое медленное из всех решений

TensorFlow







NCNN (Tencent)

- No third-party library dependencies, does not rely on BLAS / NNPACK or any other computing framework
- Pure C ++ implementation, cross-platform, supports android, ios and so on ARM - NEON assembly level of careful optimization, calculation speed is extremely high
- Sophisticated memory management and data structure design, very low memory footprint
- Supports multi-core parallel computing acceleration, ARM big.LITTLE cpu scheduling optimization
- The overall library size is less than 500K, and can be easily reduced to less than 300K
- Can import caffe/pytorch/mxnet/onnx models
- Benchmarking tool
- Used in QQ, Qzone, WeChat, Pitu and so on.

NCNN benchmark tools

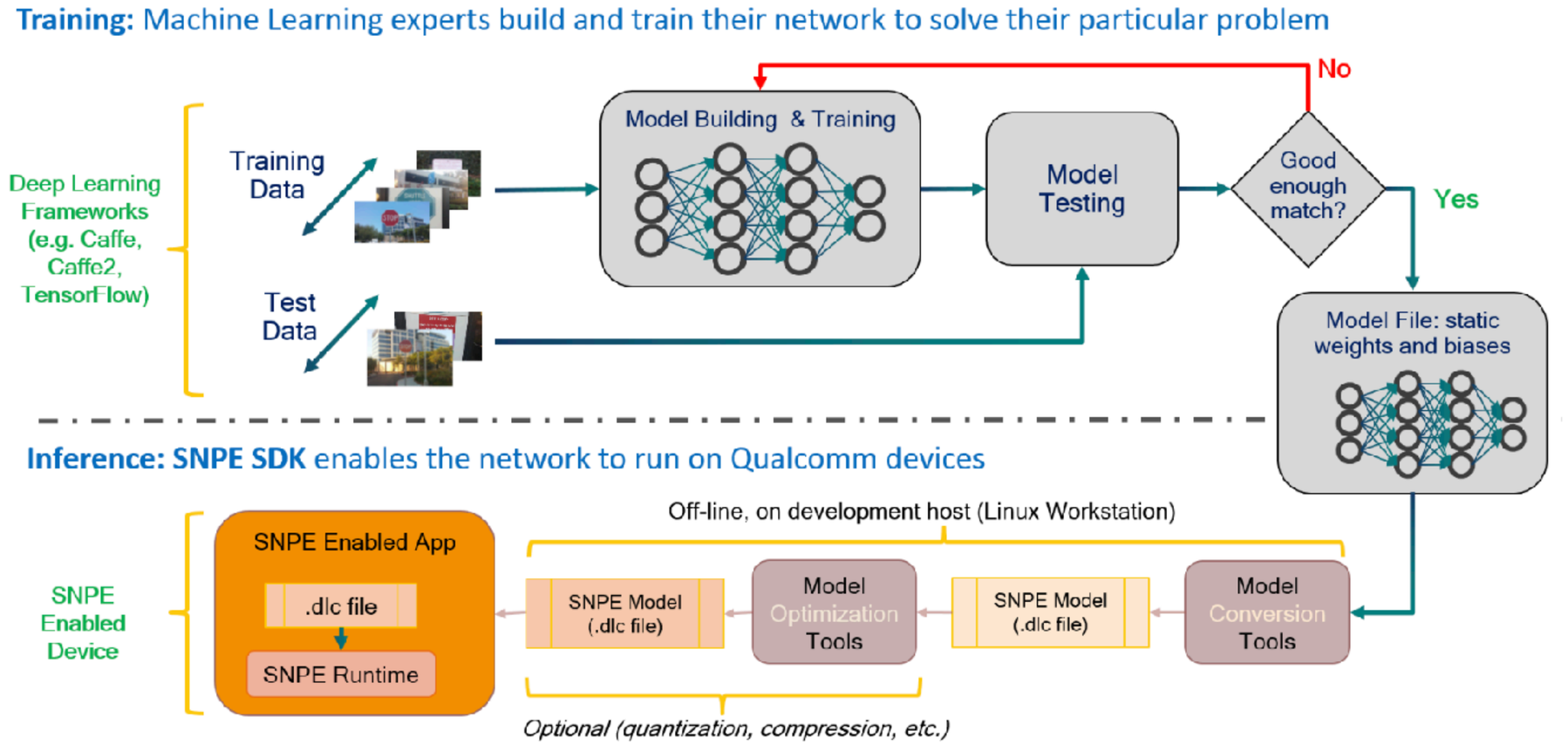
Qualcomm MSM8996 Snapdragon 820 (Kyro 2.15GHz x 2 + Kyro 1.6GHz x 2)

```
root@msm8996:/data/local/tmp/ncnn # ./benchncnn 8 4 0
loop_count = 8
num_threads = 4
powersave = 0
      squeezenet  min =   23.20  max =   24.06  avg =   23.63
       mobilenet  min =   35.89  max =   36.41  avg =   36.09
    mobilenet_v2  min =   27.04  max =   28.62  avg =   27.39
      shufflenet  min =   15.47  max =   16.45  avg =   16.00
       googlenet  min =   85.42  max =   86.15  avg =   85.81
        resnet18  min =   76.82  max =   79.63  avg =   78.50
         alexnet  min =  147.66  max =  156.92  avg =  152.95
           vgg16  min =  493.50  max =  515.03  avg =  507.34
  squeezenet-ssd  min =   56.31  max =   59.35  avg =   57.49
   mobilenet-ssd  min =   68.95  max =   74.24  avg =   71.39
  mobilenet-yolo  min =  142.52  max =  149.72  avg =  148.23
```

SNPE SDK

- Execute an arbitrarily deep neural network
- Execute the network on the SnapdragonTM CPU, the AdrenoTM GPU or the HexagonTM DSP.
- Debug the network execution on x86 Ubuntu Linux
- Convert Caffe, Caffe2, ONNXTM and TensorFlowTM models to a SNPE Deep Learning Container (DLC) file
- Quantize DLC files to 8 bit fixed point for running on the Hexagon DSP
- Debug and analyze the performance of the network with SNPE tools
- Integrate a network into applications and other code via C++ or Java
- Need OpenCL for GPU runtime

Model Workflow



Snapdragon Device Support Matrix

Snapdragon Device	CPU	GPU	DSP
Qualcomm Snapdragon 845	Yes	Yes	Yes (CDSP)
Qualcomm Snapdragon 835	Yes	Yes	Yes (ADSP)
Qualcomm Snapdragon 821	Yes	Yes	Yes (ADSP)
Qualcomm Snapdragon 820	Yes	Yes	Yes (ADSP)
Qualcomm Snapdragon 710	Yes	Yes	Yes (CDSP)
Qualcomm Snapdragon 660	Yes	Yes	Yes (CDSP)
Qualcomm Snapdragon 652	Yes	Yes	No
Qualcomm Snapdragon 630	Yes	Yes	No
Qualcomm Snapdragon 636	Yes	Yes	No
Qualcomm Snapdragon 625	Yes	Yes	No
Qualcomm Snapdragon 450	Yes	Yes	No

mi mix2s	GPU_FLOAT16	GPU	CPU	DSP
test_16_06_2018.dlc	33	45	470	59
test_16_06_2018-quantized.dlc	33	46	466	58
test_fast_16_06_2018-4channel.dlc (no	38	43	470	77
test_8.dlc (bad output)	26	32	677	43
test_8-nodepthwise.dlc (bad output)	31	39	305	41
test_t8-3x3deconv.dlc (bad output)	39	45	1102	49
bench_ResBlock-quantized.dlc	41	32	542	30
bench_ResBlock.dlc	40	44	511	29
bench_ResBlock-long-quantized.dlc	16	19	617	46
bench_ResBlock-long.dlc	19	16	1021	47
test_24_06_2018-quantized.dlc. - 1,9 m	62	55	-	56
test_fast_24_06_2018.dlc. - 7,4 mb	33	42	-	57
test_fast_25_06_2018.dlc. - 7,4 mb	45	46	-	44
test_02_07.dlc. - 256	87	120	-	192
test_02_07.dlc. - 128	25	31	-	55
test_03_07.dlc	25	30		61
test_v2_03_07_v2	26	33		61
test_2_03_07_v3	26	29		59
test_v3	crash	crash	-	247610
test-quantized.dlc 3,3 mb	132	160	-	227
test.dlc. 13 mb	130	161	-	215

Android Neural Networks API (NNAPI)

- Performance: TF Lite is fast with no noticeable accuracy loss

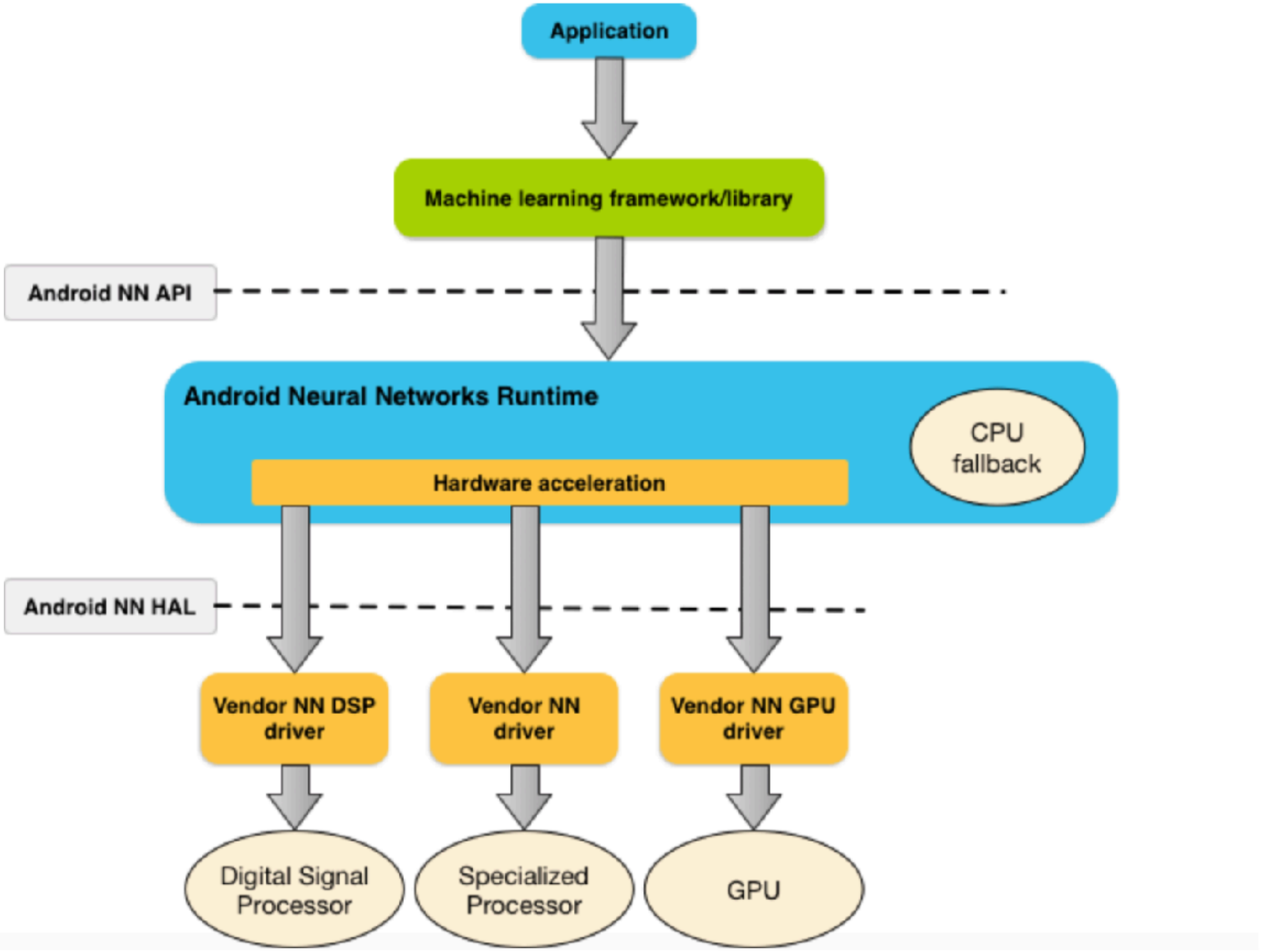
- Portability: Android, iOS, and more specialized IoT devices.

- Low latency: Optimized float- and fixed-point CPU kernels, op-fusing, and more.

- Acceleration: Integration with GPU and internal/external accelerators.

- Small model size: Controlled dependencies, quantization, and op registration.

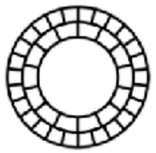
- Tooling: Conversion, compression, benchmarking, power-consumption, and more.


Application
Machine learning framework/library
Android NN API
Android Neural Networks Runtime
CPU fallback
Hardware acceleration
Android NN HAL
Vendor NN DSP driver
Vendor NN driver
Vendor NN GPU driver
Digital Signal Processor
Specialized Processor
GPU

Companies using TensorFlow Lite








HiAI Mobile Computing Platform

- Works only on Kirin 970, 980

- Platform with a dedicated Neural-network Processing Unit (NPU)

HiAI problems

		cpu convert		
model32	512x512		pass	3 блока по 32->32(3x3)
model16	512x512	714	pass	6 блоков по 16->16(3x3)
model32v2	512x512	260	bad output	3 блока по 32->32(3x3) with upsample
model16v2	512x512		no	6 блоков по 16->16(3x3) with upsample
upsample	512x512		bad output	only upsample layer
model32with1x1	512x512		pass	2 блока по 32->32(3x3) + 1 32->32(1x1)
model32with5x5	512x512	190-290	pass	2 блока по 32->32(3x3) + 1 32->32(5x5)
model32with	512x512		pass	1 блока по 32->32(3x3) + 1 32->32(1x1) + 1 32->32(5x5)

- Performance
  - Runtime is optimized with NEON, OpenCL and Hexagon, and Winograd algorithm is introduced to speed up convolution operations. The initialization is also optimized to be faster.
- Power consumption
  - Chip dependent power options like big.LITTLE scheduling, Adreno GPU hints are included as advanced APIs.
- Responsiveness
  - UI responsiveness guarantee is sometimes obligatory when running a model. Mechanism like automatically breaking OpenCL kernel into small units is introduced to allow better preemption for the UI rendering task.
- Memory usage and library footprint
  - Graph level memory allocation optimization and buffer reuse are supported. The core library tries to keep minimum external dependencies to keep the library footprint small.
- Model protection
  - Model protection has been the highest priority since the beginning of the design. Various techniques are introduced like converting models to C++ code and literal obfuscations.
- Platform coverage
  - Good coverage of recent Qualcomm, MediaTek, Pinecone and other ARM based chips. CPU runtime is also compatible with most POSIX systems and architectures with limited performance.

- CPU
  Almost all of mobile SoCs use ARM-based CPU architecture, so your model could run on different SoCs in theory.

- GPU
  Although most GPUs use OpenCL standard, but there are some SoCs not fully complying with the standard, or the GPU is too low-level to use. So you should have some fallback strategies when the GPU run failed.

- Tuning for specific SoC's GPU

- DSP
  MACE only support Qualcomm DSP.

Tools for bench

Node Type	Start	First	Avg(ms)	%	cdf%	Stride	Pad	Filter Shape	Output Shape	Dilation	name
DepthwiseConv2d	89.750	0.330	0.329	2.941	2.941	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise_35
DepthwiseConv2d	2.570	0.327	0.328	2.938	5.879	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise
DepthwiseConv2d	94.666	0.326	0.326	2.918	8.797	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise_37
DepthwiseConv2d	92.405	0.326	0.326	2.915	11.712	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise_36
DepthwiseConv2d	87.365	0.323	0.325	2.905	14.617	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise_34
DepthwiseConv2d	4.922	0.320	0.319	2.850	17.468	[1,1]	VALID	[1,8,3,3]	[1,256,256,8]	[1,1]	depthwise_1
Conv2D	0.692	0.300	0.301	2.691	20.159	[1,1]	VALID	[8,3,3,3]	[1,256,256,8]	[1,1]	convolution
Conv2D	97.223	0.298	0.299	2.674	22.832	[1,1]	VALID	[1,8,3,3]	[1,256,256,1]	[1,1]	convolution_43
Deconv2D	85.570	0.228	0.230	2.058	24.890				[1,256,256,8]		conv2d_transpose_3
Conv2D	7.295	0.173	0.174	1.557	26.446	[2,2]	VALID	[8,8,3,3]	[1,128,128,8]	[1,1]	convolution_3

Stat by node type

Node Type	Count	Avg(ms)	%	cdf%	Called times
DepthwiseConv2d	38	3.078	27.820	27.820	38
Eltwise	108	2.760	24.946	52.766	108
Conv2D	44	2.342	21.168	73.933	44
Pad	44	1.159	10.475	84.409	44
Activation	41	0.992	8.966	93.375	41
Deconv2D	4	0.623	5.631	99.006	4
BufferToImage	1	0.064	0.578	99.584	1
ImageToBuffer	1	0.046	0.416	100.000	1

Summary of Ops' Stat

round	first(ms)	curr(ms)	min(ms)	max(ms)	avg(ms)	std
100	11.175	11.171	11.113	11.321	11.176	41.089

MNN:mobile neural network

- Supports `Tensorflow`, `Caffe`, `ONNX`, and supports common neural networks such as `CNN`, `RNN`, `GAN`.
- Supports 86 `Tensorflow` ops, 34 `Caffe` ops; MNN ops: 71 for CPU, 55 for Metal, 29 for OpenCL, and 31 for Vulkan.
- Supports iOS 8.0+, Android 4.3+ and embedded devices with POSIX interface.
- Supports hybrid computing on multiple devices. Currently supports CPU and GPU. GPU op plugin can be loaded dynamically to replace default (CPU) op implementation.
- Implements core computing with lots of optimized assembly code to make full use of the ARM CPU.
- For iOS, GPU acceleration (Metal) can be turned on, which is faster than Apple's native CoreML.
- For Android, `OpenCL`, `Vulkan`, and `OpenGL` are available and deep tuned for mainstream GPUs (`Adreno` and `Mali`).
- Convolution and transposition convolution algorithms are efficient and stable. The Winograd convolution algorithm is widely used to better symmetric convolutions such as 3x3 -> 7x7.
- Additional optimizations for the new architecture ARM v8.2 with half-precision calculation support.

Pre- and post- processing

- Java

- C++

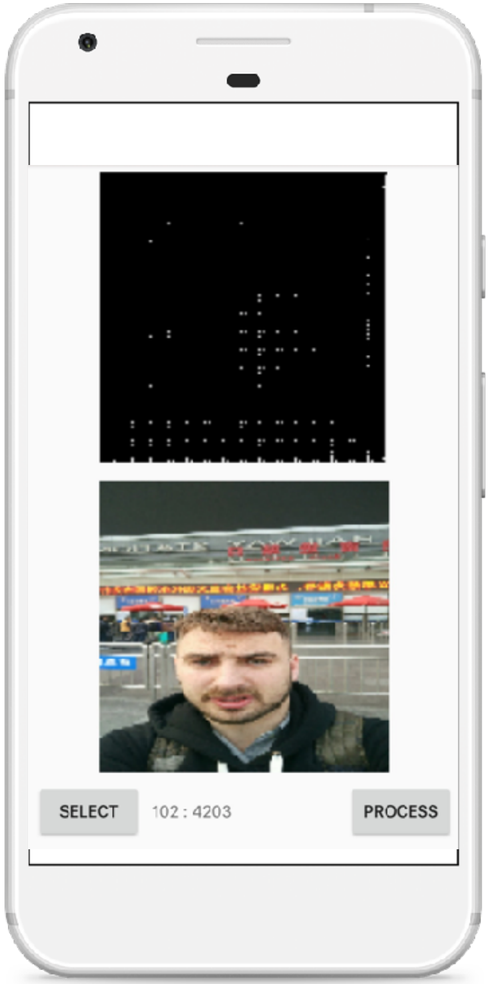
- Intel TBB или OpenMPI

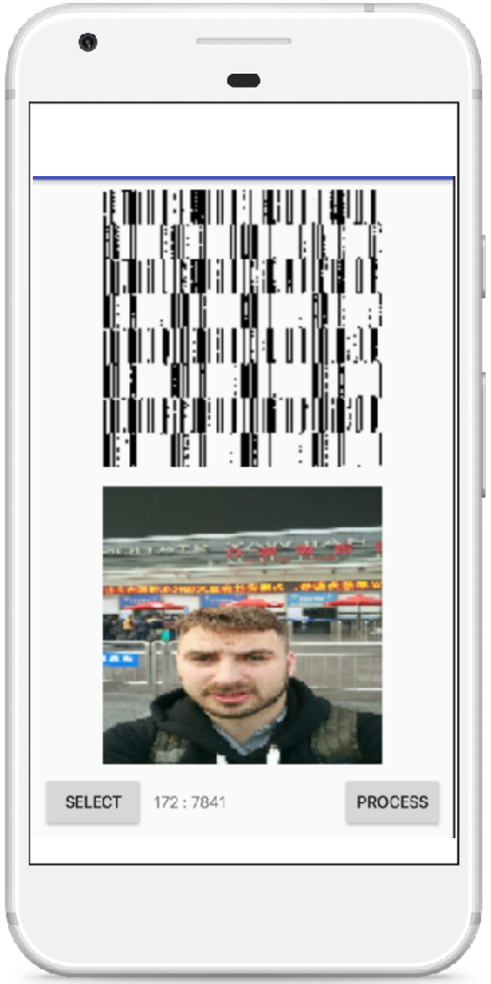
- Filterscript

- ?GPGPU

- Добавить операции как слои

Pre- and post- processing fail

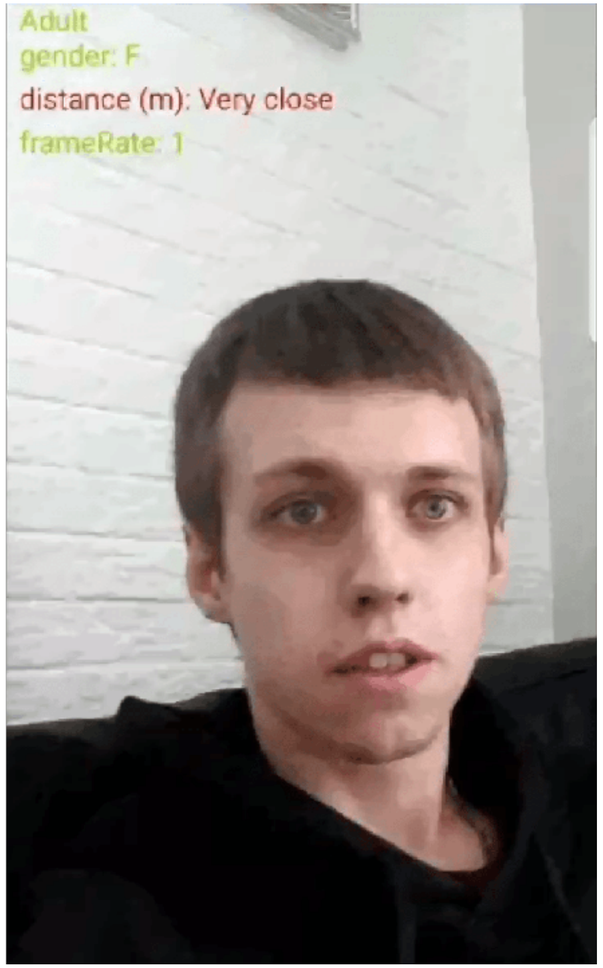




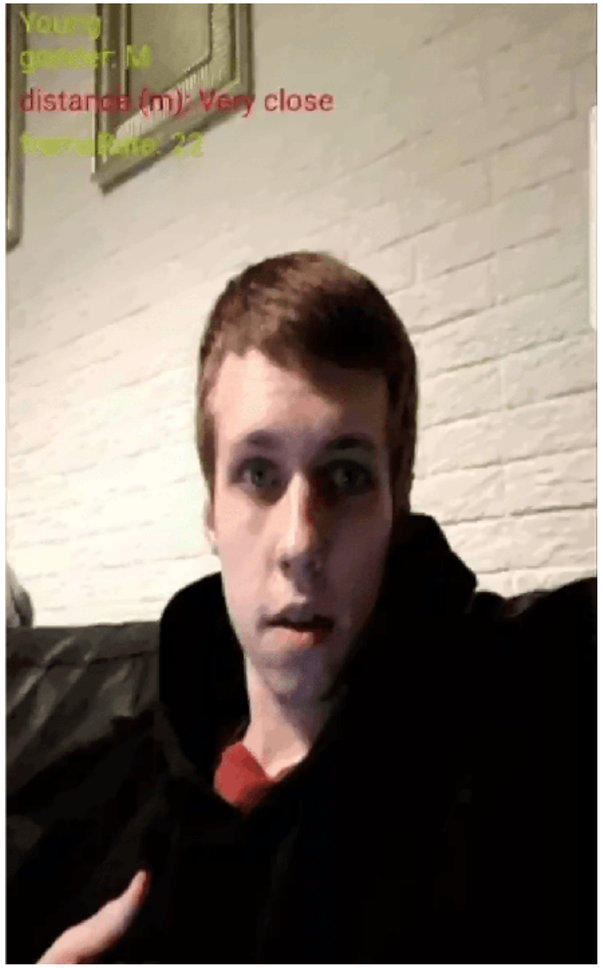
Pre- and post-processing. Benches

Tool	FPS	Script time
java	2	500-600 ms
c++	15-20	60-70 ms
renderscript	35-40	25-30 ms
filterscript	60-70	12-15 ms

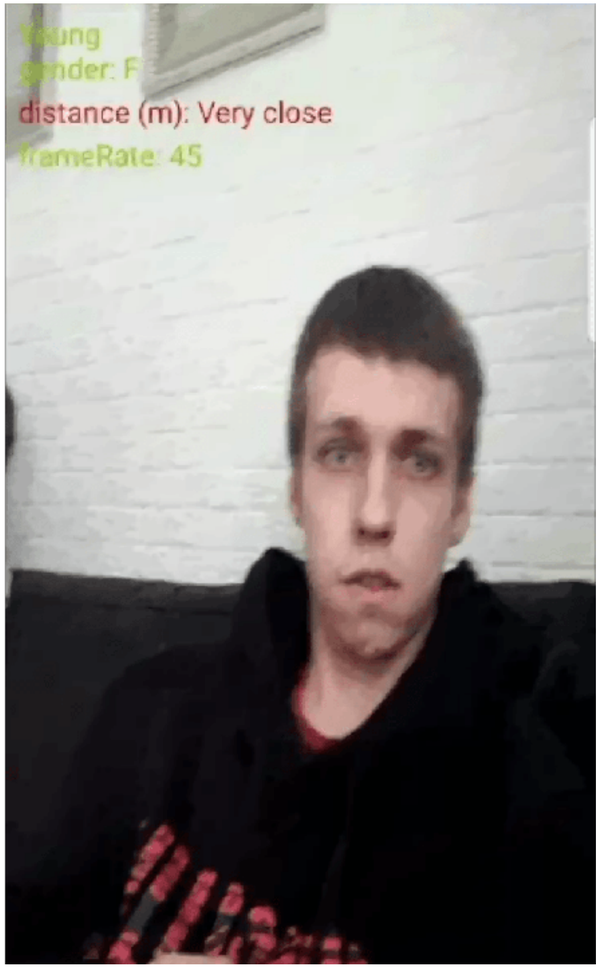
Java


Adult
gender: F
distance (m): Very close
frameRate: 1

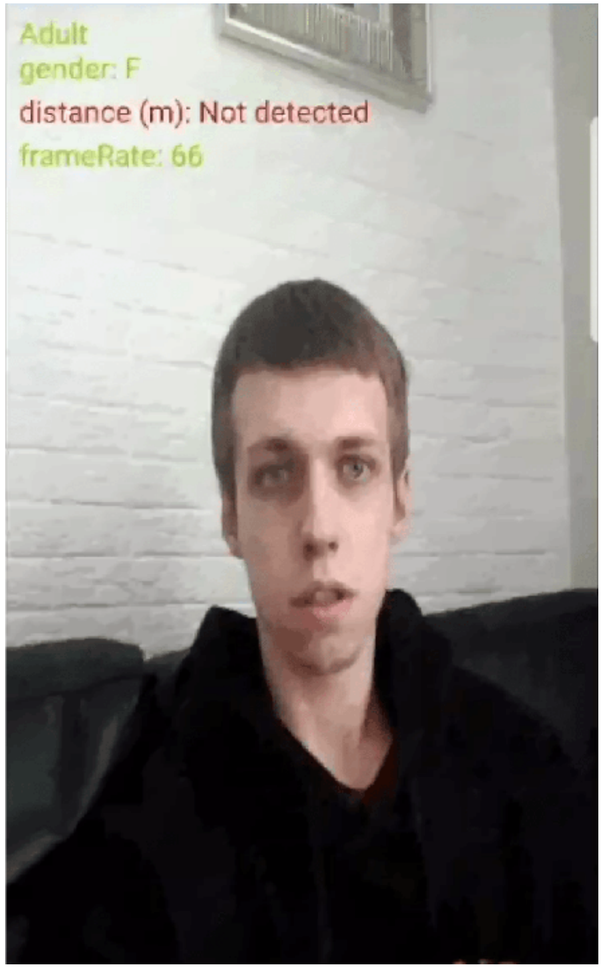
C++

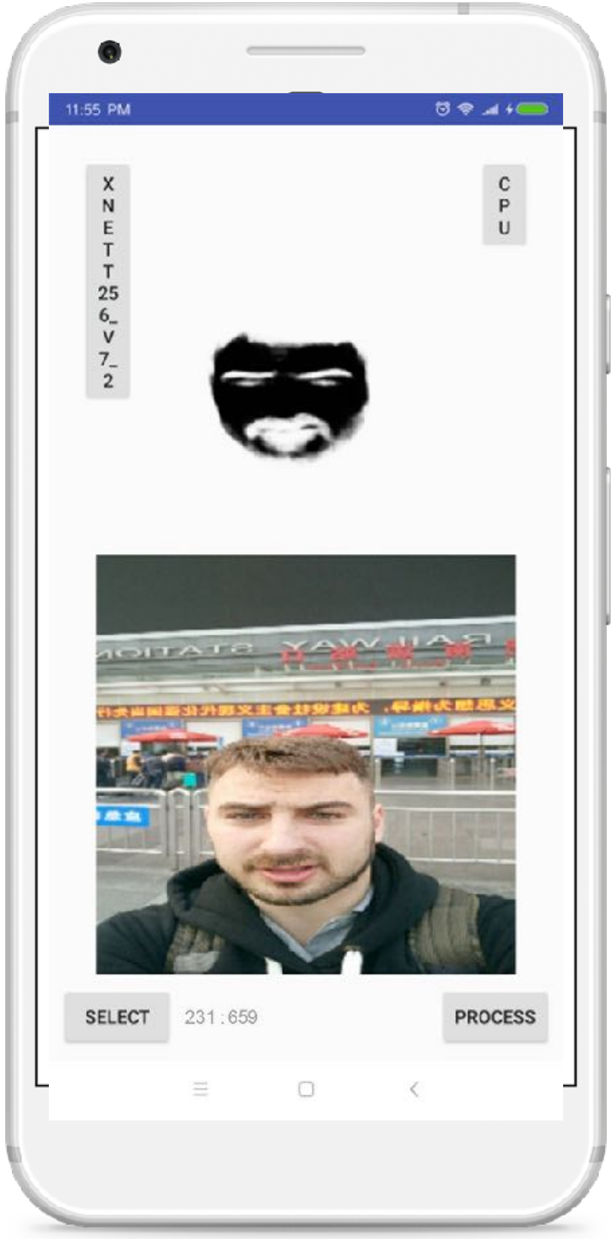

distance (m): Very close

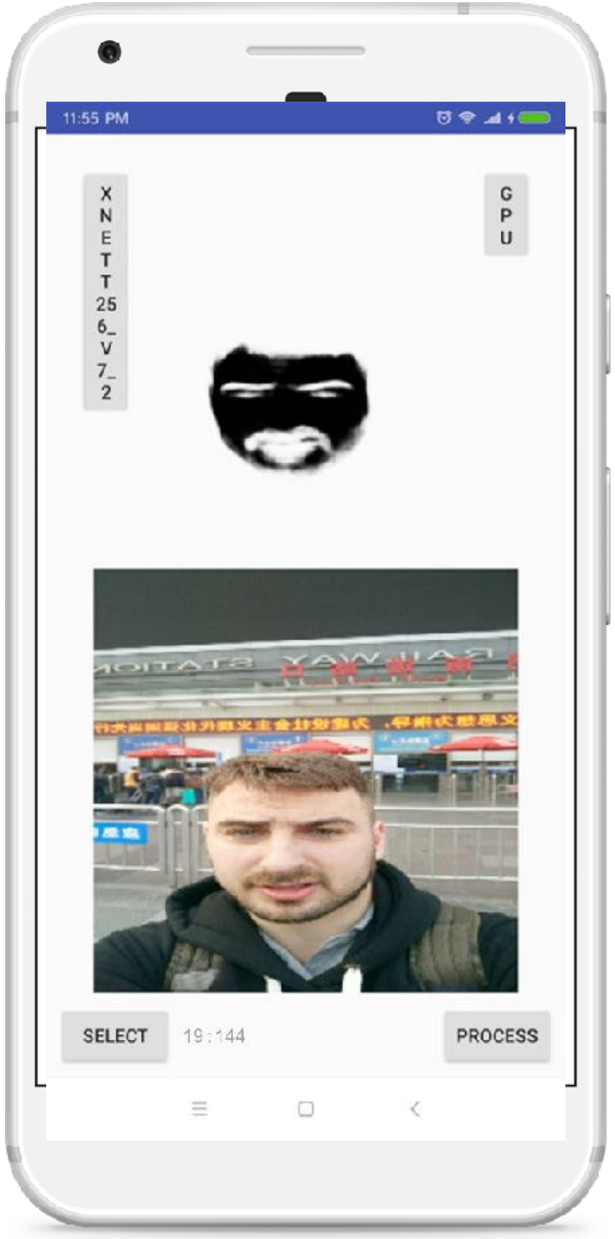
rs


distance (m): Very close
frameRate: 45

fs

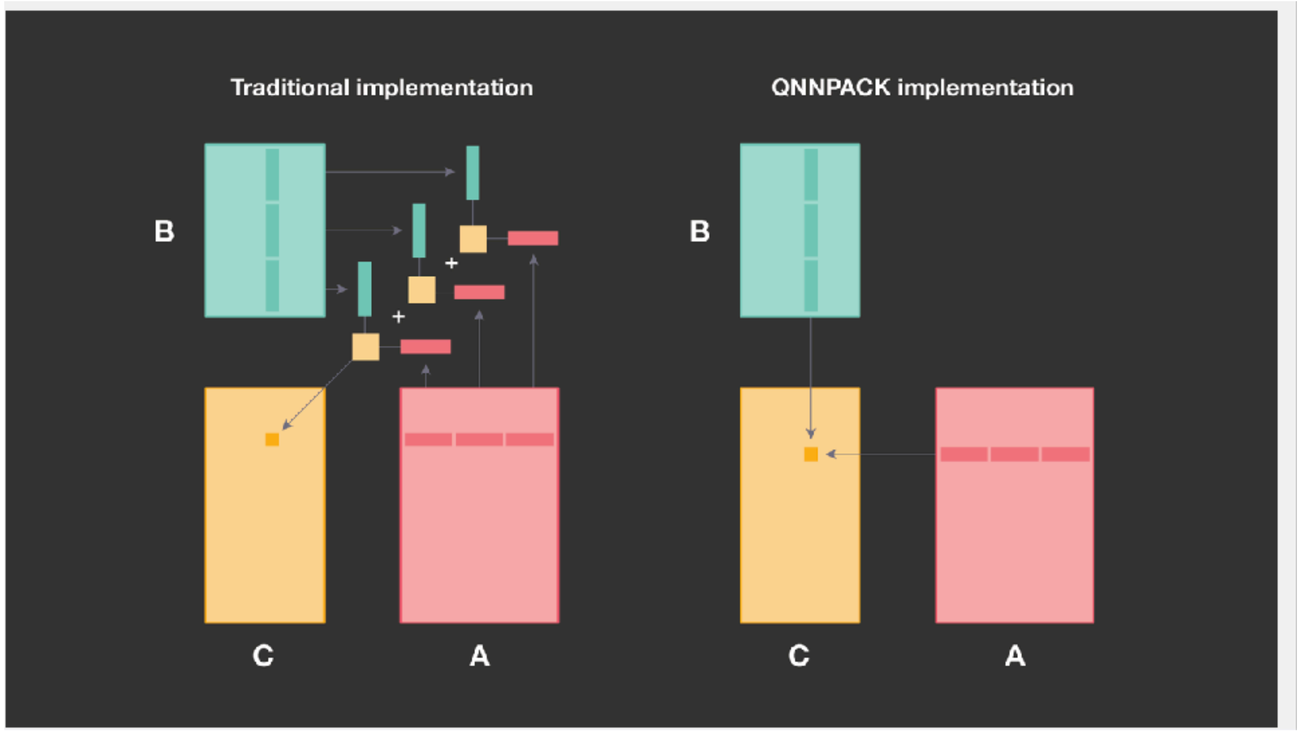

Adult
gender: F
distance (m): Not detected
frameRate: 66


11:55 PM
XNETT256_V7_2
CPU
SELECT
231 : 659
PROCESS


11:55 PM
XNETT256_V7_2
GPU
SELECT
19 : 144
PROCESS

Trends

- **QNNPACK:** Open source library for optimized mobile deep learning



- **DSP**

Итог

- Идеального рецепта нет
- Охватить весь рынок сразу – невозможно
- Много тестов
- Большой вес моделей
- Разный математический выход у библиотек
- Поддержка разных слоев – невозможно сделать модель для всех
- За последний год ситуация изменилась кардинально

Спасибо за подготовку данных

Efanov Evgeny

**Senior Flutter, Android developer
in Exploud.
Studying android GPU processing**

Adeshchenko Kirill

**Middle Flutter developer
in Exploud.
Loves to try something new.**