2020 embedded VISION summit®

Deep Learning On Mobile

A Practitioner's Guide

Siddha Ganju

September 2020

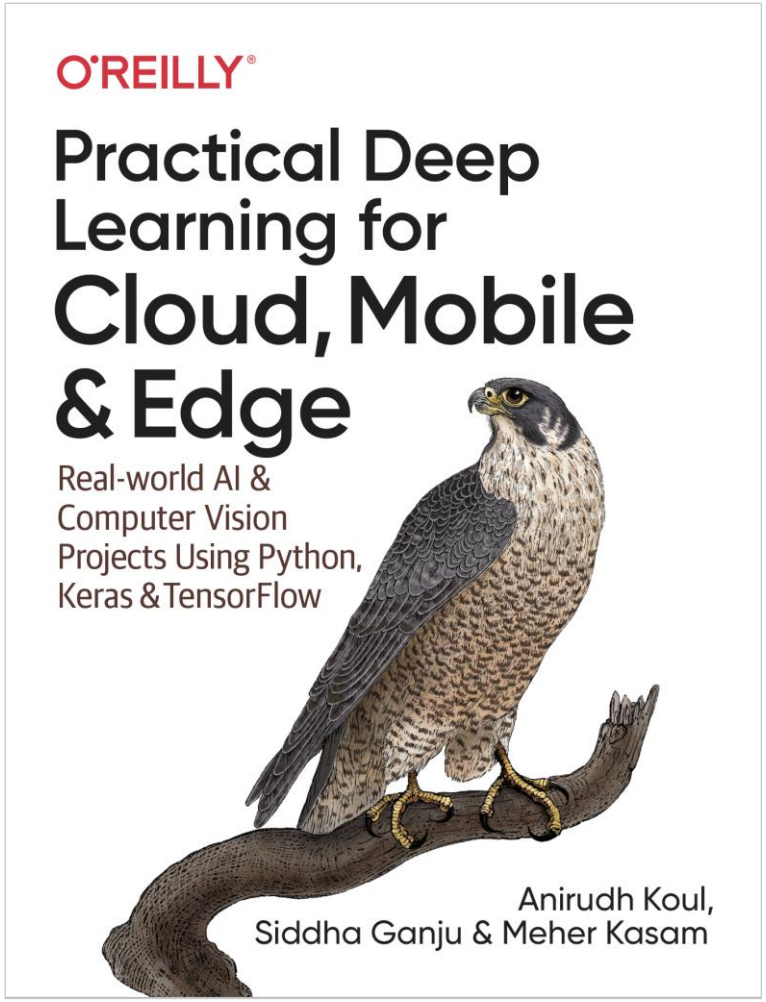
2020 embedded VISION summit®

Deep Learning On Mobile

A Practitioner's Guide

Siddha Ganju

September 2020



@SiddhaGanju

@MeherKasam

@AnirudhKoul

Why Deep Learning on Mobile?

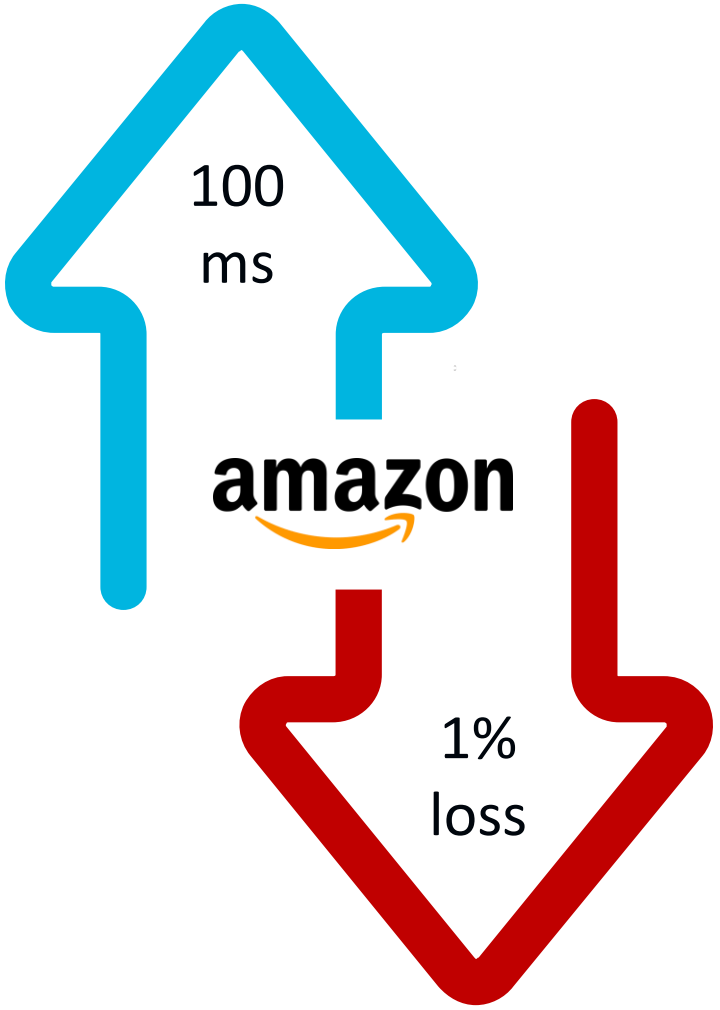
Privacy

Reliability

Cost

Latency

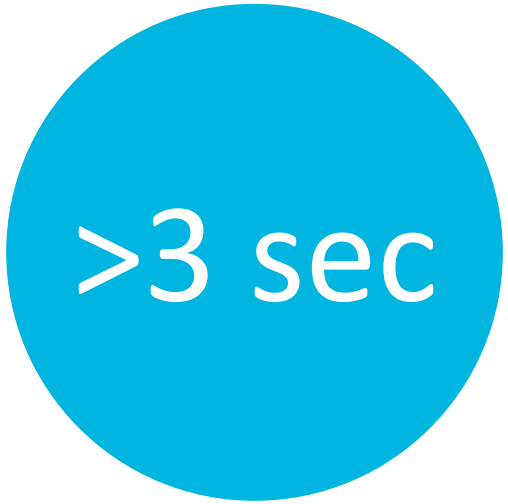
Latency Is Expensive!



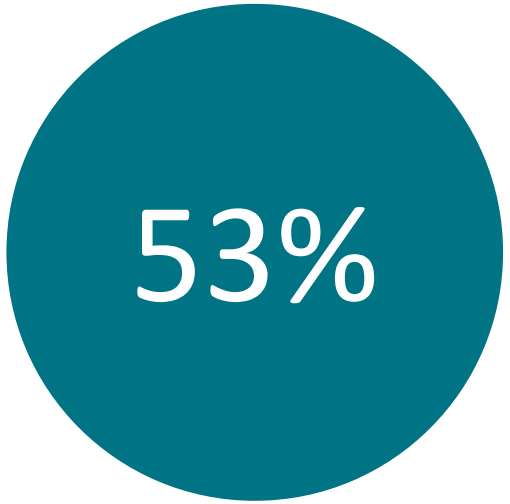
[Amazon 2008]

Latency Is Expensive!

Mobile Site Visits



Load time



Bounce

[Google Research, Webpagetest.org]

Power of 10

Seamless

Uninterrupted flow of thought

Limit of attention

[Miller 1968; Card et al. 1991; Nielsen 1993]

High Quality Dataset + Hardware + Efficient Mobile Inference Engine + Efficient Model = DL App

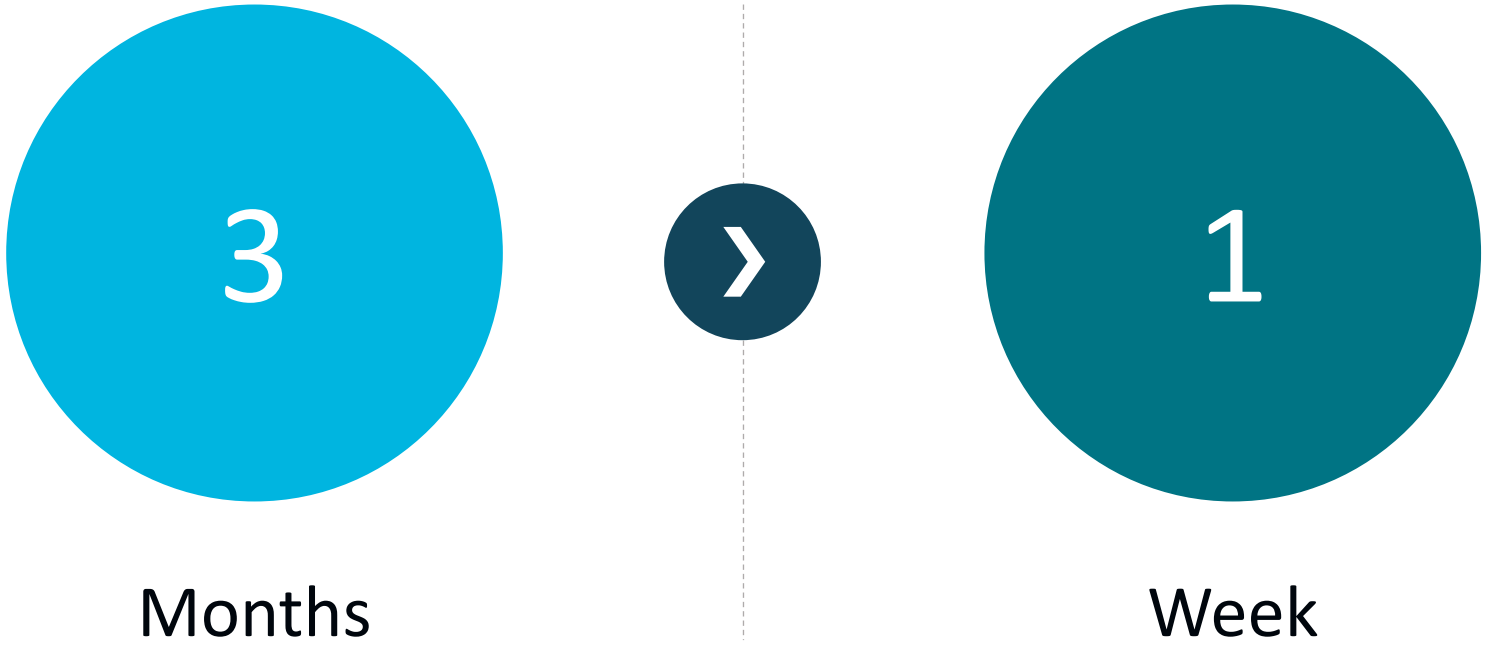
How do I train my model?

Learn to
Play Melodica

3 Months

Already Play Piano?

FINE TUNE
Your skills

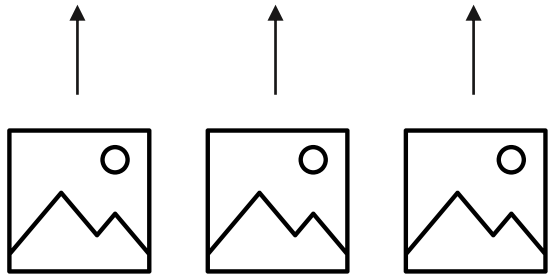


Fine tuning

Assemble a dataset → Find a pre-trained model → Fine-tune a pre-trained model → Run using existing frameworks

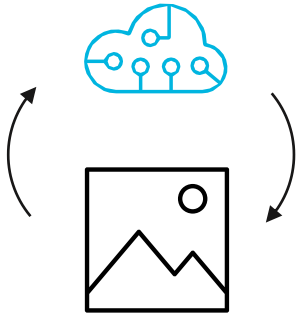
> Don't Be A Hero
>
> — Andrej Karpathy

CustomVision.ai



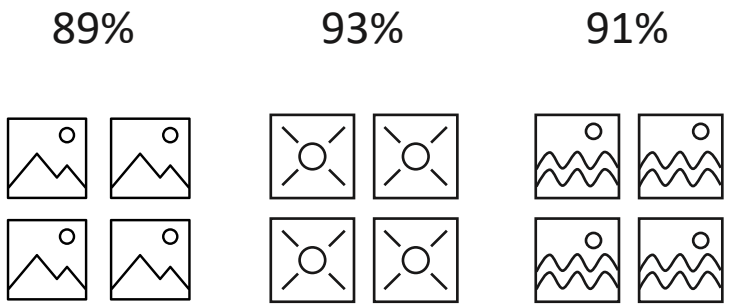
Upload Images

Bring your own labeled images, or use custom vision to quickly add tags to any unlabeled images



Train

Use your labeled images to teach custom vision the concepts you care about



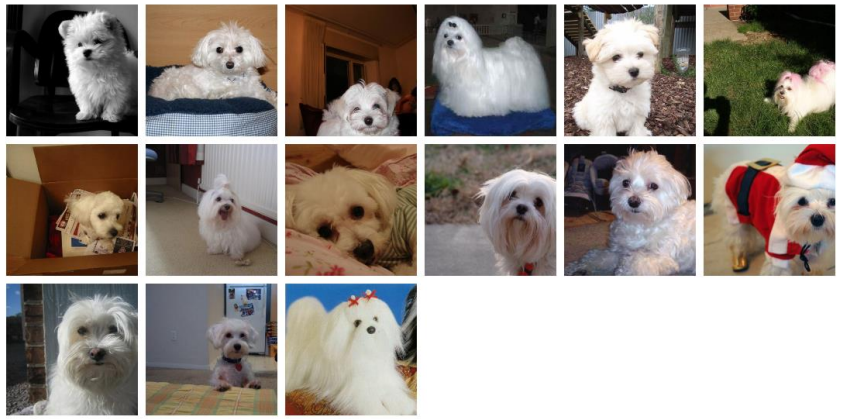
Evaluate

Use simple REST API calls to quickly tag images with your new custom computer vision model

Use Fatkun Browser Extension to download images from Search Engine,
or use Bing Image Search API to programmatically download photos with proper rights

Image upload

Add Tags — Uploading — Summary



33 images will be added...

Add some tags to this batch of images...

My Tags

maltese

Upload 33 files

New project

Name

Dog Breed Classifier

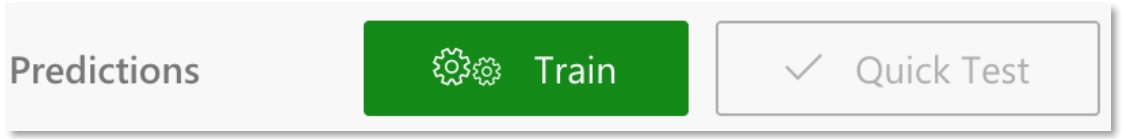
Description

Enter project description

Domains ⓘ

- (•) General
- ( ) Food
- ( ) Landmarks
- ( ) Retail
- ( ) Adult
- ( ) General (compact) ⓘ
- ( ) Landmarks (compact) ⓘ
- ( ) Retail (compact) ⓘ

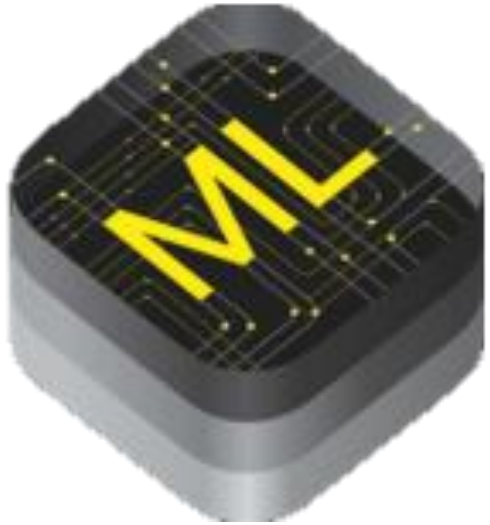
Cancel | Create project

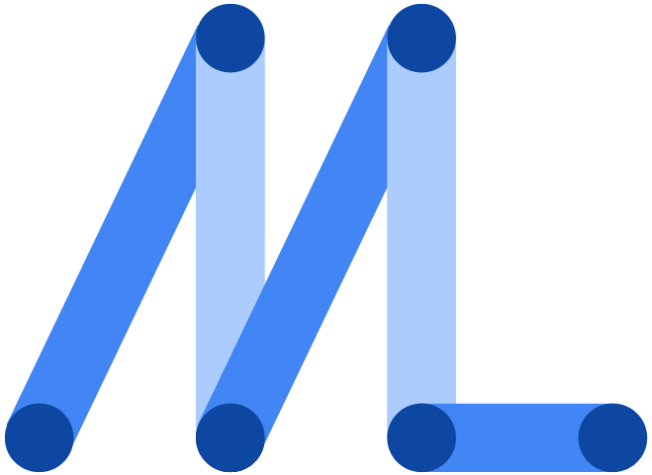


Performance Per Tag

Tag	Precision	Recall
afghan_hound	87.5%	92.0%
airedale	96.0%	92.5%
basenji	97.4%	93.0%
bernese_mountain_dog	91.3%	91.0%
entlebucher	97.2%	87.5%
great_pyrenees	87.7%	85.0%
irish_wolfhound	87.8%	85.0%
leonberg	98.9%	87.0%
maltese_dogs	96.4%	91.5%
pomeranian	97.4%	91.5%

How do I run my models?





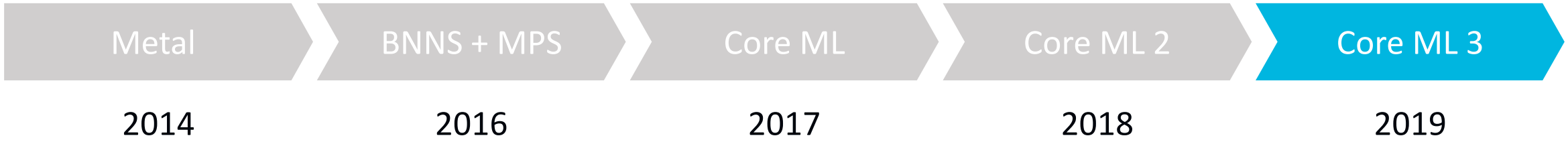
Core ML | **TF Lite** | **ML Kit**

Apple Ecosystem

Metal	BNNS + MPS	Core ML	Core ML 2	Core ML 3
2014	2016	2017	2018	2019

- Tiny models (~ KB)!
- 1-bit model quantization support
- Batch API for improved performance
- Conversion support for MXNet, ONNX
- tf-coreml

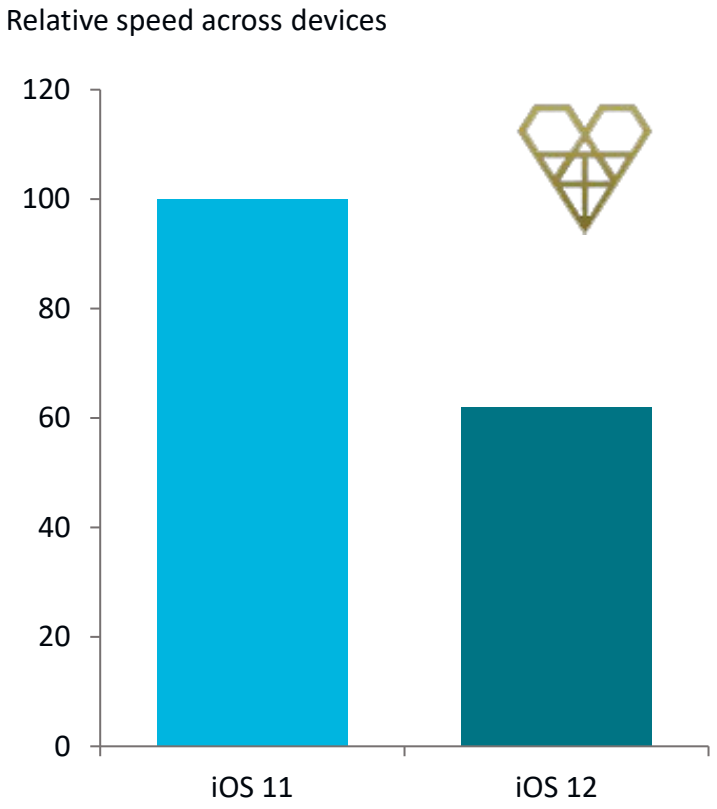
Apple Ecosystem



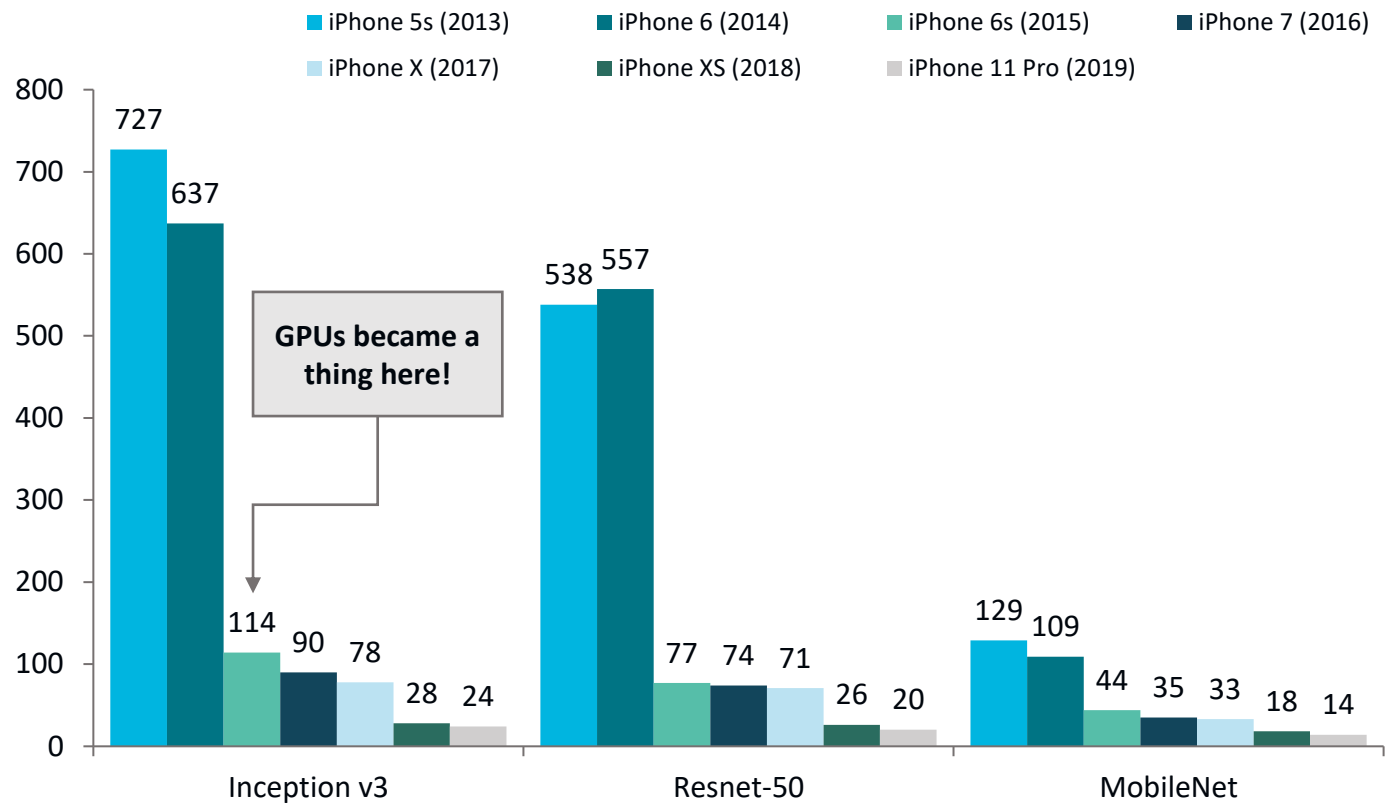
- On-device training
- Personalization
- Create ML UI

Core ML Benchmark

Core ML Runtime Speed by OS



Execution Time (MS) ON Apple Devices



https://heartbeat.fritz.ai/ios-12-core-ml-benchmarks-b7a79811aac1

TensorFlow Ecosystem

TensorFlow → TensorFlow Mobile → TensorFlow Lite

2015 2016 2018

Smaller

Faster

Minimal dependencies

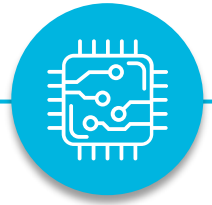
Allows running custom operators

TensorFlow Lite is small

TensorFlow
Mobile

Core Interpreter +
Supported Operations

TensorFlow Lite is Fast



Takes advantage of on-device hardware acceleration

FlatBuffers

- Reduces code footprint, memory usage
- Reduces CPU cycles on serialization and deserialization
- Improves startup time

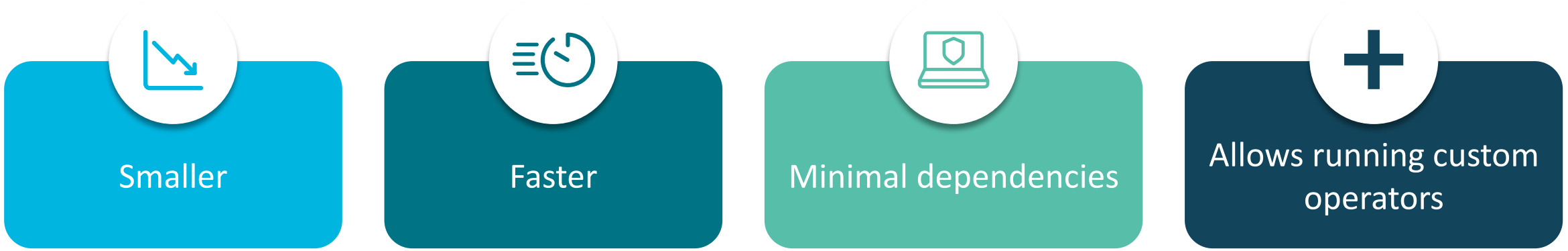
Pre-fused activations

- Combining batch normalization layer with previous convolution

Static memory and static execution plan

- Decreases load time

TensorFlow Ecosystem



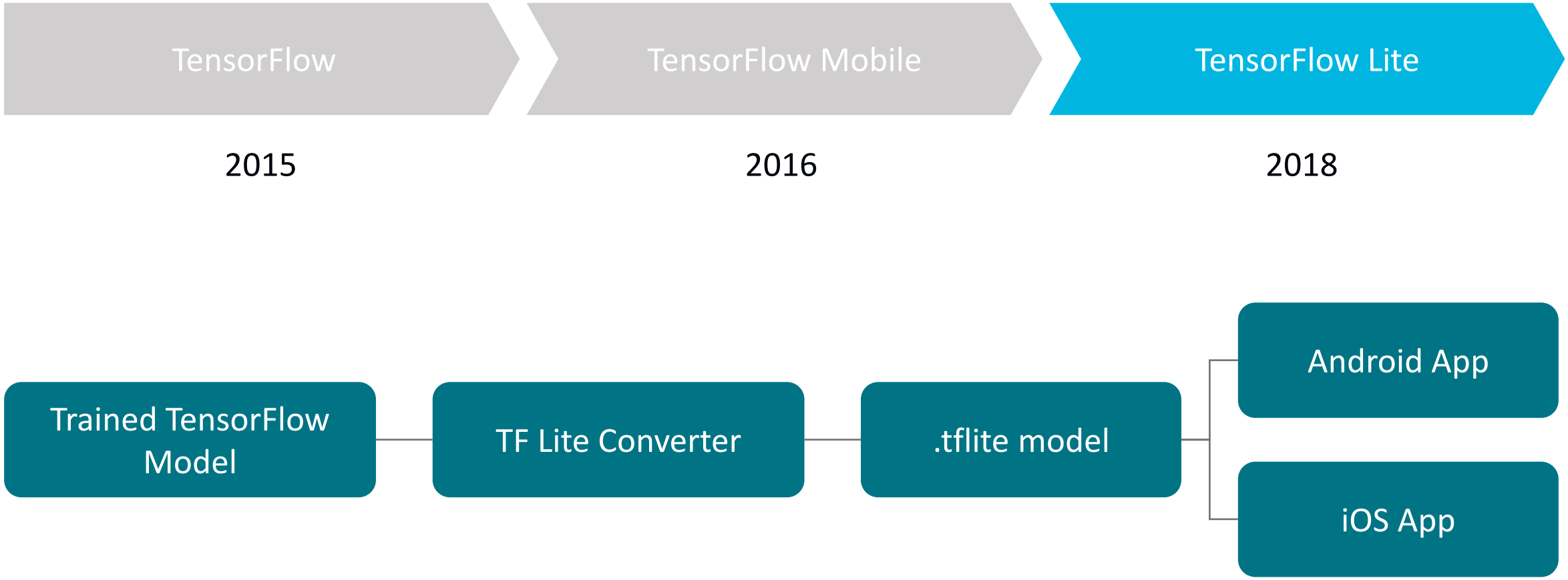
TensorFlow Ecosystem

TensorFlow → TensorFlow Mobile → TensorFlow Lite

2015 2016 2018

```
$ tflite_convert  --keras_model_file = keras_model.h5 --output_file=foo.tflite
```

TensorFlow Ecosystem



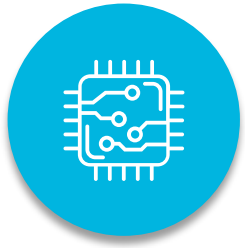
ML Kit

Easy to use

Abstraction over TensorFlow Lite

Built-in APIs for image labeling, OCR, face detection, barcode scanning, landmark detection, smart reply

Model management with Firebase



A/B testing

```
var vision = Vision.vision()
let faceDetector = vision.faceDetector(options: options)
let image = VisionImage(image: uiImage)
faceDetector.process(visionImage) { // callback }
```

How do I keep my IP safe?

Fritz

Full fledged mobile lifecycle support

Deployment, instrumentation, etc. from Python

- Image Labeling
- Image Segmentation
- Object Detection
- Style Transfer
- Pose Estimation
- Analytics + Monitoring
- Model Management
- Model Protection

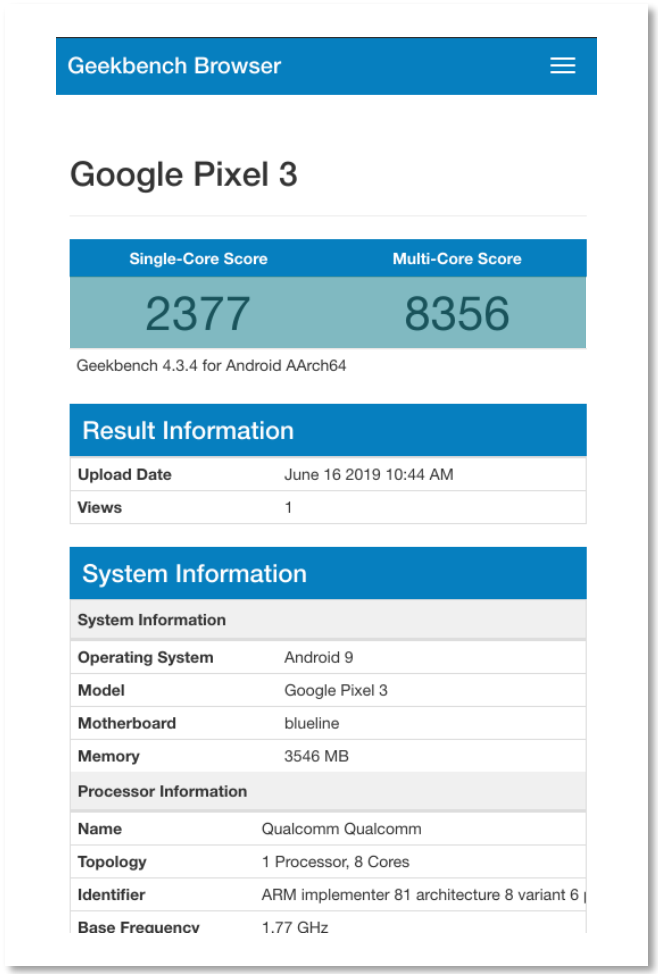
Does my model make me look fat?

Apple does not allow apps over 200 MB to be downloaded over cellular network

Download on demand, and interpret on device instead

What effect does hardware have on performance?

Big things come in small packages



Geekbench Browser

Google Pixel 3

Single-Core Score	Multi-Core Score
2377	8356

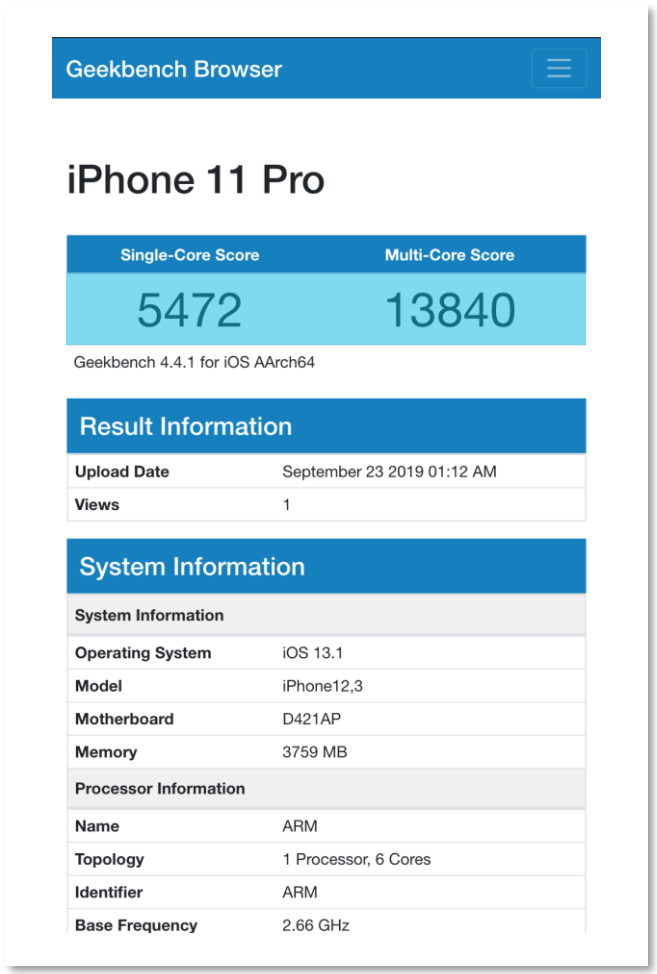
Geekbench 4.3.4 for Android AArch64

Result Information

Upload Date	June 16 2019 10:44 AM
Views	1

System Information

System Information	
Operating System	Android 9
Model	Google Pixel 3
Motherboard	blueline
Memory	3546 MB
Processor Information	
Name	Qualcomm Qualcomm
Topology	1 Processor, 8 Cores
Identifier	ARM implementer 81 architecture 8 variant 6
Base Frequency	1.77 GHz



Geekbench Browser

iPhone 11 Pro

Single-Core Score	Multi-Core Score
5472	13840

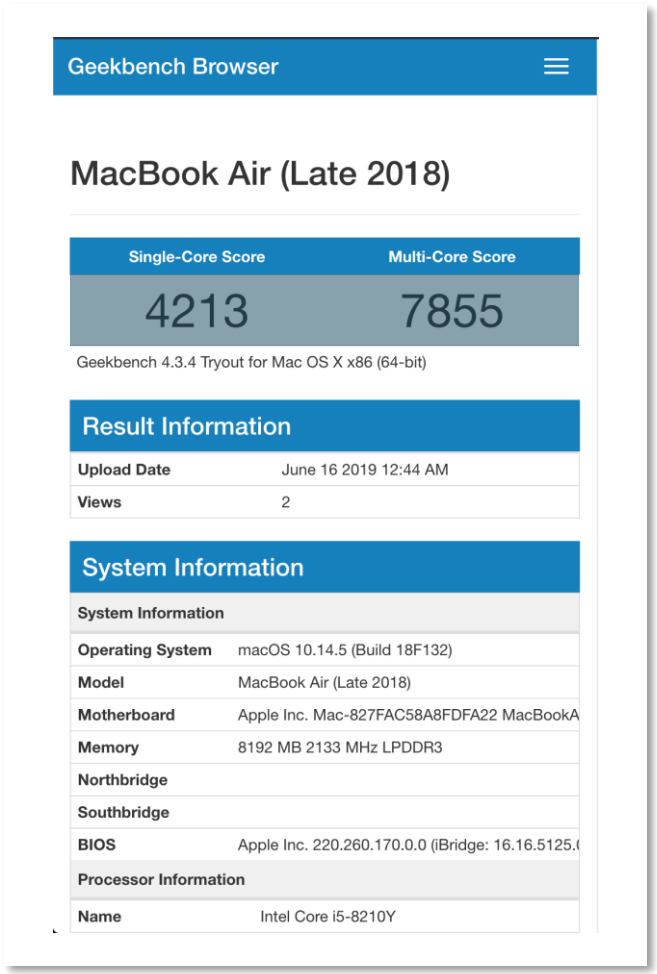
Geekbench 4.4.1 for iOS AArch64

Result Information

Upload Date	September 23 2019 01:12 AM
Views	1

System Information

System Information	
Operating System	iOS 13.1
Model	iPhone12,3
Motherboard	D421AP
Memory	3759 MB
Processor Information	
Name	ARM
Topology	1 Processor, 6 Cores
Identifier	ARM
Base Frequency	2.66 GHz



Geekbench Browser

MacBook Air (Late 2018)

Single-Core Score	Multi-Core Score
4213	7855

Geekbench 4.3.4 Tryout for Mac OS X x86 (64-bit)

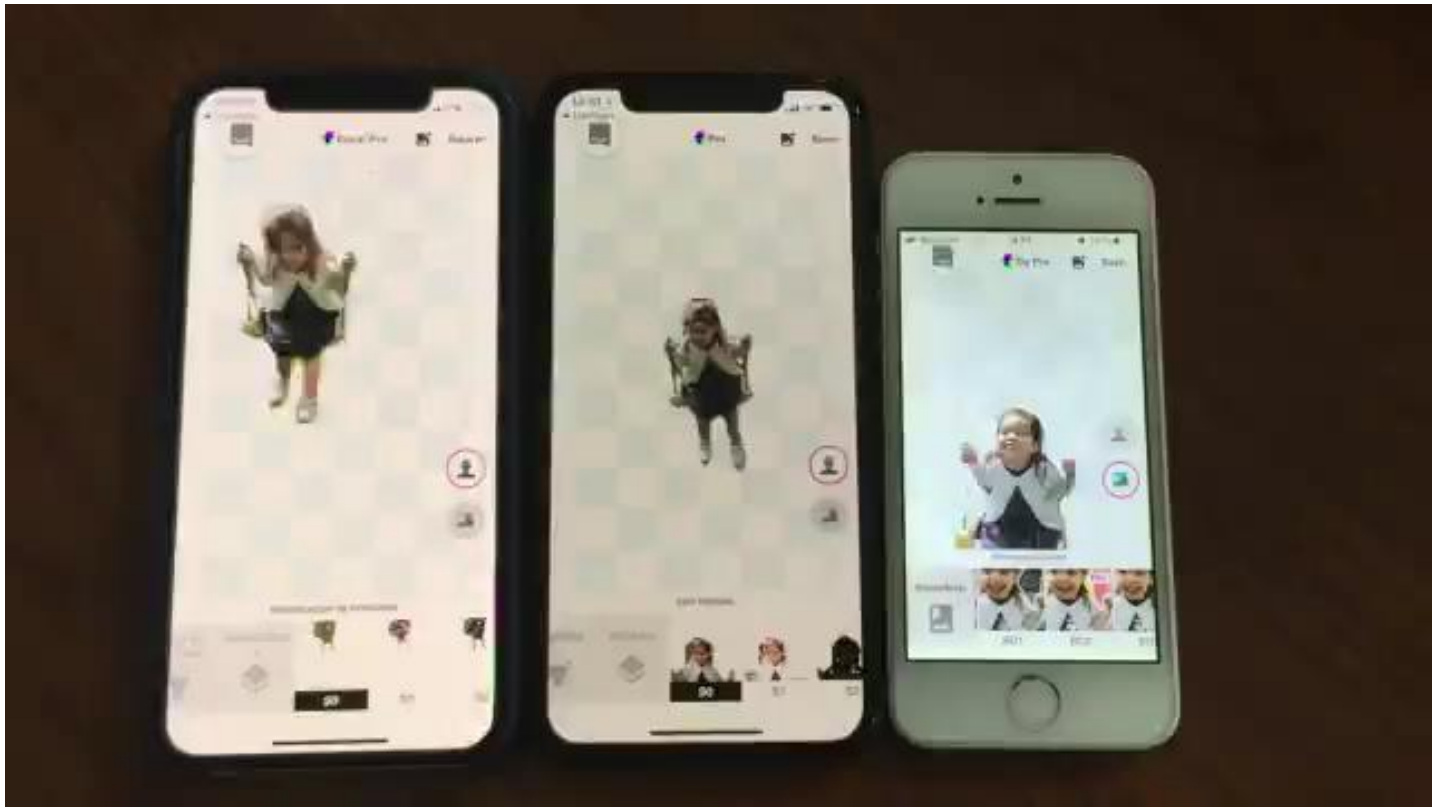
Result Information

Upload Date	June 16 2019 12:44 AM
Views	2

System Information

System Information	
Operating System	macOS 10.14.5 (Build 18F132)
Model	MacBook Air (Late 2018)
Motherboard	Apple Inc. Mac-827FAC58A8FDFA22 MacBookA
Memory	8192 MB 2133 MHz LPDDR3
Northbridge	
Southbridge	
BIOS	Apple Inc. 220.260.170.0.0 (iBridge: 16.16.5125.(
Processor Information	
Name	Intel Core i5-8210Y

Effect of Hardware

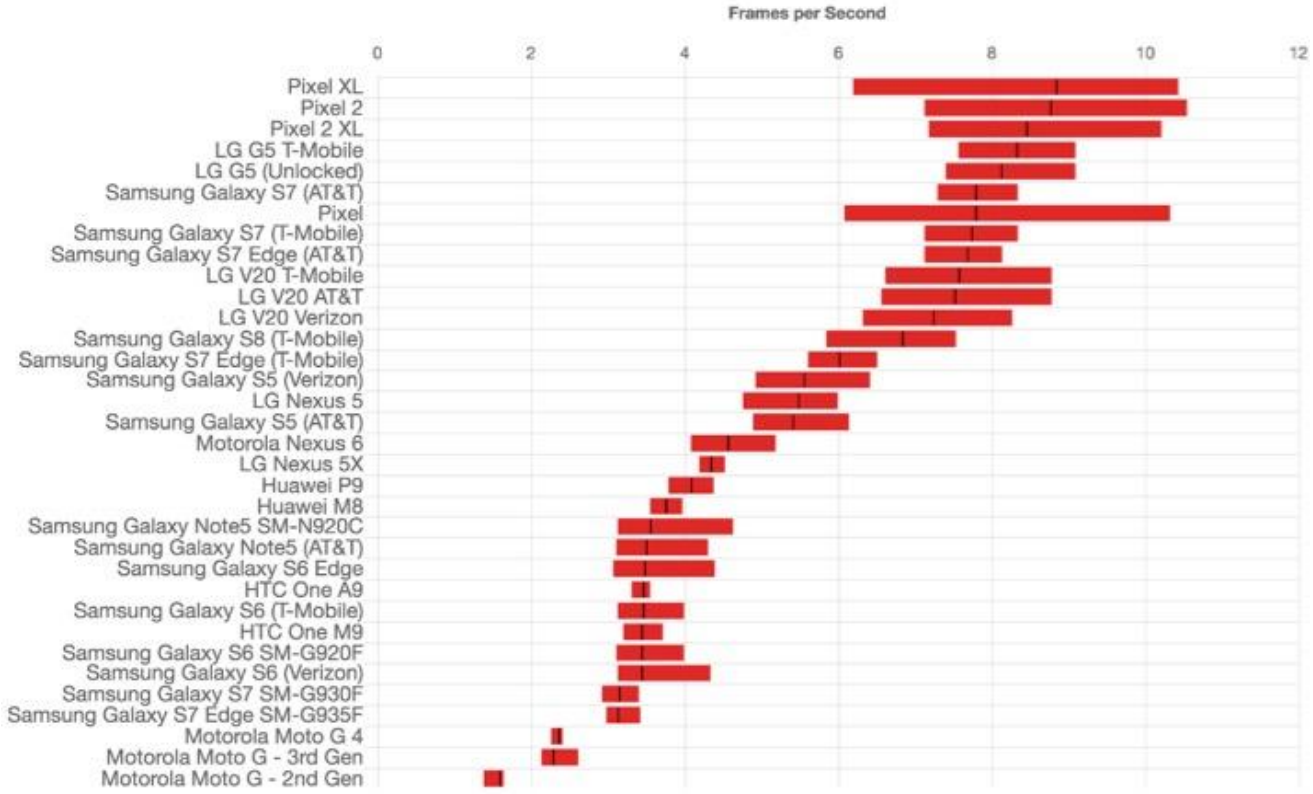


L-R: iPhone XS, iPhone X, iPhone 5

https://twitter.com/matthieurouif/status/1126575118812110854?s=11

TensorFlow Lite benchmarks

Alpha Lab releases Numericcal: http://alpha.lab.numericcal.com/



TensorFlow Lite benchmarks

Crowdsourcing AI Benchmark App by Andrey Ignatov from ETH Zurich. http://ai-benchmark.com/

Model	CPU	RAM	Year	Android	Test 1, ms (Q)	Test 2, ms (F)	Test 3, ms (C)	Test 4, ms (F)	Test 5, ms (F)	Test 6, ms (C)	Test 7, ms (C)	Test 8, ms (F)	Test 9, px (F)	AI-Score
Huawei P20 Pro	HiSilicon Kirin 970 (NPU)	6GB	2018	8.1	144	130	2634	279	241	4390	779	193	6	6519 [1.7]
Huawei Honor 10	HiSilicon Kirin 970 (NPU)	4GB	2018	8.1	107	140	2293	277	239	4476	606	194	6	6496 [1.7]
Huawei P20	HiSilicon Kirin 970 (NPU)	4GB	2018	8.1	166	133	2541	273	242	5104	742	195	6	6444 [1.7]
Mediatek P60 Dev Platform	Mediatek Helio P60	4GB	2018	8.1 proto	21	439	2230	846	1419	4499	394	1562	5	2257 [3.4]
OnePlus 6	Snapdragon 845	8GB	2018	9.0 proto	24	892	1365	928	1999	2885	303	1244	5	2053 [2.4]
Google Pixelbook	Intel Core i5-7Y57	8GB	2017	7.1	75	613	1430	1357	3368	3686	288	1486	14	1794 [5]
HTC U12+	Snapdragon 845	6GB	2018	8.0	60	620	1433	1229	2792	3542	329	1485	11	1708
Asus Zenfone 5z	Snapdragon 845	6GB	2018	8.0	60	626	1401	1198	2788	3477	326	1439	10	1698
Samsung Galaxy S9+	Exynos 9810 Octa	6GB	2018	8.0	148	1208	1572	958	1672	2430	612	1230	8	1628
Samsung Galaxy S9+	Snapdragon 845	6GB	2018	8.0	65	651	1459	1239	2681	3120	311	1592	8	1590
Samsung Galaxy S9	Snapdragon 845	4GB	2018	8.0	63	690	1516	1273	2856	3587	319	1592	7	1539

Alchemy by Fritz

```
$ fritz model benchmark <path to keras model.h5>
 ...
 ------------------------
 Fritz Model Grade Report
 ------------------------

 Core ML Compatible:                True
 Predicted Runtime (iPhone X):      31.4 ms (31.9 fps)
 Total MFLOPS:                      686.90
 Total Parameters:                  1,258,580
 Fritz Version ID:                  <Version UID>

$ fritz model benchmark --version-uid <Version UID>
```

https://alchemy.fritz.ai/

Python library to analyze and estimate mobile performance

No need to deploy on mobile

Which devices should I support?

To get 95% device coverage, support phones released in the last 4 years

For unsupported phones, offer graceful degradation (lower frame rate, cloud inference, etc.)

Could all of this result in heavy energy use?

★★★☆☆

[redacted] 25/03/2019

Glitches & Battery!!!

Recently, I haven't been able to watch videos properly at ALL. The video glitches within the first few seconds, freezing. Then the only way for me to actually get the video to get out of my screen is the skip thee video. If this app's "a way to keep connected with friends", then at least let me see what my friends are saying!! Also, it's a mega battery drainer and using it while I'm on trips usually ends up being a pain for me.

★☆☆☆☆

[redacted] 31/10/2017

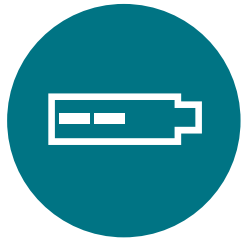
Burning through my battery

Your space-time continuum update broke the app. Watching videos burns through the battery and the phone gets very hot. Lost 20% battery watching a 10min video on my iPhone 7 Plus. Other video streaming apps work as usual.

Energy considerations

You don’t usually run AI models constantly; you run it for a few seconds

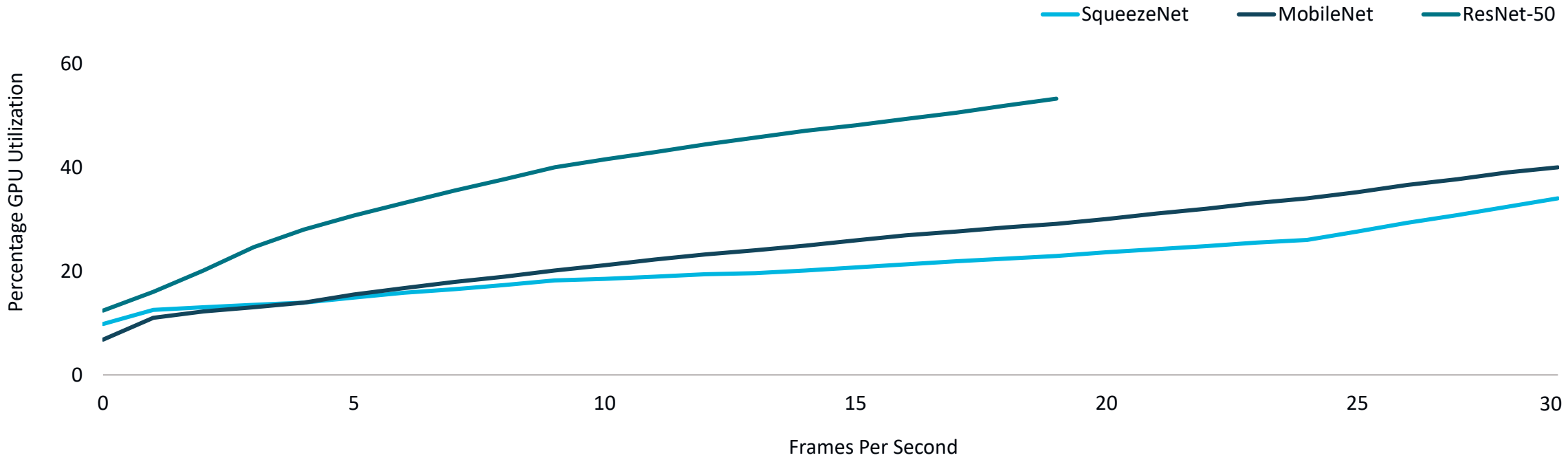
With a modern flagship phone, running MobileNet at 30 FPS should burn battery in 2–3 hours



Bigger question — do you really need to run it at 30 FPS? Could it be run at 1 FPS?

Energy reduction from 30 FPS to 1 FPS

Percentage GPU utilization with varying frames per second



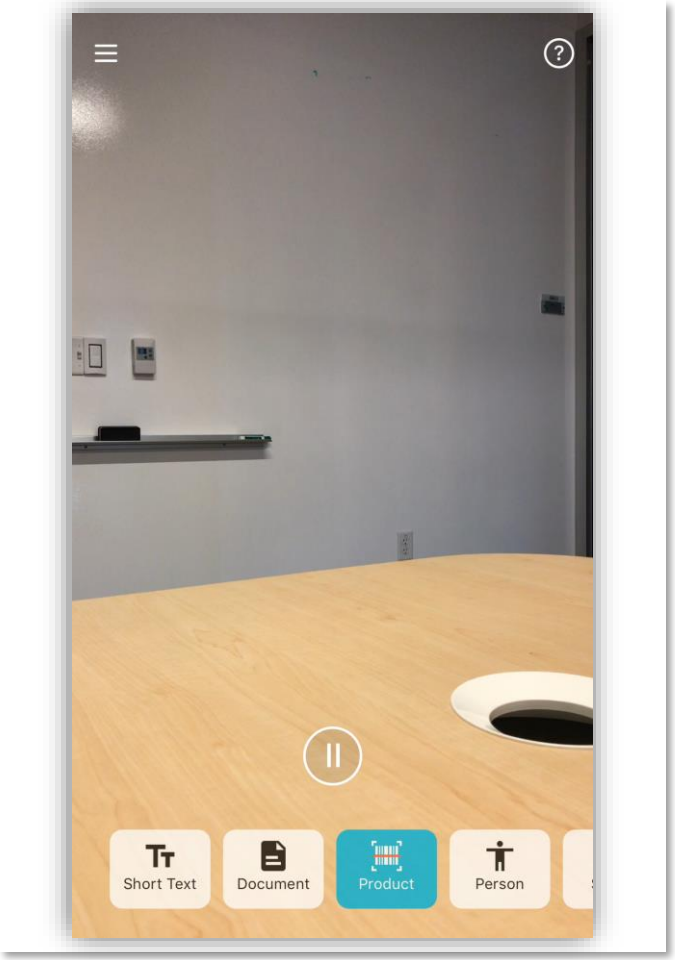
iPad Pro 2017

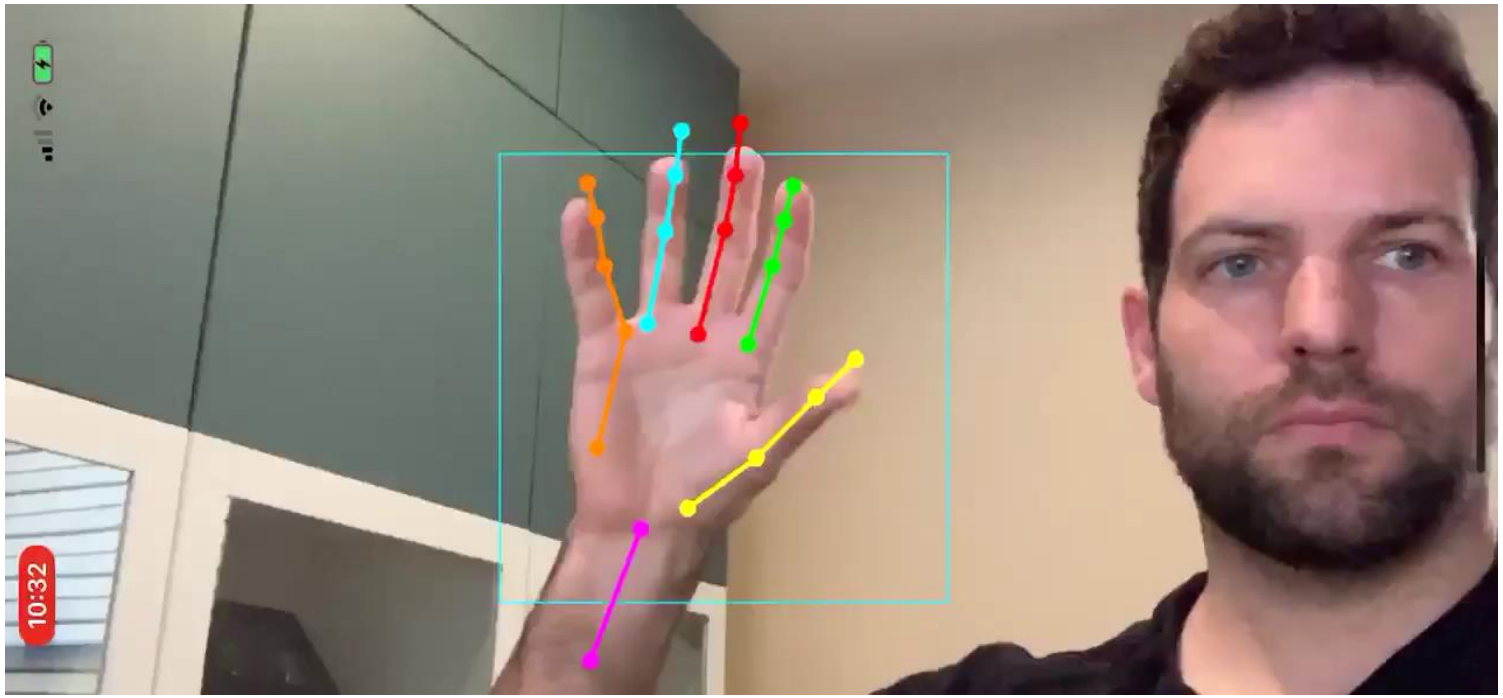
What exciting applications can I build?

Seeing AI

Audible Barcode recognition

Aim: Help blind users identify products using barcode
Issue: Blind users don't know where the barcode is
Solution: Guide user in finding a barcode with audio cues





AR Hand Puppets, Hart Woolery from 2020CV, Object Detection (Hand) + Key Point Estimation

[https://twitter.com/2020cv_inc/status/1093219359676280832]

Object Detection (Ball, Hoop, Player) + Body Pose + Perspective Transformation

[HomeCourt.ai]

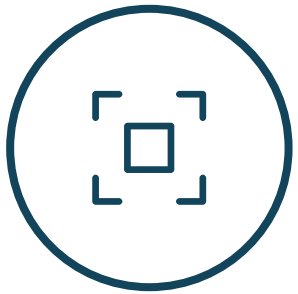
Remove objects

Brian Schulman, Adventurous Co.

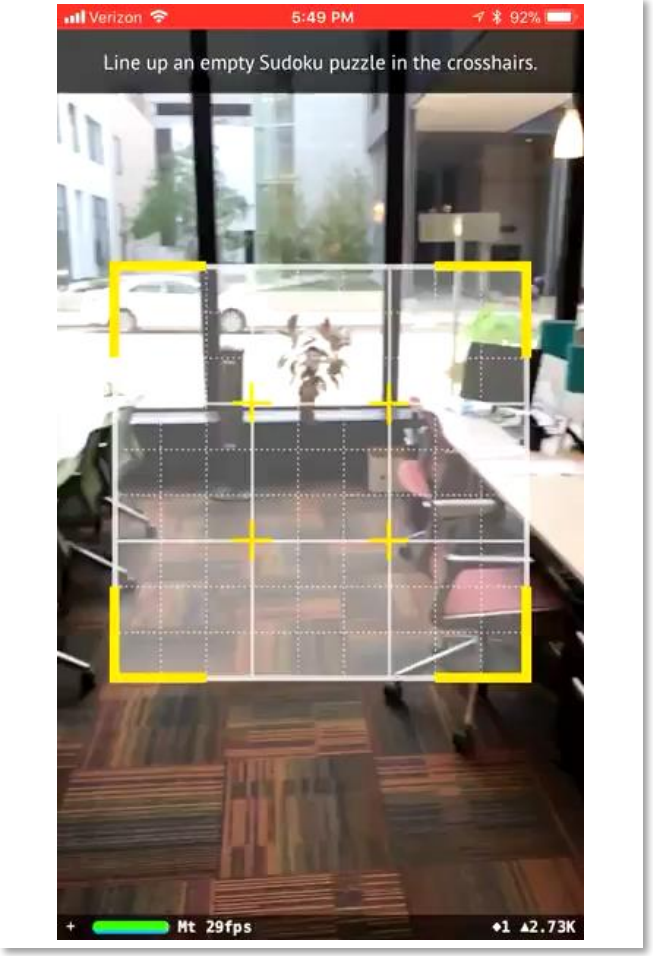
Object Segmentation + Image Inpainting

https://twitter.com/smashfactory/status/1139461813710442496

Magic Sudoku App

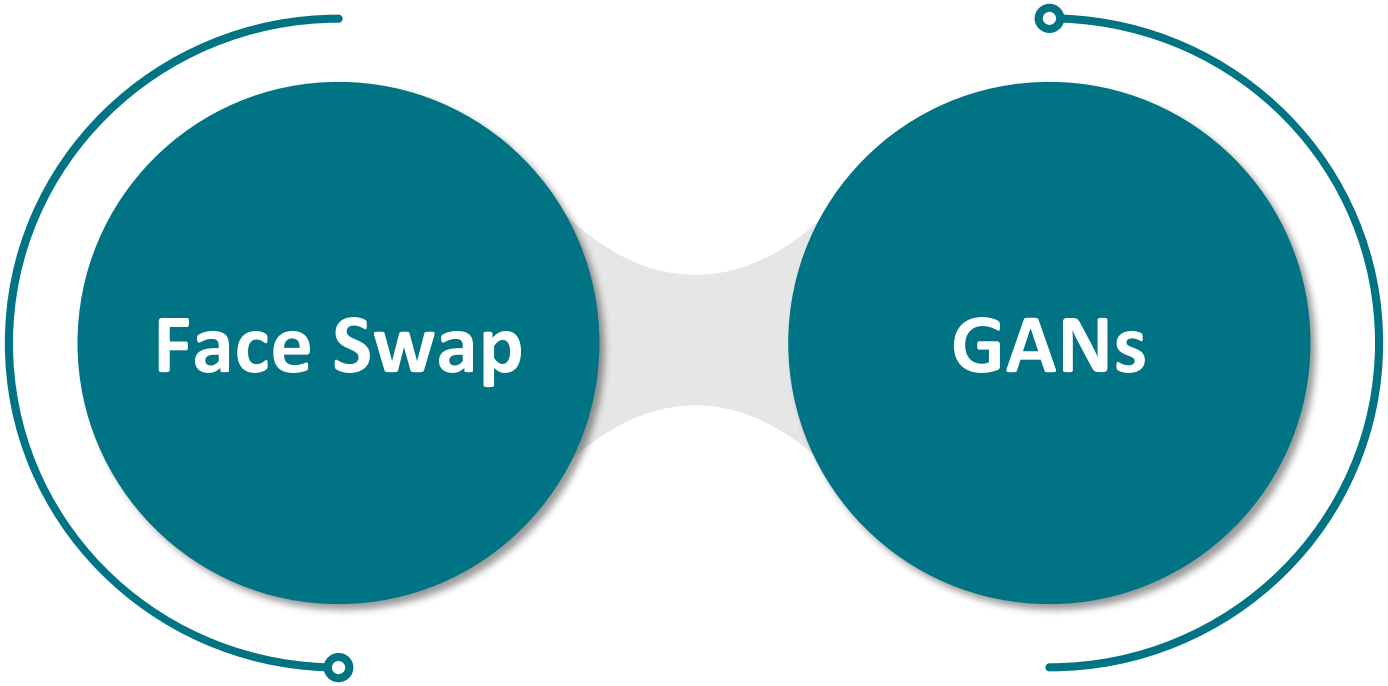


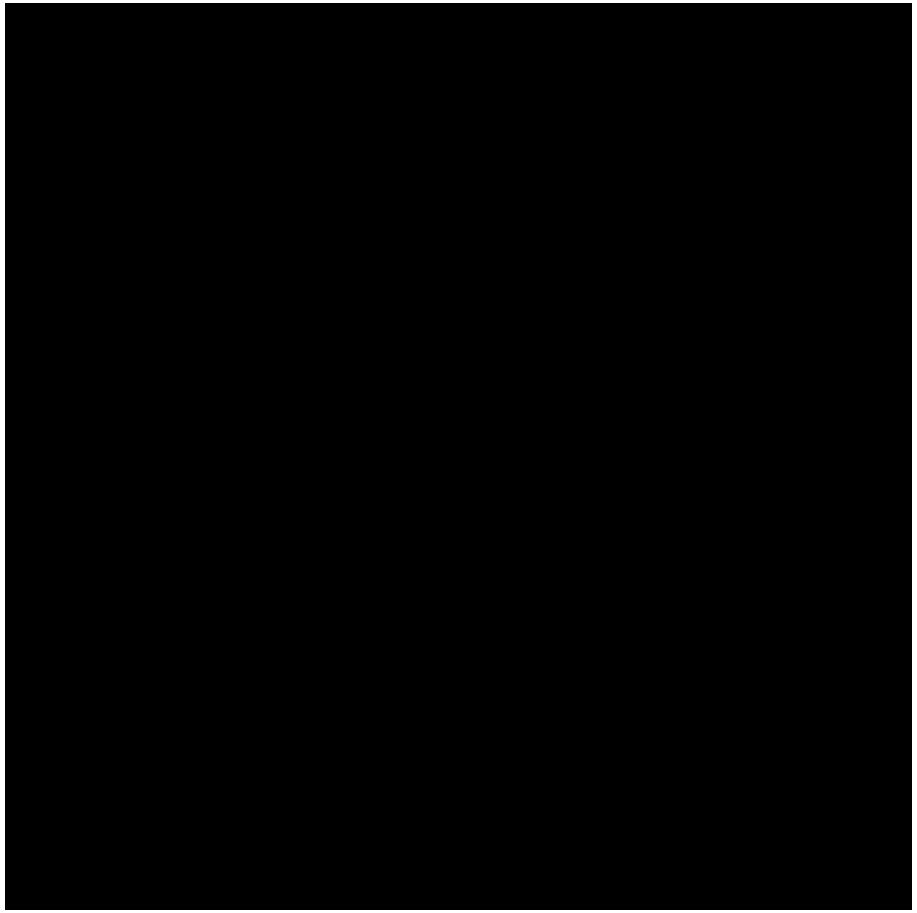
Edge Detection + Classification + AR Kit



https://twitter.com/braddwyer/status/910030265006923776

Snapchat

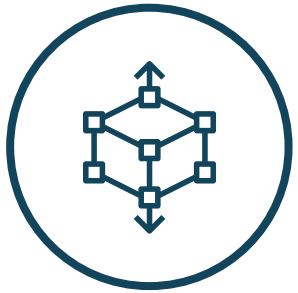




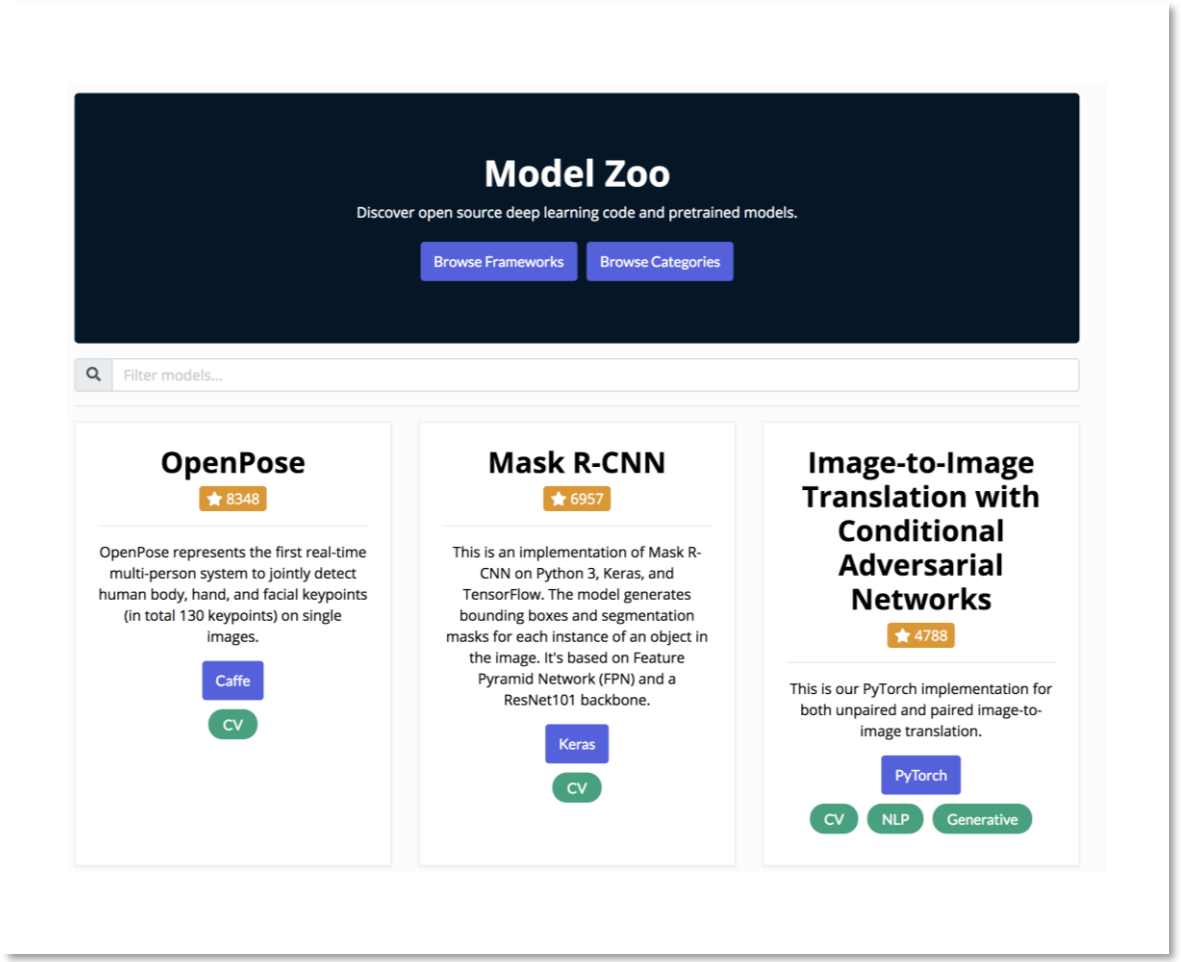
Can I make my model even more efficient?

How To Find Efficient Pre-Trained Models

Papers with Code
https://paperswithcode.com/sota

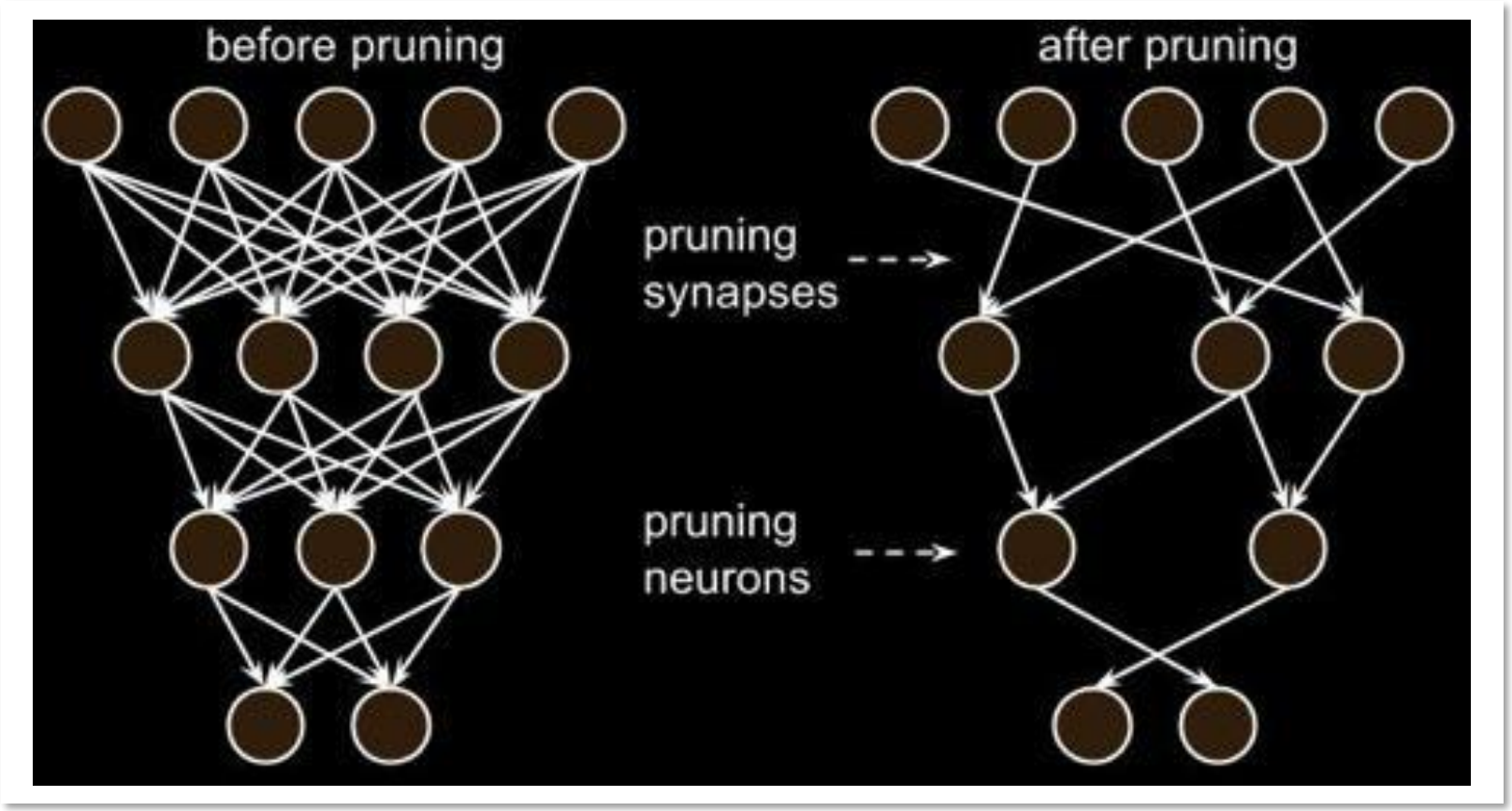


Model Zoo
https://modelzoo.co



Model Pruning

Aim: Remove all connections with absolute weights below a threshold



Song Han, Jeff Pool, John Tran, William J. Dally, "Learning both Weights and Connections for Efficient Neural Networks", 2015

Pruning in Keras

```
model = tf.keras.models.Sequential([
      tf.keras.layers.Flatten(),
      tf.keras.layers.Dense(512, activation=tf.nn.relu),
      tf.keras.layers.Dropout(0.2),
      tf.keras.layers.Dense(10, activation=tf.nn.softmax)
])

model = tf.keras.models.Sequential([
      tf.keras.layers.Flatten(),
      prune.Prune(tf.keras.layers.Dense(512, activation=tf.nn.relu)),
      tf.keras.layers.Dropout(0.2),
      prune.Prune(tf.keras.layers.Dense(10, activation=tf.nn.softmax))
])
```

So many techniques — So little time!

01 Channel pruning

02 Model quantization

03 ThiNet (Filter pruning)

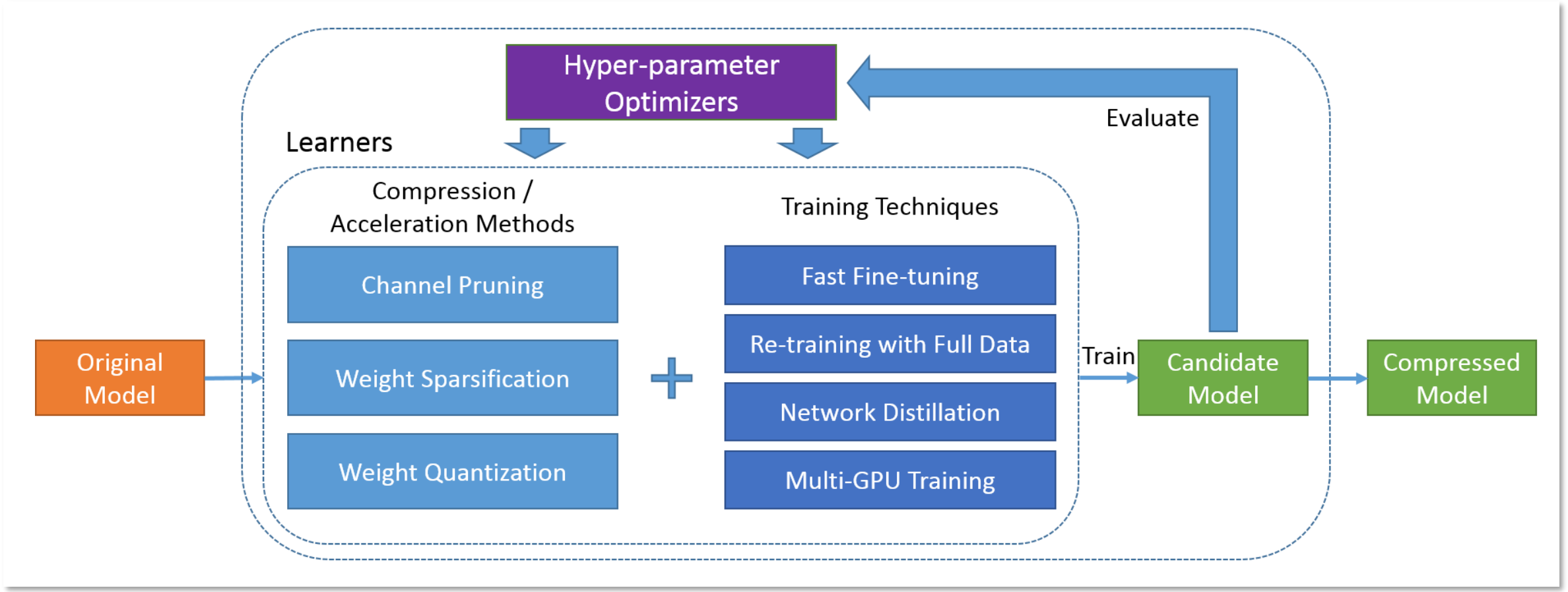
04 Weight sharing

05 Automatic Mixed Precision

06 Network distillation

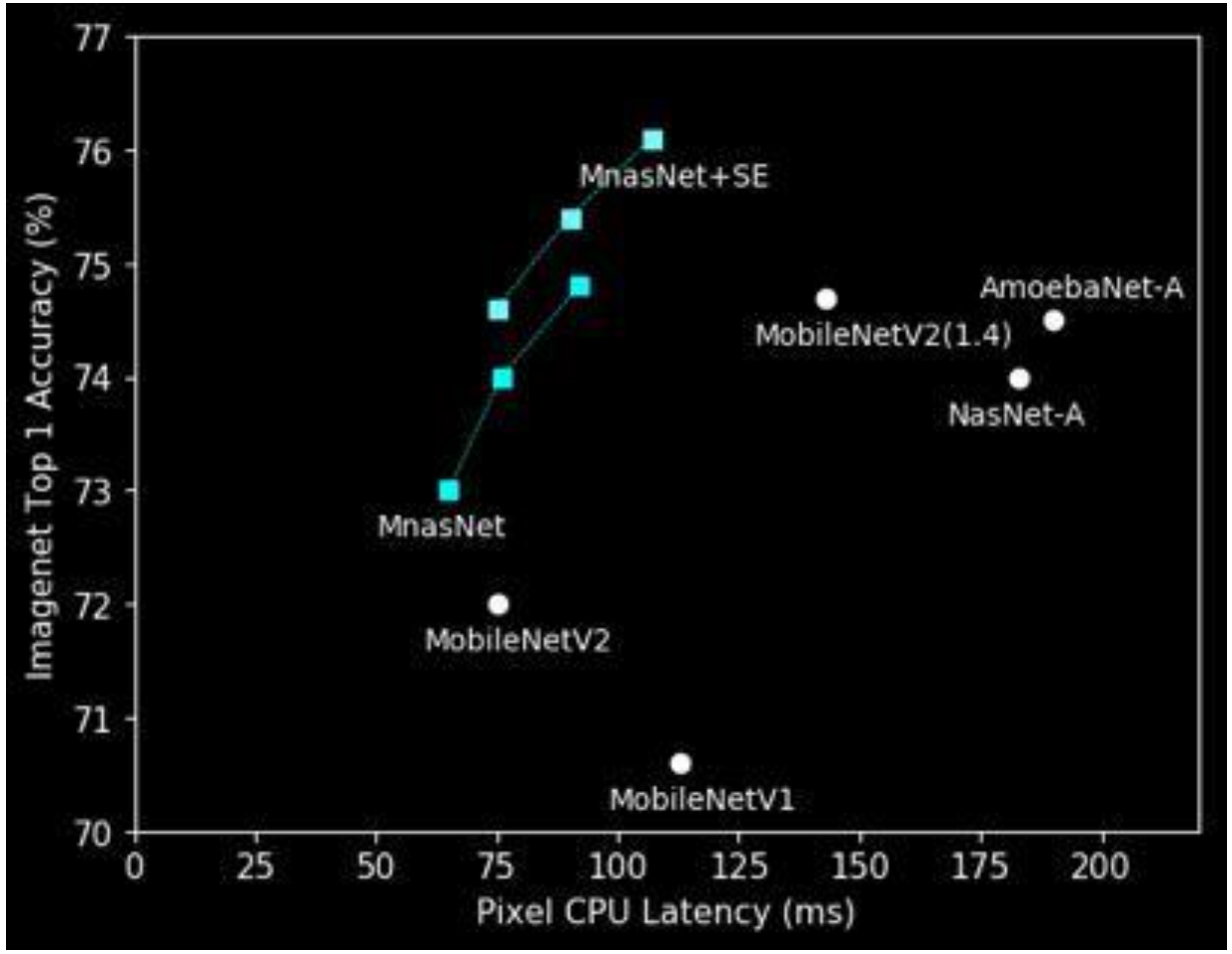
Pocket Flow – 1 Line to Make a Model Efficient

Tencent AI Labs created an Automatic Model Compression (AutoMC) framework



AutoML – Let AI Design an Efficient Arch

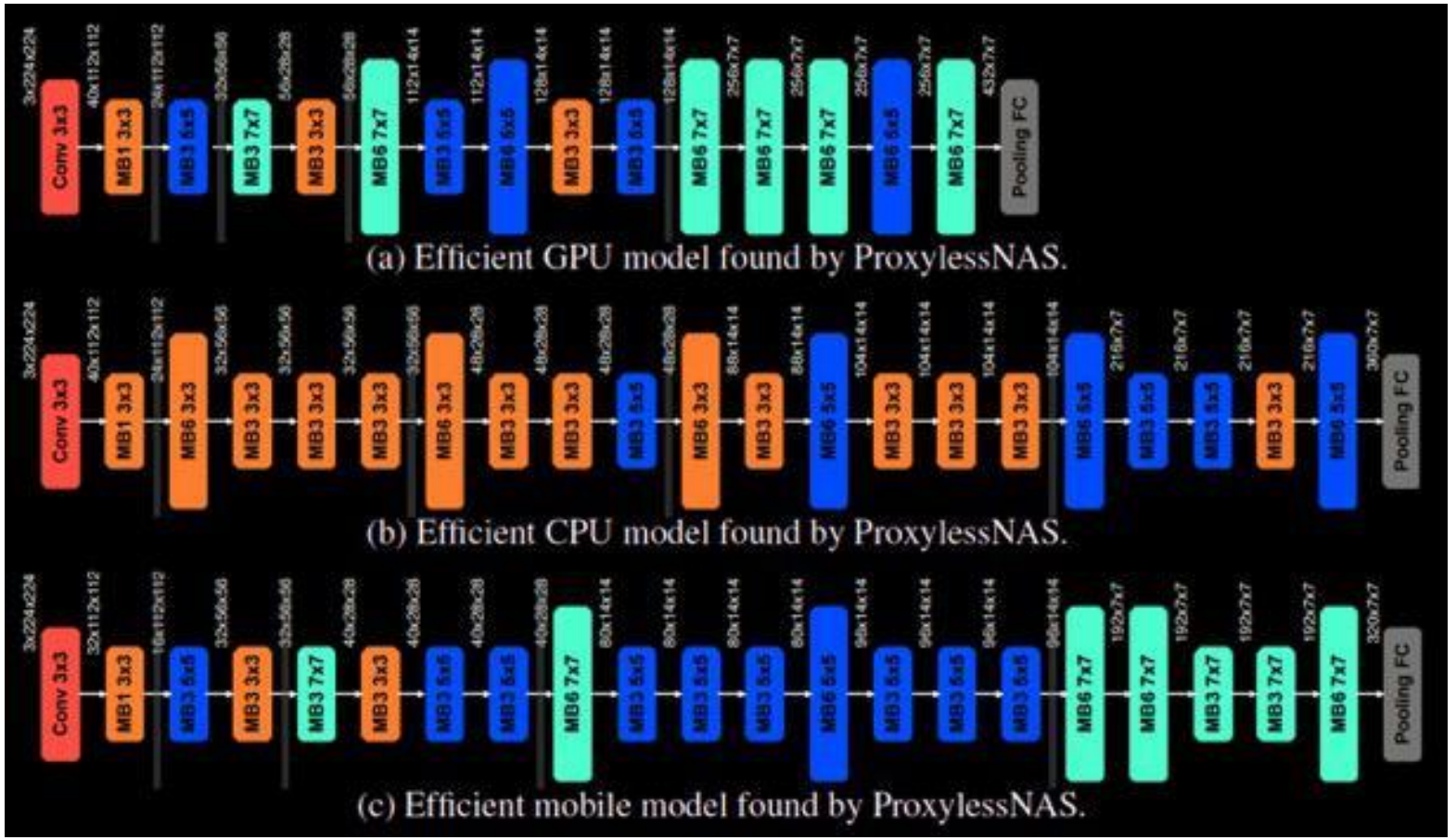
- Neural Architecture Search (NAS) — An automated approach for designing models using reinforcement learning while maximizing accuracy.
- Hardware Aware NAS = Maximizes accuracy while minimizing run-time on device
- Incorporates latency information into the reward objective function
- Measure real-world inference latency by executing on a particular platform
- 1.5x faster than MobileNetV2 (MnasNet)
- ResNet-50 accuracy with 19x less parameters
- SSD300 mAP with 35x fewer FLOPs



Evolution of Mobile NAS Methods

Method	Top-1 Acc (%)	Pixel-1 Runtime	Search Cost (GPU Hours)
MobileNetV1	70.6	113	Manual
MobileNetV2	72.0	75	Manual
MnasNet	74.0	76	40,000 (4 years+)

ProxylessNAS – Per Hardware Tuned CNNs



Han Cai and Ligeng Zhu and Song Han, "ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware", ICLR 2019

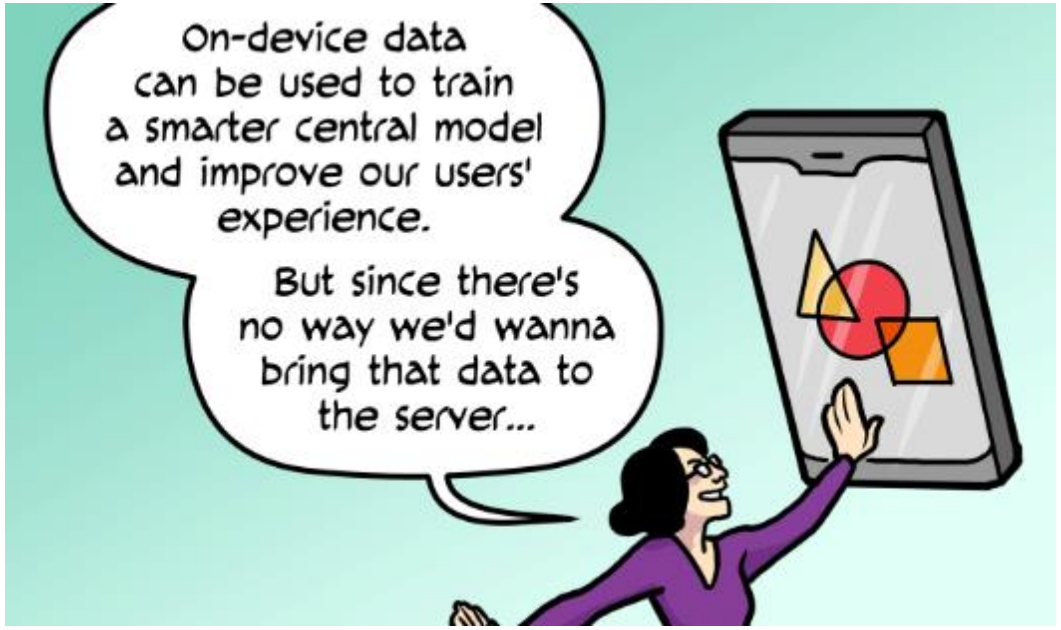
Can I improve my model without accessing user data?

On-Device Training in Core ML

- Core ML 3 introduced on device learning
- Never have to send training data to the server with the help of MLUpdateTask
- Schedule training when device is charging to save power

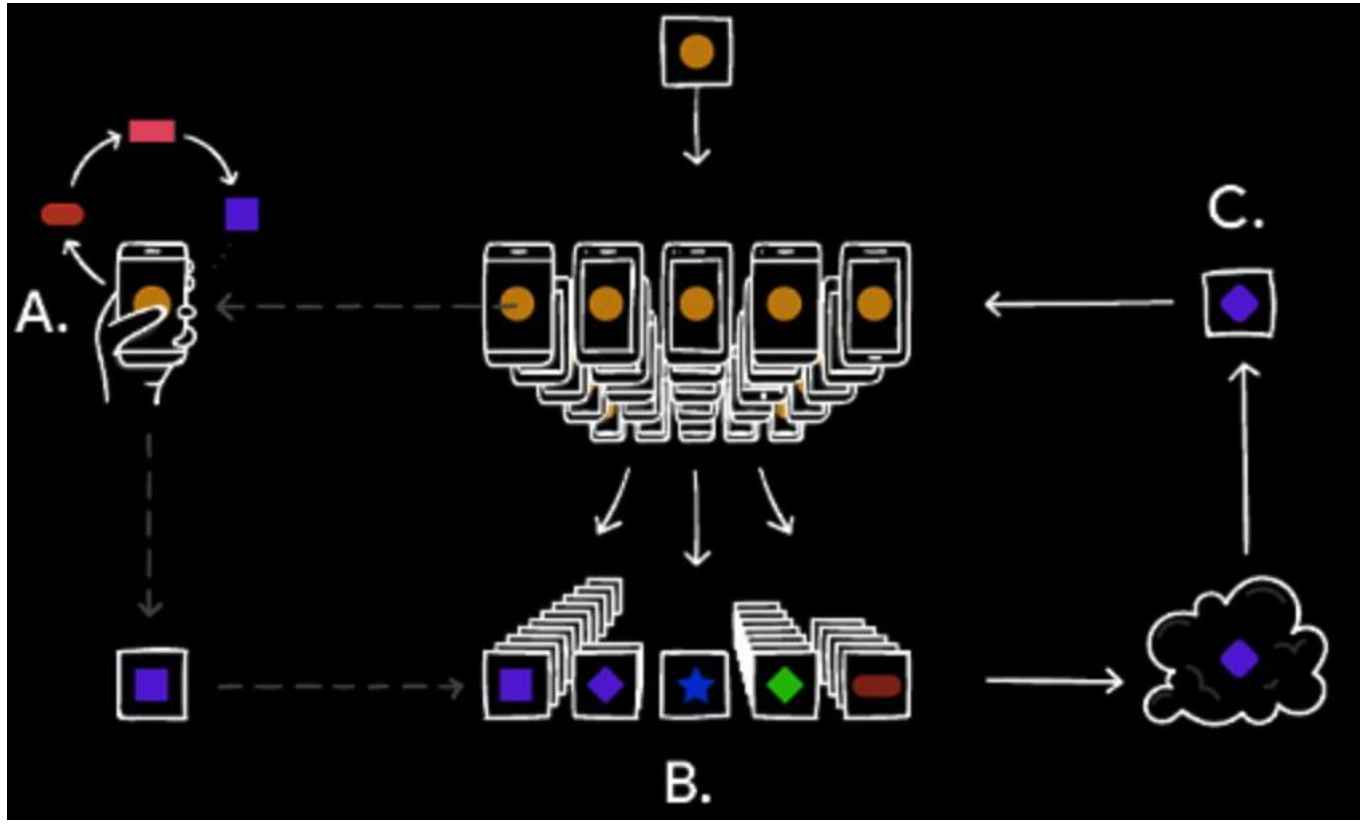
```
let updateTask = try MLUpdateTask(
    forModelAt: modelUrl,
    trainingData: trainingData,
    configuration: configuration,
    completionHandler: { [weak self]
        self.model = context.model context.model.write(to: newModelUrl)
})
```

Federated Learning!!!



https://federated.withgoogle.com/

TensorFlow Federated



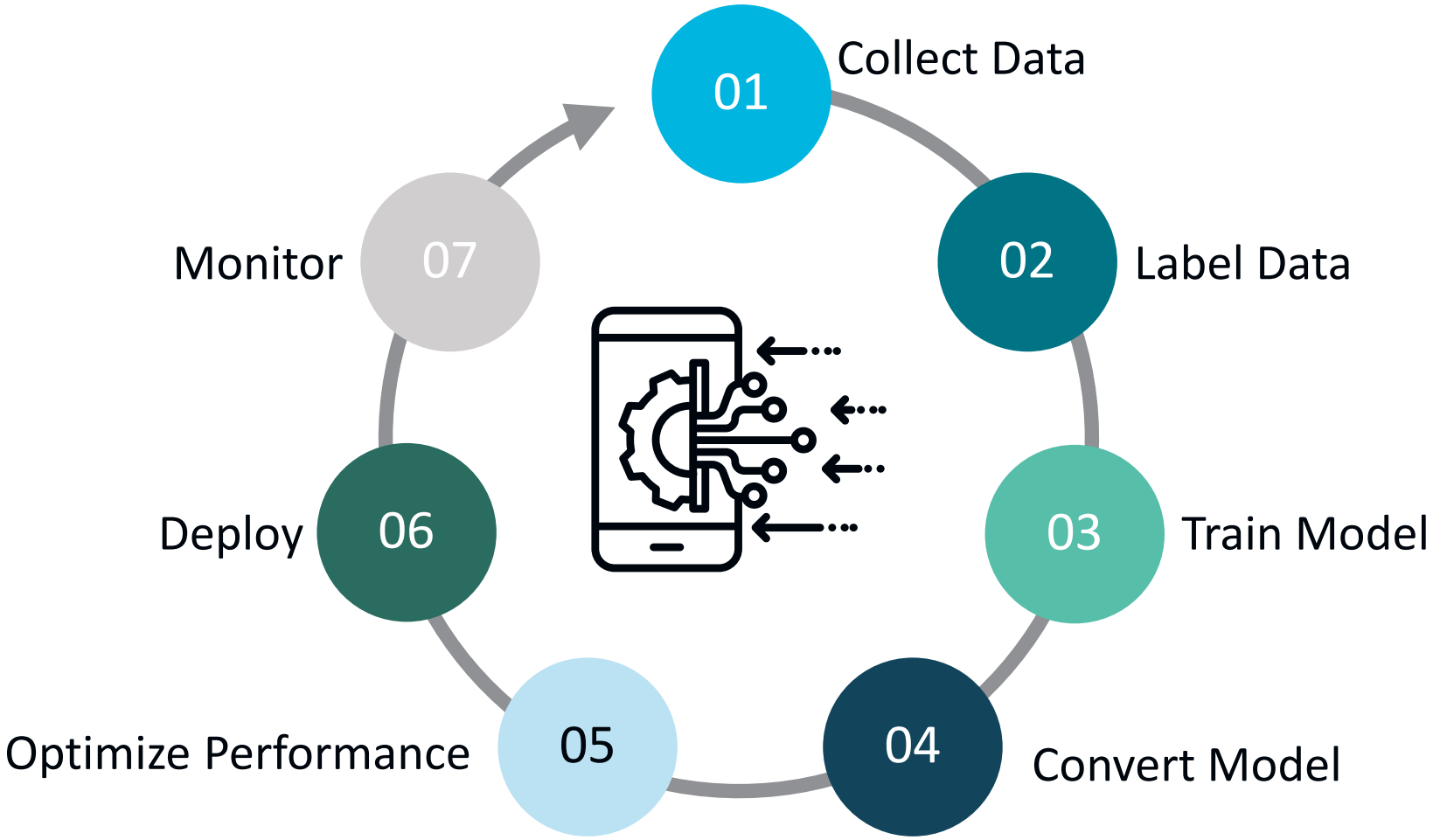
Train a global model using 1000s of devices without access to data

Encryption + Secure Aggregation Protocol

Can take a few days to wait for aggregations to build up

https://github.com/tensorflow/federated

Mobile AI Development Lifecycle



What we learned today

01 Why deep learning on mobile?

02 Building a model

03 Running a model

04 Hardware factors

05 Benchmarking

06 State-of-the-art applications

07 Making a model more efficient

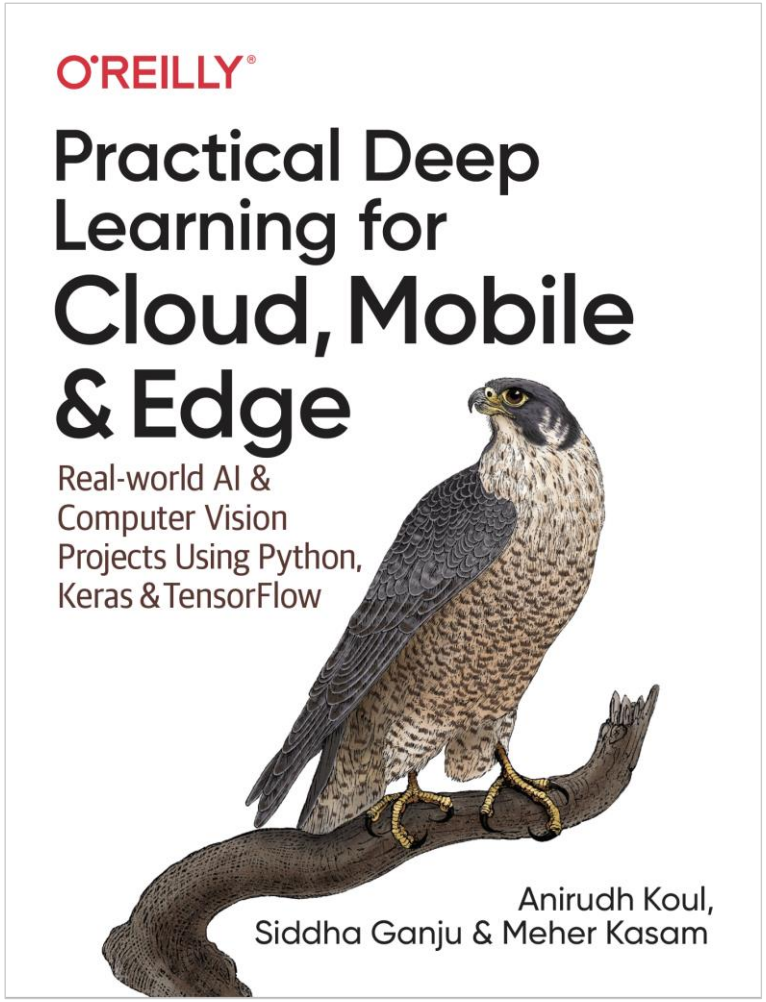
08 Federated learning

How do I access the slides instantly?

http://PracticalDeepLearning.ai

@PracticalDLBook

Icons credit - Orion Icon Library https://orioniconlibrary.com



@SiddhaGanju

@MeherKasam

@AnirudhKoul

That’s all, folks!