WOTD

51CTO

全球软件开发技术峰会

World Of Tech 2017

2017年12月1日-2日 • 深圳中洲万豪酒店

DEVELOPMENT

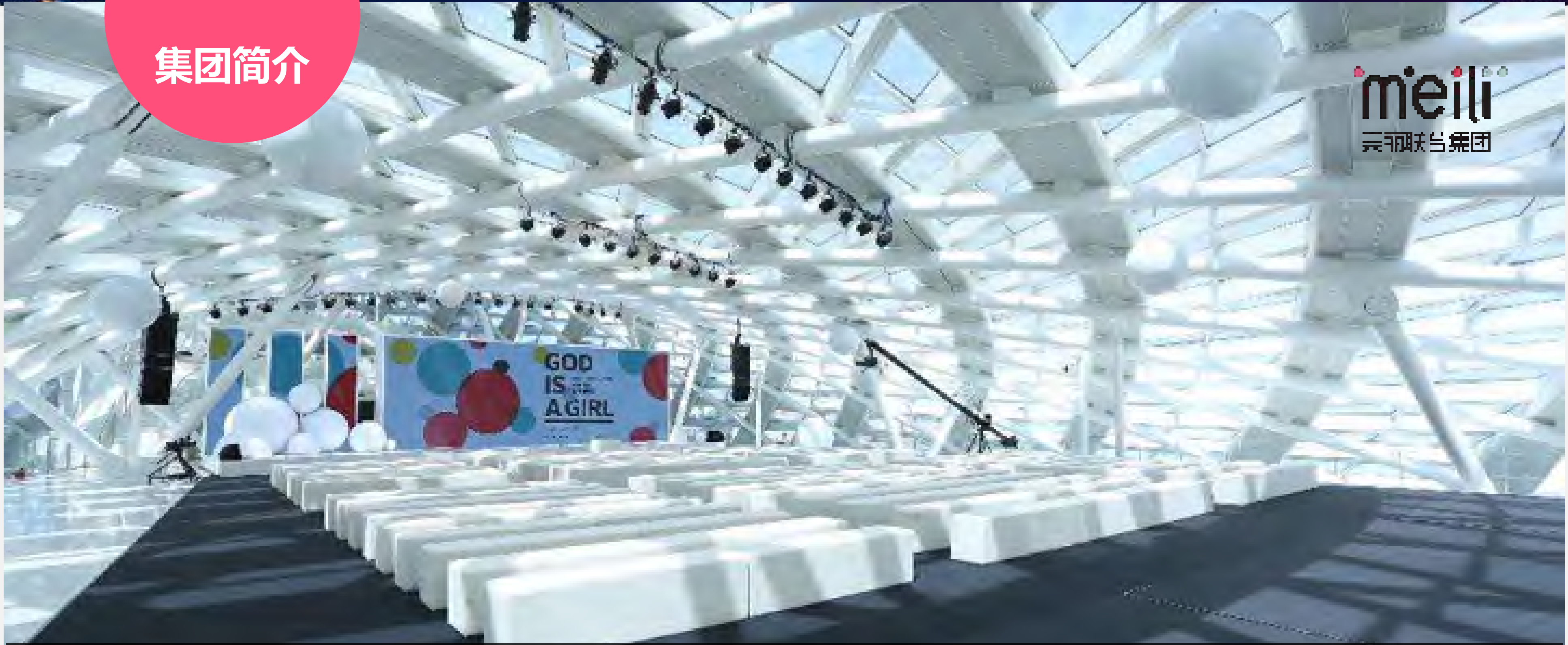
WOTD 51CTO

深度学习在移动端的优化实践

黄文波(鬼谷)
美丽联合集团

meili

集团简介



美丽联合集团是专注服务女性的时尚消费平台，成立于2016 年 6 月 15 日。美丽联合集团旗下包括：蘑菇街、美丽说、uni、锐鲨、MOGU STATION等产品与服务。覆盖时尚消费的各个领域，满足不同年龄层、消费力和审美品位的女性用户日常时尚资讯与时尚消费所需。

整体数据

时尚红人
120,000+

日活用户
10,000,000+

注册用户数
200,000,000+

移动用户占比
95%+

女性用户占比
95%+

成交规模
¥20,000,000,000+

WOTD 51CTO

主要内容

01

背景及现状

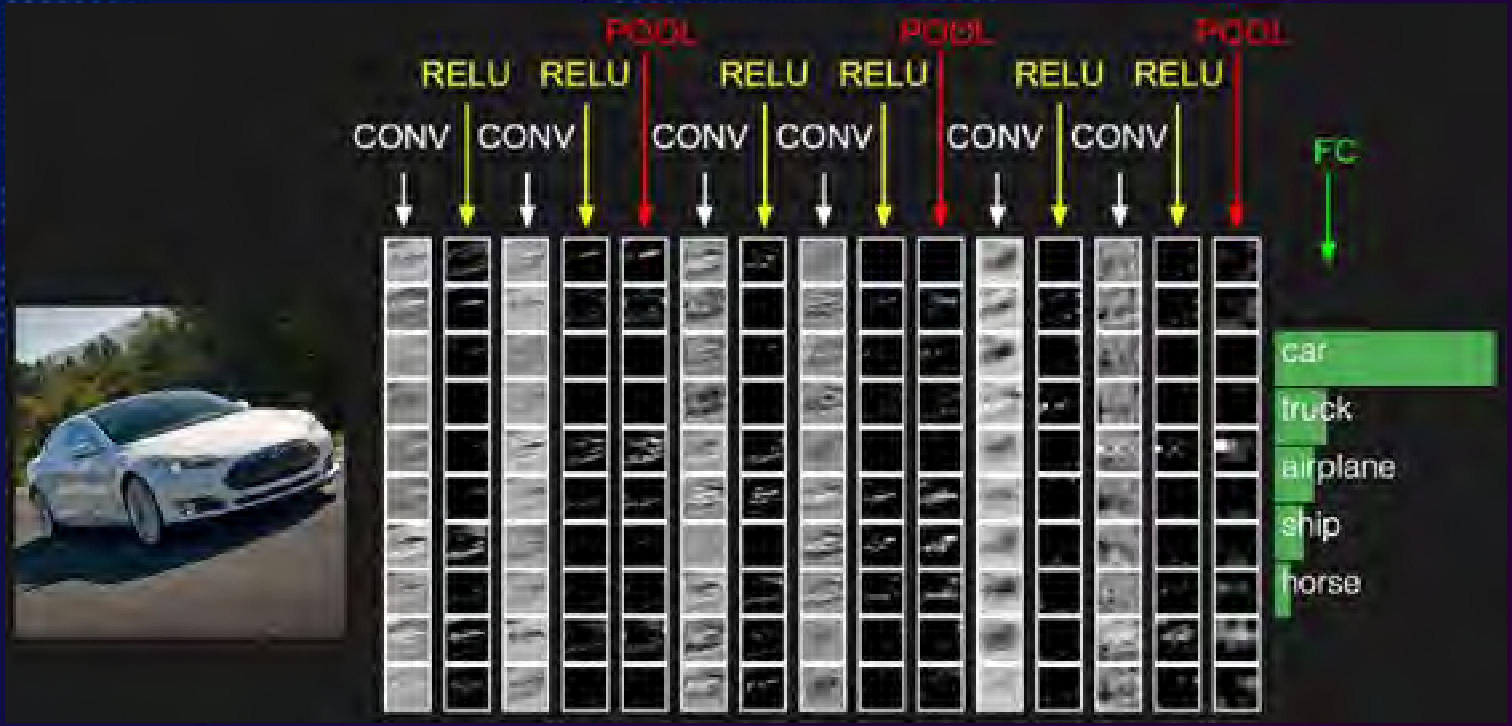
深度学习：从云端到边缘计算

蘑菇街为什么做深度学习优化？

服务器
- 减少训练、预测的时间
- 节约GPU资源，节约电

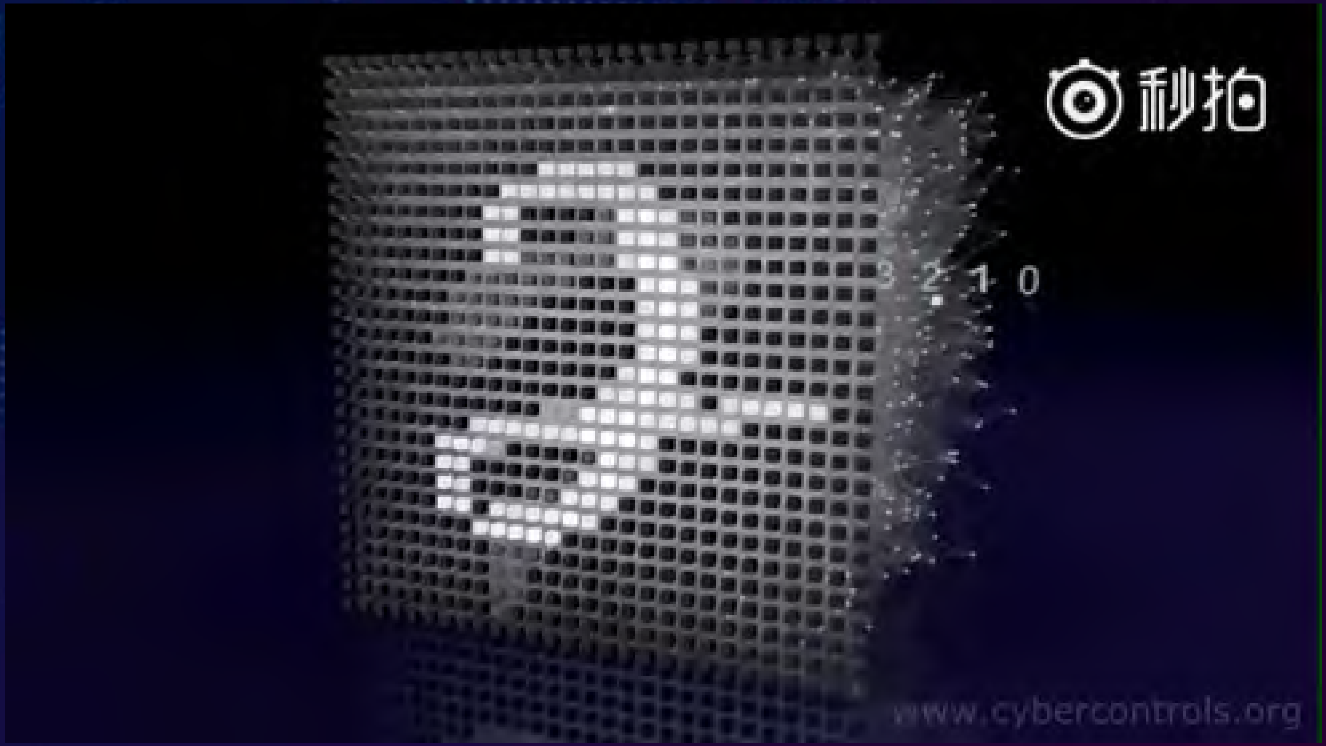
移动端
- 实时响应需求
- 本地化运行，减少服务器压力
- 保护用户隐私

CNN基础



POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

CNN基础



Challenge

深度学习：网络越来越深，准确率越来越高

模型越来越大 → 越多的存储和计算 → 耗费越多能量

移动设备：内存有限、计算性能有限、功耗有限

02

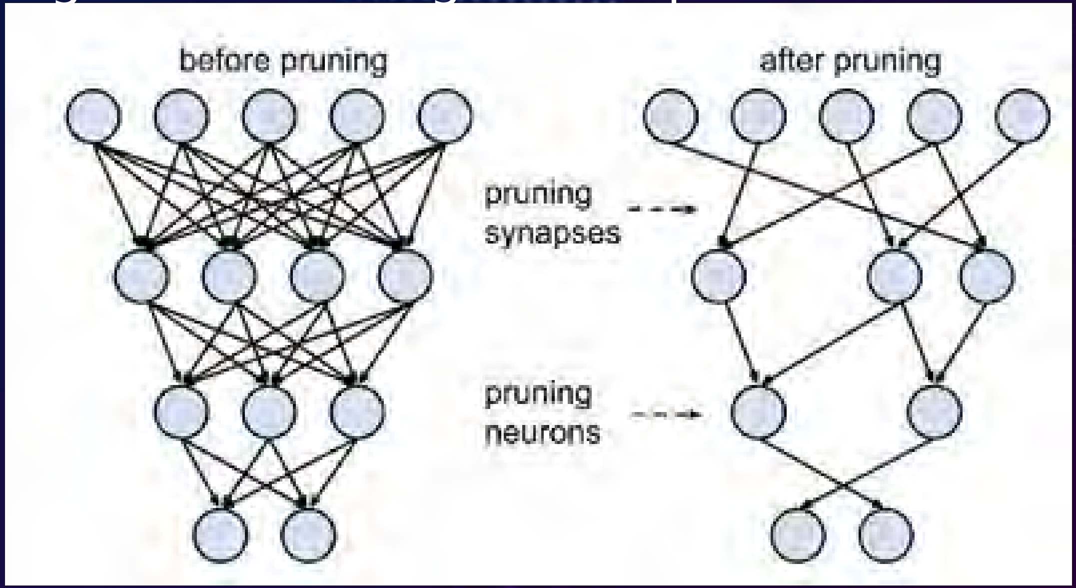
模型压缩与设计

Model Compression

- Pruning
- Quantization
- Huffman Encoding

Pruning

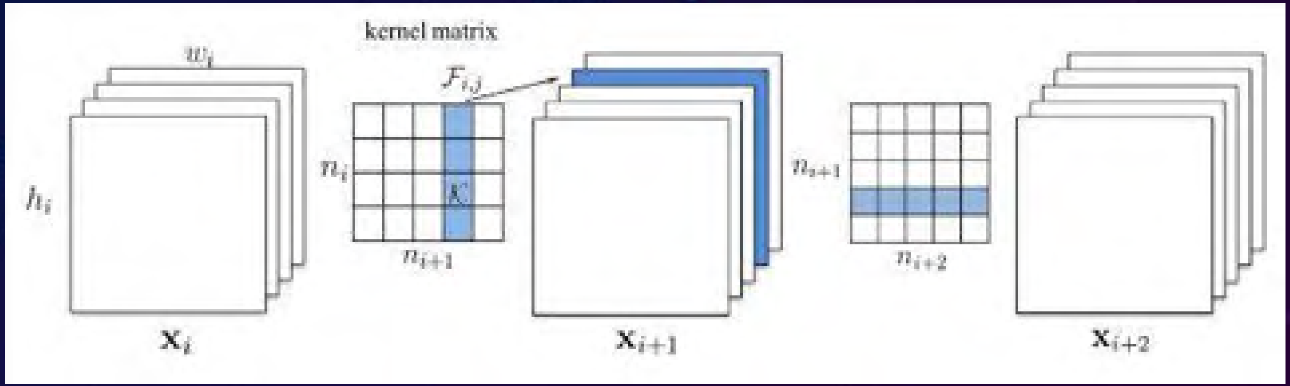
Weight-Level Pruning for the sparse connections



Han et al, “Learning both weights and connections for efficient neural networks”, NIPS 2015

Pruning

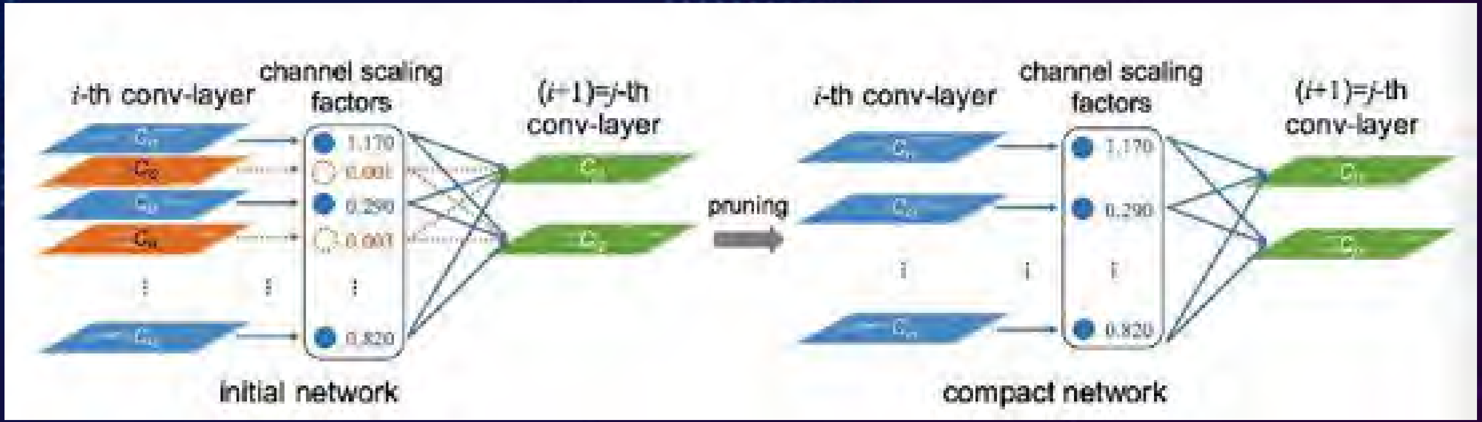
Channel-Level Pruning and retraining iteratively



Li et al, "Pruning filter for efficient convnets", ICLR 2017

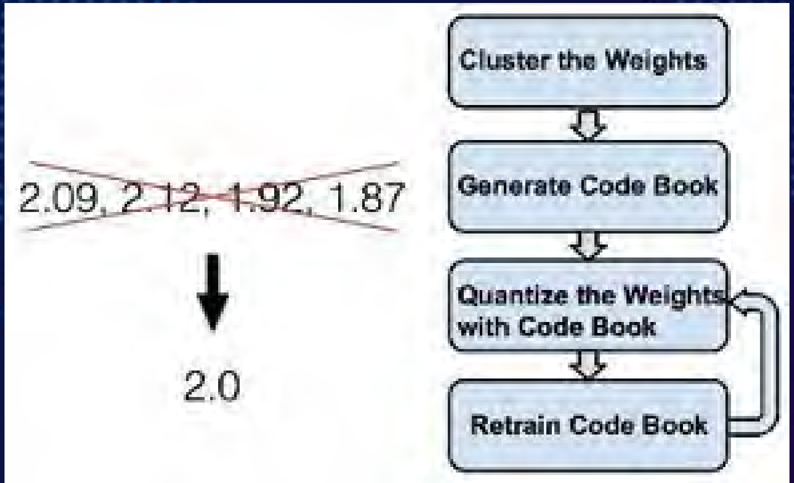
Pruning

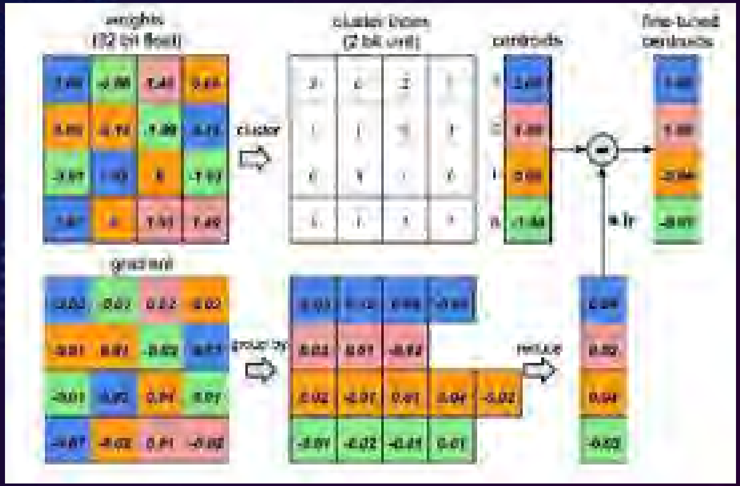
Channel-Level Pruning with L1 regularization



Liu et al, "Learning efficient convolutional networks through network slimming", ICCV 2017

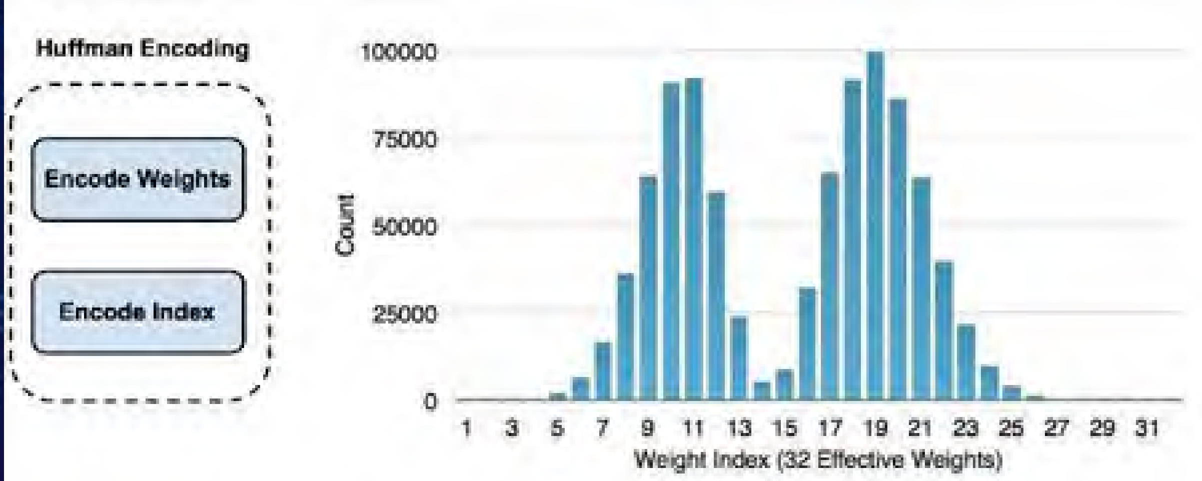
Quantization





Han et al, “Deep Compression: Compressing deep neural networks with pruning, trained quantization and huffman coding”,

Huffman Encoding



- In-frequent weights: use more bits to represent
- Frequent weights: use less bits to represent

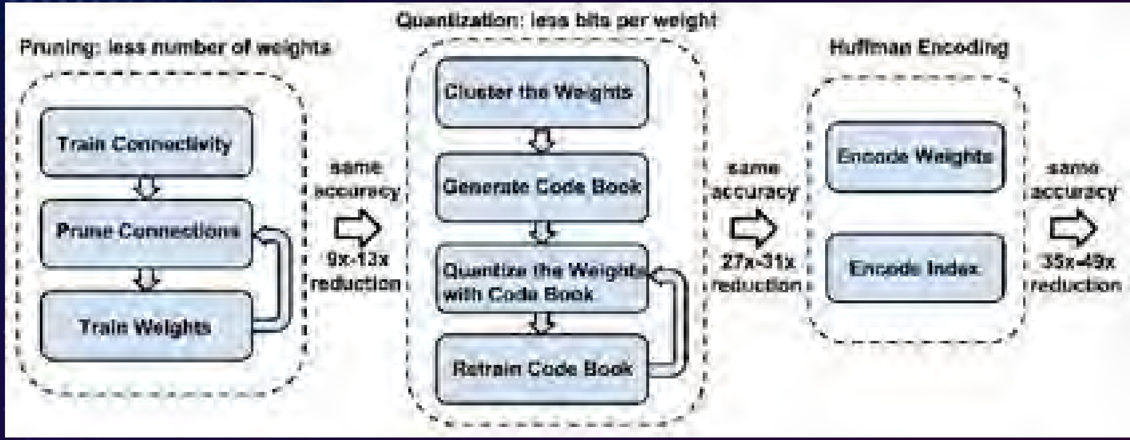
Han et al, “Deep Compression: Compressing deep neural networks with pruning, trained quantization and huffman coding”

Summary of model compression

Pruning: less number of channels

channel-level pruning and retraining iteratively

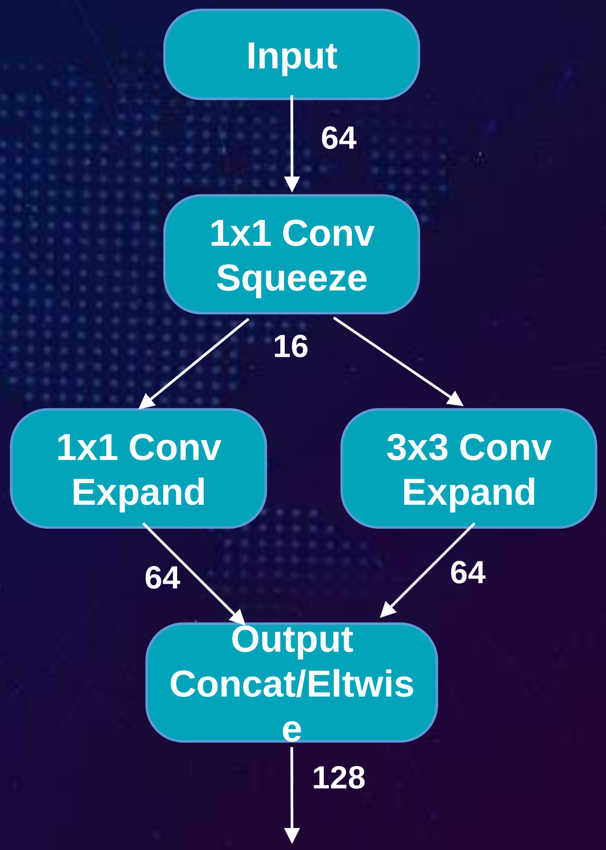
channel-level pruning with L1 regularization



Smaller CNNs architecture design

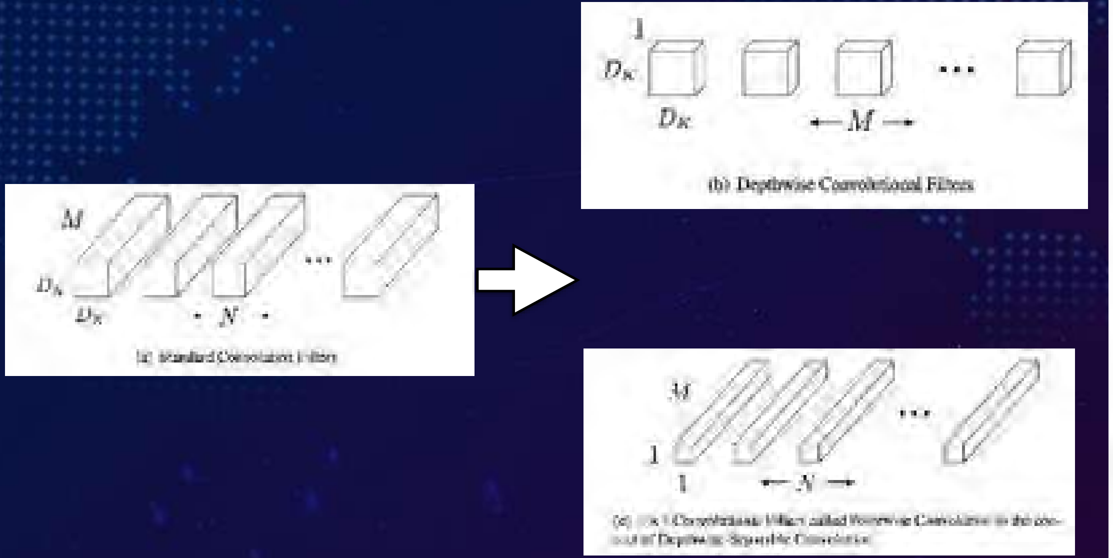
- SqueezeNet
- MobileNet
- ShuffleNet

SqueezeNet



Iandola et al, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5MB model size”, arXiv 2016

MobileNets



(a) Standard Convolution Filters

(b) Depthwise Convolutional Filters

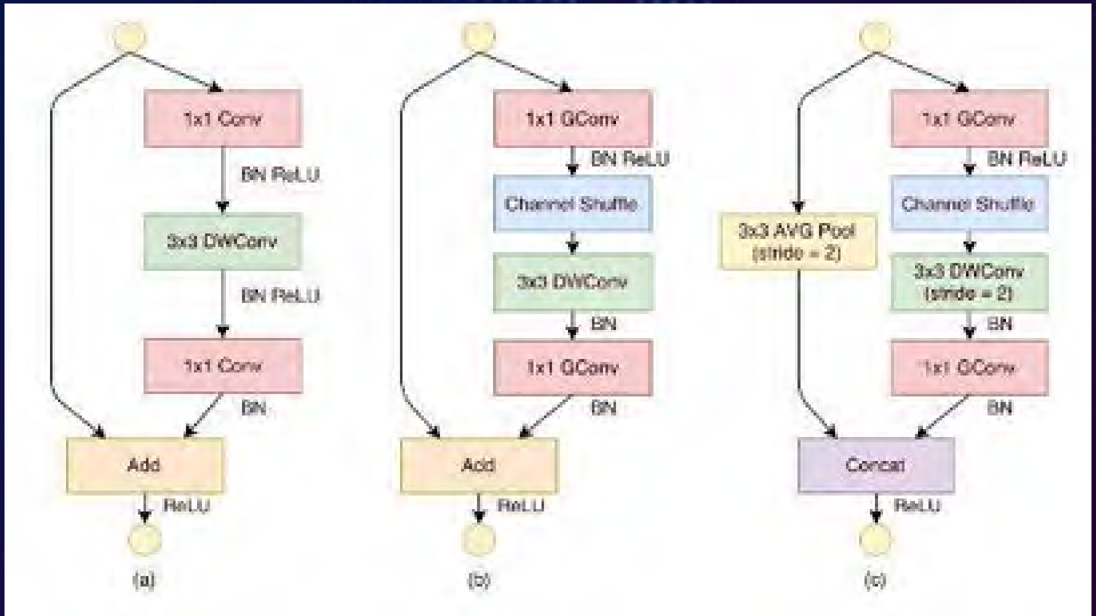
(c) 1×1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	3 × 3 × 3 × 32	224 × 224 × 3
Conv dw / s1	3 × 3 × 32 dw	112 × 112 × 32
Conv / s1	1 × 1 × 32 × 64	112 × 112 × 32
Conv dw / s2	3 × 3 × 64 dw	112 × 112 × 64
Conv / s1	1 × 1 × 64 × 128	56 × 56 × 64
Conv dw / s1	3 × 3 × 128 dw	56 × 56 × 128
Conv / s1	1 × 1 × 128 × 128	56 × 56 × 128
Conv dw / s2	3 × 3 × 128 dw	56 × 56 × 128
Conv / s1	1 × 1 × 128 × 256	28 × 28 × 128
Conv dw / s1	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 256	28 × 28 × 256
Conv dw / s2	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 512	14 × 14 × 256
5× Conv dw / s1	3 × 3 × 512 dw	14 × 14 × 512
5× Conv / s1	1 × 1 × 512 × 512	14 × 14 × 512
Conv dw / s2	3 × 3 × 512 dw	14 × 14 × 512
Conv / s1	1 × 1 × 512 × 1024	7 × 7 × 512
Conv dw / s2	3 × 3 × 1024 dw	7 × 7 × 1024
Conv / s1	1 × 1 × 1024 × 1024	7 × 7 × 1024
Avg Pool / s1	Pool 7 × 7	7 × 7 × 1024
FC / s1	1024 × 1000	1 × 1 × 1024
Softmax / s1	Classifier	1 × 1 × 1000

Howard et al, "MobileNets: Efficient convolutional neural networks for mobile vision applications", arXiv 2017

ShuffleNet



Zhang et al, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices", arXiv 2017

Our practice

Overall Performance of Pruning ResNet50 on ImageNet

Model	strategy	Top-1	Top-5	Model Size
Original	-	75%	92.27%	98M
Pruned-50	Pruning	72.5%	90.9%	49M
Pruned-Q-50	Pruning + Quantization	72.4%	90.6%	15M

Our practice

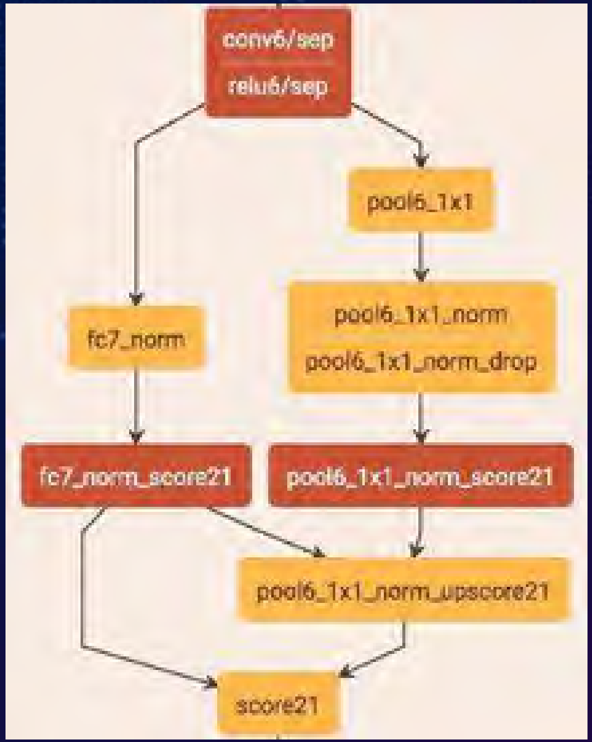
Performance of Pruning ResNet-34 on Our Dataset

Model	Top-1	Top-5	Inference Time	Model Size
Original	48.92%	82.2%	96ms	86M
Pruned-64	48.27%	81.5%	45ms	31M

(2319 categories, 1200W samples)

Our practice

ParseNet 18类(基础网络：MobileNet)



Model	mIOU	Pixel-Level-Accuracy	Model Size
ParseNet	56%	93.5%	13M

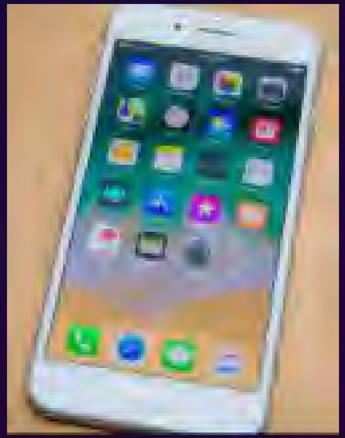
03

移动端工程实践

WOTD 51CTO

移动端服务端分工

Training

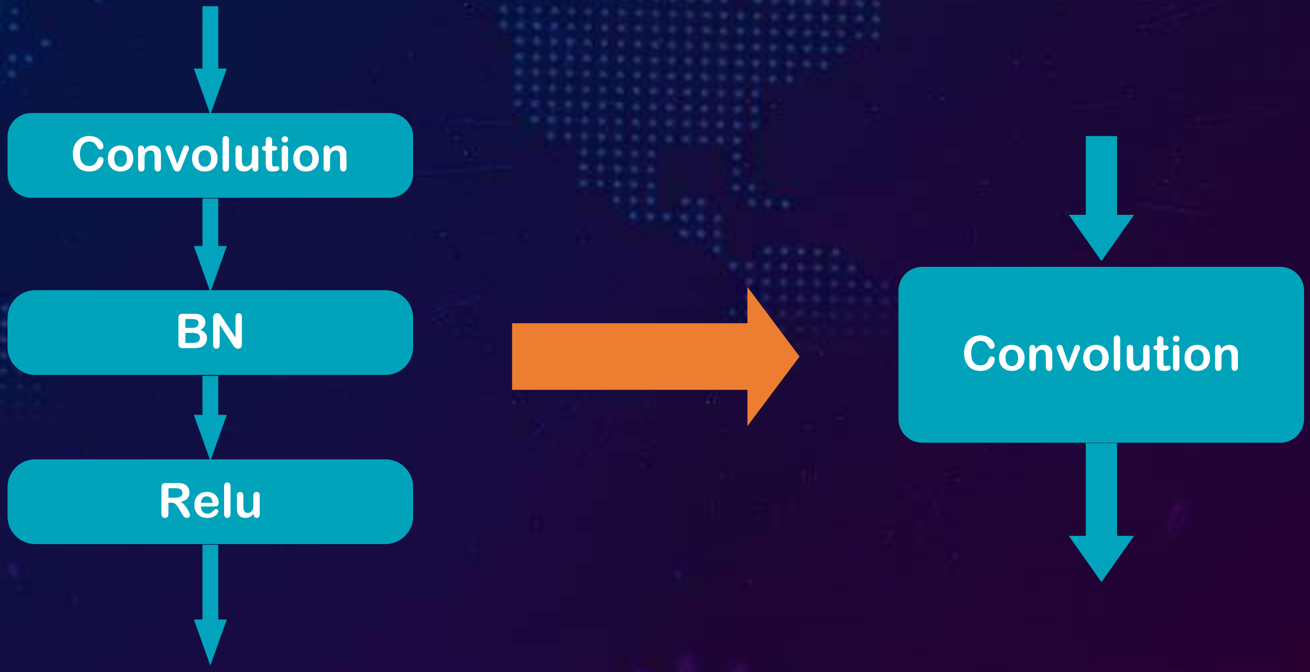


Inference

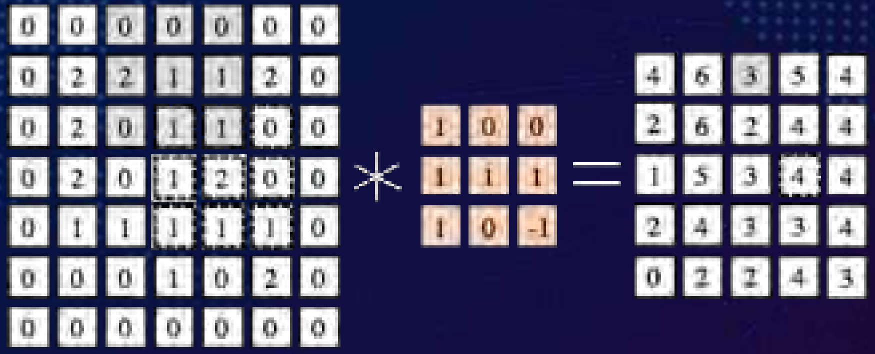
DL frameworks

- Caffe Caffe2 MXNet Tensorflow Torch ....
- NCNN、MDL
- CoreML
- Tensorflow Lite

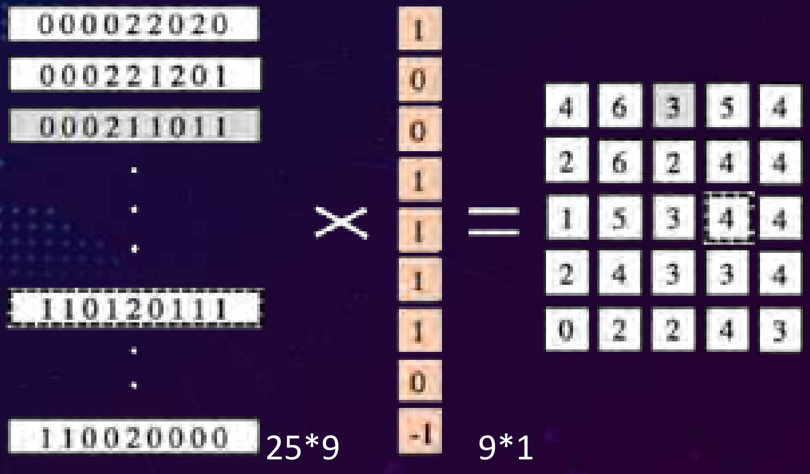
From training to inference



优化卷积计算

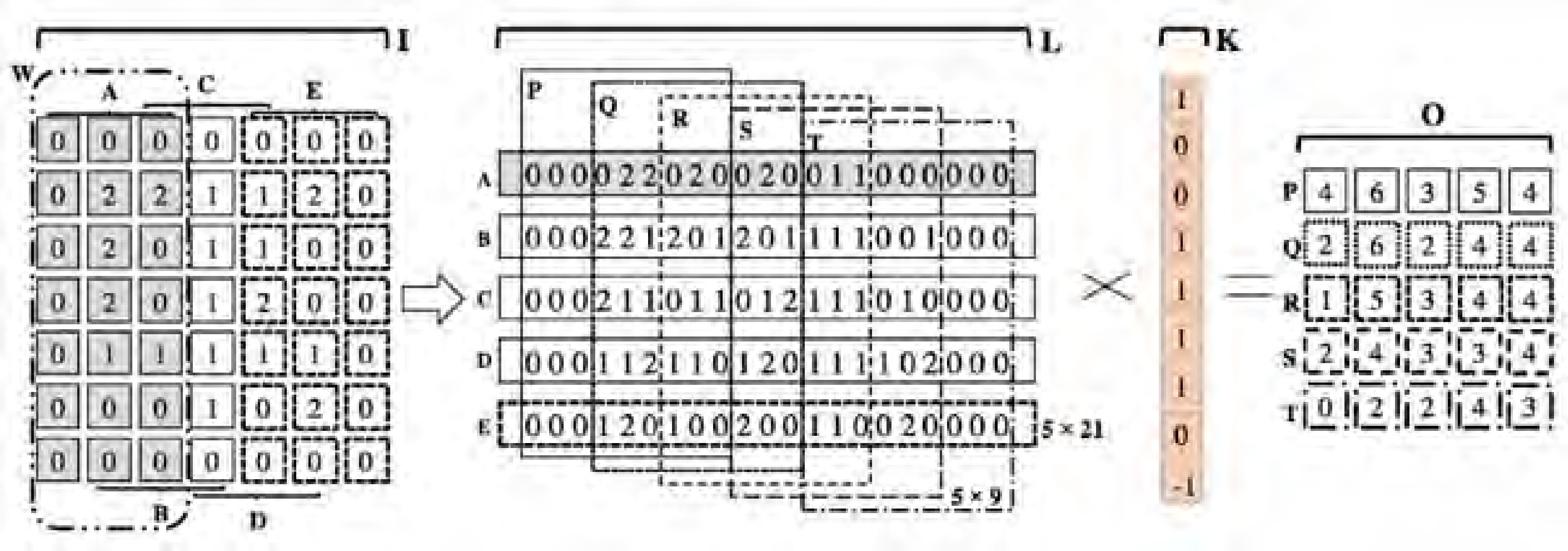


Direct convolution



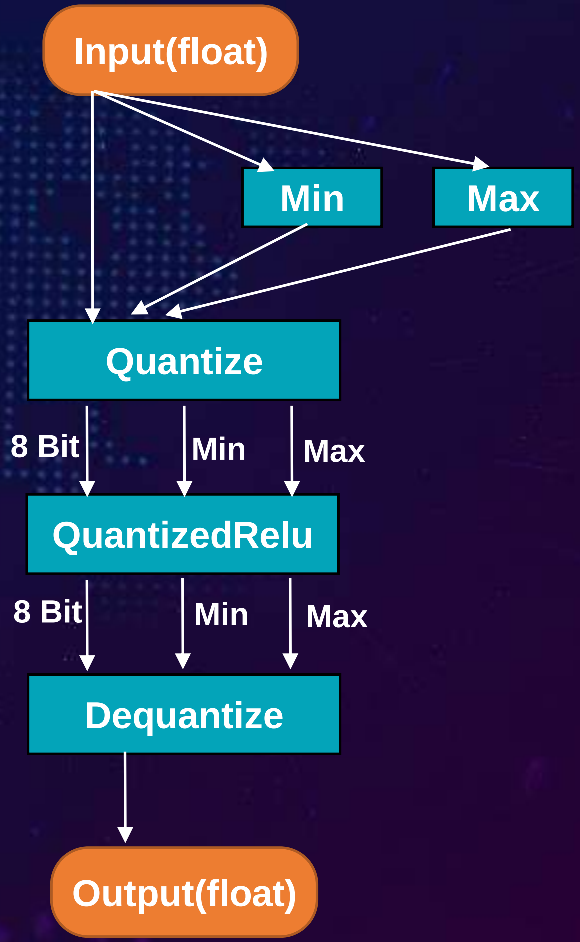
im2col-based convolution

优化卷积计算



Cho et al, “MEC: Memory-efficient convolution for deep neural network”,

浮点运算定点化



Input(float)
Min
Max
Quantize
8 Bit
Min
Max
QuantizedRelu
8 Bit
Min
Max
Dequantize
Output(float)

卷积计算还能怎么进化？

- 再牛逼的优化算法，都不如硬件实现来得直接
- 通用卷积 VS 特定卷积

Android端深度学习框架

NCNN vs MDL

FrameWork	单线程	四线程	内存
NCNN	370ms	200ms	25M
MDL	360ms	190ms	30M

MobileNet on HuaweiP9

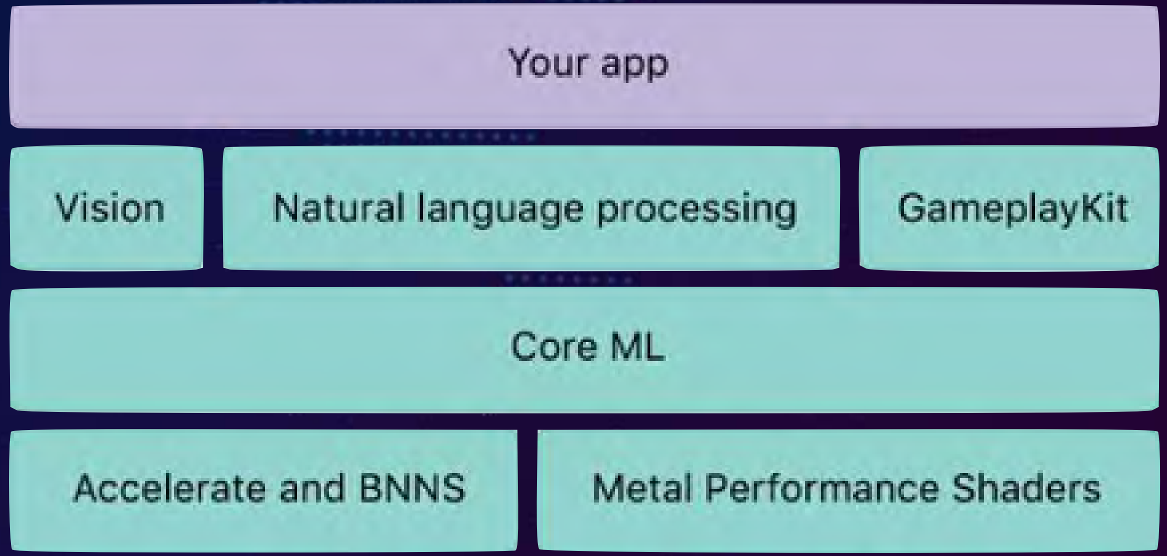
Tensorflow Lite

Quantize MobileNet	Float Mobilenet
85ms	400ms

WOTD
51CTO

iOS 上的DL

CoreM
L



可扩展性不强，不适合部署新算法；需要iOS 11+

MPSCNN

- 充分利用GPU资源，不用抢占CPU
- 利用Metal开发新的层很方便

Tips： 半精度计算；权重存储格式为NHWC

MPSCNN

MPSImage

The layout of a 9-channel CNN image with a width of 3 and a height of 2.

Metal Performance Shader

```
kernel void eltwiseSum_array(
                texture2d_array<half, access::sample> inTexture1 [[texture(0)]],
                texture2d_array<half, access::sample> inTexture2 [[texture(1)]],
                texture2d_array<half, access::write> outTexture [[texture(2)]],
                ushort3 gid [[thread_position_in_grid]])
 {
    if (gid.x >= outTexture.get_width() ||
       gid.y >= outTexture.get_height() ||
       gid.z >= outTexture.get_array_size()) return;
    constexpr sampler s(coord::pixel, filter::nearest, address::clamp_to_zero);
    const ushort2 pos = gid.xy;
    const ushort slice = gid.z;
    half4 in[2];
    in[0] = inTexture1.sample(s, float2(pos.x, pos.y), slice);
    in[1] = inTexture2.sample(s, float2(pos.x, pos.y), slice);
    float4 out = float4(0.0f);
    out =float4( in[0]+in[1]);
    outTexture.write(half4(out), gid.xy, gid.z);
 }
```

MPSCNN VS NCNN on iPhone

FrameWork	Time
NCNN	110ms
MPSCNN	45ms

Device: iPhone 6s

How to create a new framework

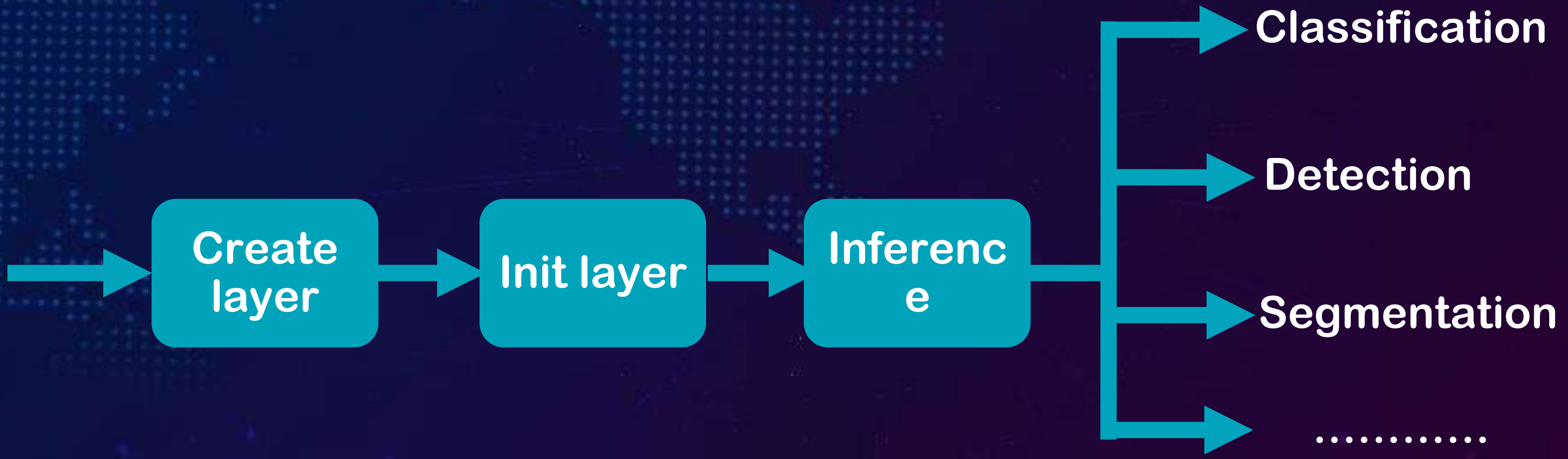
- 优化inference网络结构
- GPU加速
- 指令集加速
- 多线程
- 内存布局优化 NCHW—
- >NHWC
  浮点运算定点化

Mogu Deep Learning Toolkit

Mogu DL Toolkit

- For professional
- Highly Flexible
- High Cohesion & Low Coupling
- Easy to use

Mogu Deep Learning Toolkit



Mogu DL Toolkit-Example

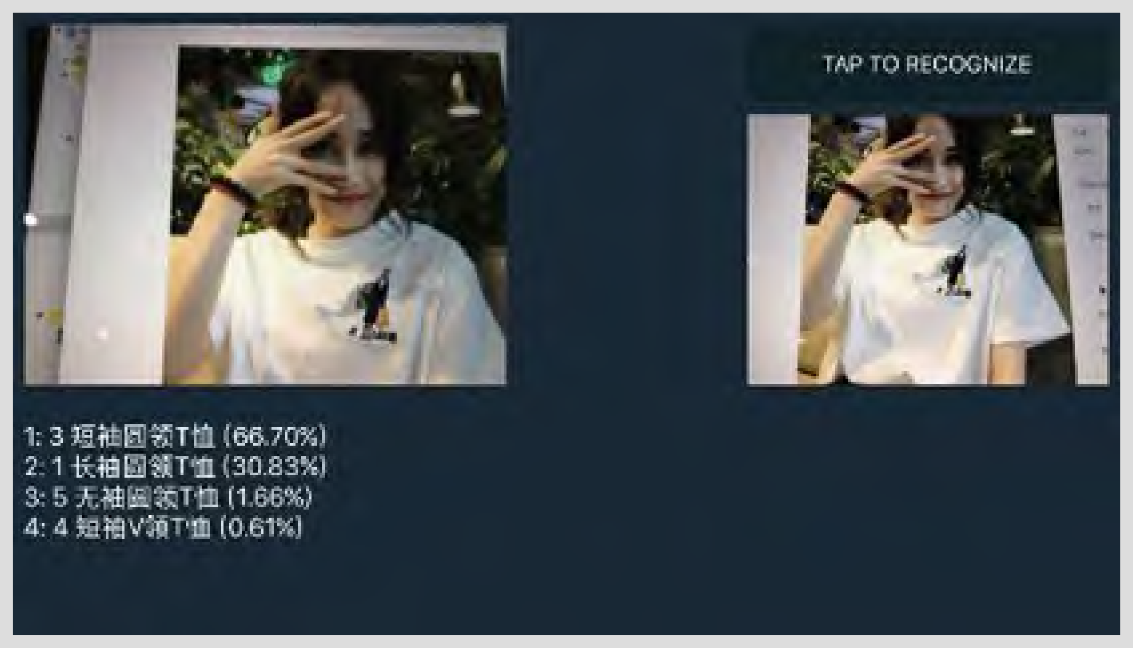
MobileNet

```
class MobileNet{
public:
    Input           input;
    Convolution        fc7;

    int Init(const char* modelpath);
    int infer(Mat &input,Mat &output);

private:
    Convolution        conv1_s2;
    ReLU              relu1;
    ConvolutionDepthWise conv2_1_dw;
    ReLU              relu2_1_dw;
    Convolution        conv2_1_s1;
    ReLU              relu2_1_s1;
    ConvolutionDepthWise conv2_2_dw;
    …….
}
```

Demo

WOTD 51CTO

Demo



总结

- 模型压缩的两类方式
- 移动端优化实践
- Mogu DL Toolkit
- 深度学习优化在蘑菇街业务中的尝试

致谢

- 感谢蘑菇街图像算法部门深度学习优化小组全体成员的共同努力！！

Thanks!

敬请关注
“蘑菇街技术博客”
公众号

敬请关注
“美丽联合数据技术”
公众号