# **介绍与概述**

MobileNets是为移动和嵌入式设备提出的高效模型。MobileNets基于流线型架构(streamlined)，使用**深度可分离卷积(depthwise separable convolutions,即Xception变体结构)来构建轻量级深度神经网络**。

MobileNet描述了一个高效的网络架构，允许通过两个超参数直接构建非常小、低延迟、易满足嵌入式设备要求的模型。

# 准备工作：训练构建小型网络的方法

## 直接训练小型模型

### 设置超参数

两个超参数的加入使得开发者非常容易构建符合特定应用环境的网络，主要是在延迟性和网络尺寸（计算量和参数量）上做出优化。

### 深度可分离卷积的应用

深度可分离卷积是本模型的核心。深度可分离卷积在Inception models里面有重要应用，这减少了最开始几层的计算量。扁平化网络（Flattened networks build a network out of fully factorized convolutions and showed the potential of extremely factorized networks）；Xception Network；Squeezenet使用bottleneck的方法来构建小型模型

## 压缩已有模型的方法构建小型模型

减小、分解或压缩预训练网络。例如量化压缩(product quantization)、哈希(hashing )、剪枝(pruning)、矢量编码( vector quantization)和霍夫曼编(Huffman coding)等；此外还有各种分解因子(various factorizations )用来加速预训练网络

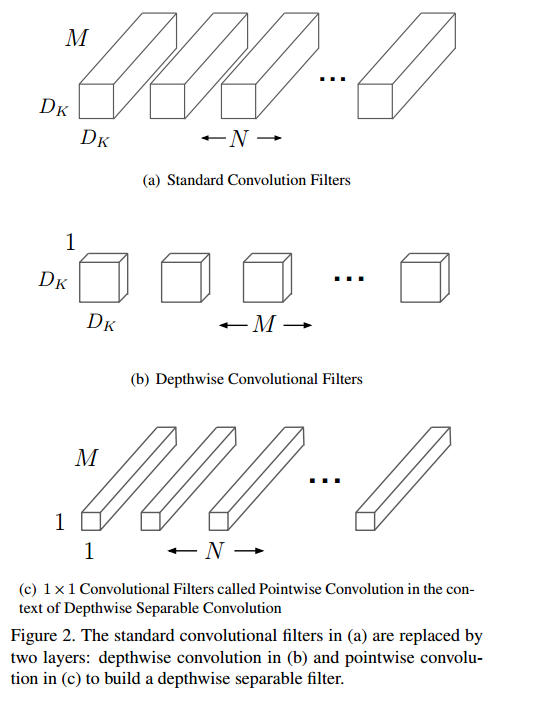
## 利用蒸馏（distillation）学习构建网络

用大型网络学到的知识训练小规模网络，具体涉及到softmax分类函数的温度常数的应用。

# MobileNet架构

## 深度可分离卷积的具体操作

将原本标准的卷积（standard convolution）操作因式分解成一个depthwise convolution和一个1\*1的pointwise convolution操作。简单讲就是将原来一个卷积层分成两个卷积层，其中前面一个卷积层的每个filter都只跟input的每个channel进行卷积，输出通道数量的特征图，后面一个卷积层则负责combine特征图，即将上一层卷积的结果进行合并。原文中说：The depthwise separable convolution splits this into two layers, a separate layer for filtering and a separate layer for combining.即深度可分离卷积看成：第一层进行滤波，第二层进行通道间的结合。卷积示意图如下：



相关变量注解如下：

DF：输出特征图的宽和高

M：输入的通道数

DG：输出特征图的宽和高

N：输出的通道数（output depth）

DK：卷积核的空间维度

K：conv kernal K

### 标准卷积的解释

Kernal K的计算参数量：*DK×DK×M×N*

标准卷积的计算公式为：

然后标准卷积的计算量为：

*DK · DK · M · N · DF · DF*

总的来说标准卷积是用卷积核过滤出特征再将不同的特征图结合来产生一种新的显示形式（representation）

### 分解卷积成两步减少计算量

深度卷积部分：imput的每个通道对应一个单通道卷积核（apply a single filter per each input channel (input depth)）。

每个输入通道下的计算公式与所有通道的计算量：

*DK · DK · M · DF · DF*

逐点卷积部分：使用1X1卷积核将深度卷积部分输出的不同通道线性结合，结合后的总计算量为：

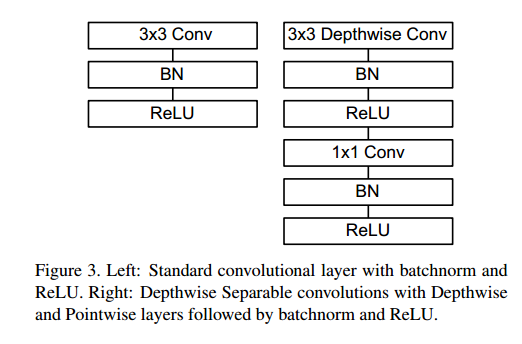
*DK · DK · M · DF · DF + M · N · DF · DF*

可以得到计算量减小的比例为：

## 网络的具体结构与训练过程

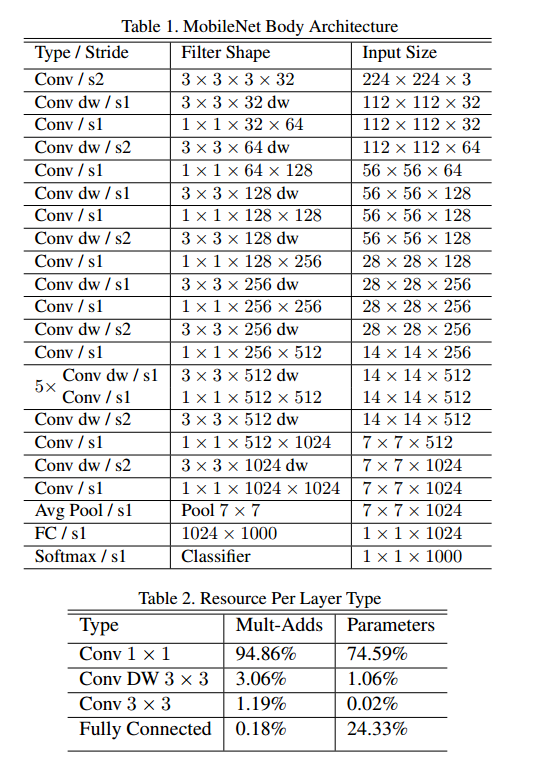
### 网络结构

第一层是普通标准卷积，其余层是深度卷积与逐点卷积交替连接。最后的平均池化使输出的空间分辨率降到1，然后连接全连接层。除全连接层外每层均使用batchnorm与Relu非线性激活函数，单元结构对比如下图。



下采样工作可以通过增加第一层和深度卷积层的strided参数完成。深度卷积与逐点卷积分别看作一层的话，模型总共有28层。我们需要在简化模型的同时确保模型能够有效率地工作

具体结构分布图如下：



### 训练过程

在TensorFlow中使用RMSprop对MobileNet做训练，使用类似InceptionV3 的异步梯度下降。与训练大型模型不同的是，我们较少使用正则和数据增强技术，因为小模型不易陷入过拟合；没有使用side heads or label smoothing，我们发现在深度卷积核上放入很少的L2正则或不设置权重衰减的很重要，因为这部分参数很少

## 超级参数的引入

### 宽度因子**α**（Width Multiplier: Thinner Models）

改变通道数：*αM，αN*

总计算量变为*：DK · DK · αM · DF · DF* + *αM · αN · DF · DF*

*α* 范围是 (0，1]，经常设置为1，0.75，0.5或0.25

注意：在生成新的网络时需要重新训练

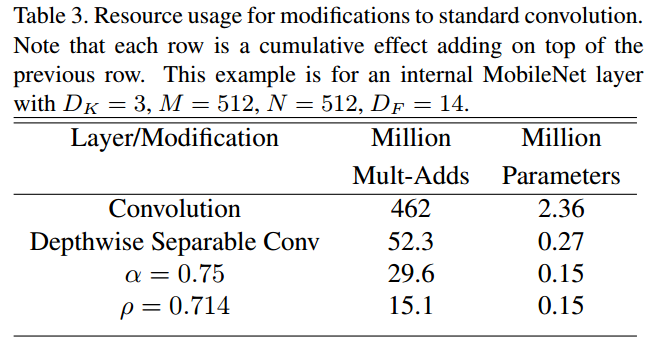
### 分辨率因子**ρ**（Resolution Multiplier: Reduced Representation）

通过设置*ρ*隐式地设置输入的分辨率：*ρDF*。

总计算量改变为：*DK · DK · αM · ρDF · ρDF + αM · αN · ρDF · ρDF*

*ρ*取值范围为(0，1]，分辨率常为224, 192, 160或128

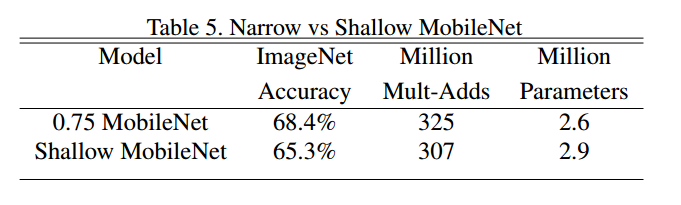
超参的作用示意图如下，可见超参因子可以很好地降低模型的复杂度：



# 实践对比实验

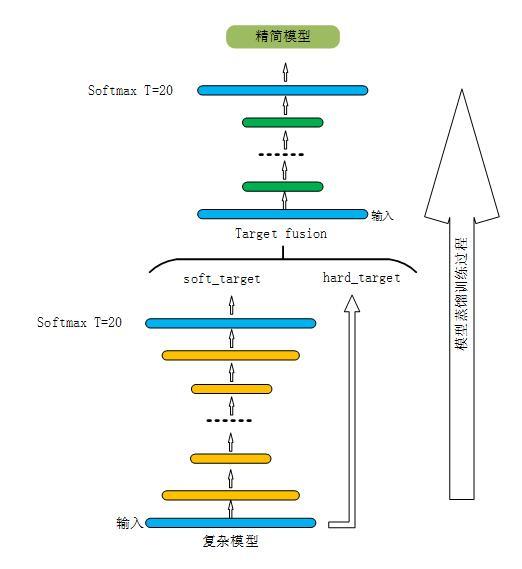
## 模型选择

直接减少模型层数（shallower models using less layers）与使用宽度因子（thinner models with width multiplier）缩小模型规模的方式对比。如图，ImageNet训练集上的测试结果，使用宽度因子简化模型得到的模型精度优于直接去掉模型层的做法。

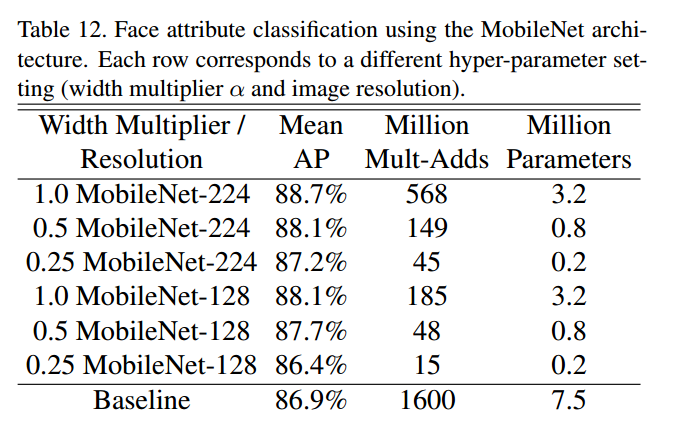


## **Face Attribute**

MobileNet 和 distillation（蒸馏）技术结合使用更佳（原文：distill a face attribute classifier using the MobileNet architecture）所谓模型蒸馏就是将训练好的复杂模型推广能力“知识”迁移到一个结构更为简单的网络中。或者通过简单的网络去学习复杂模型中“知识”。其可分为复杂模型的训练和精简模型的训练：

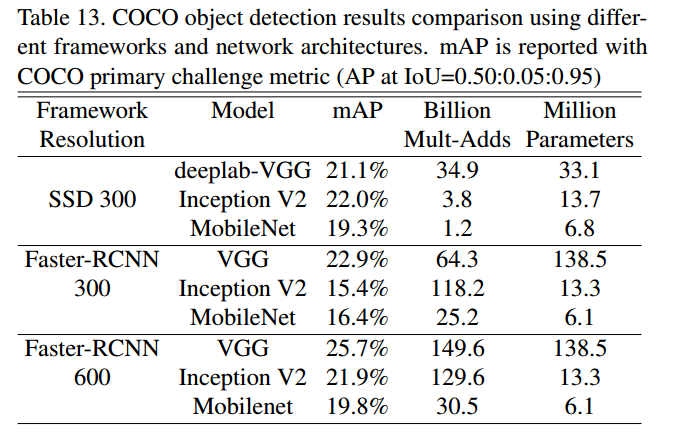


其中将large model的输出作为labels来训练mobilenet结构的分类器网络而不是直接用原始的数据集。其结果是基于mobilenet的分类器对于模型收缩的作用具有弹性（精度下降可以得到部分补偿）



## Object Detection

使用不同框架下MobileNet与VGG 以及 InceptionV2的比较



# Conclusion

论文提出了一种基于深度可分离卷积的新模型MobileNet，同时提出了两个超参数用于快速调节模型适配到特定环境。实验部分将MobileNet与许多先进模型做对比，展现出MobileNet的在尺寸、计算量、速度上的优越性。