MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications

Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam
Google Inc., 2017

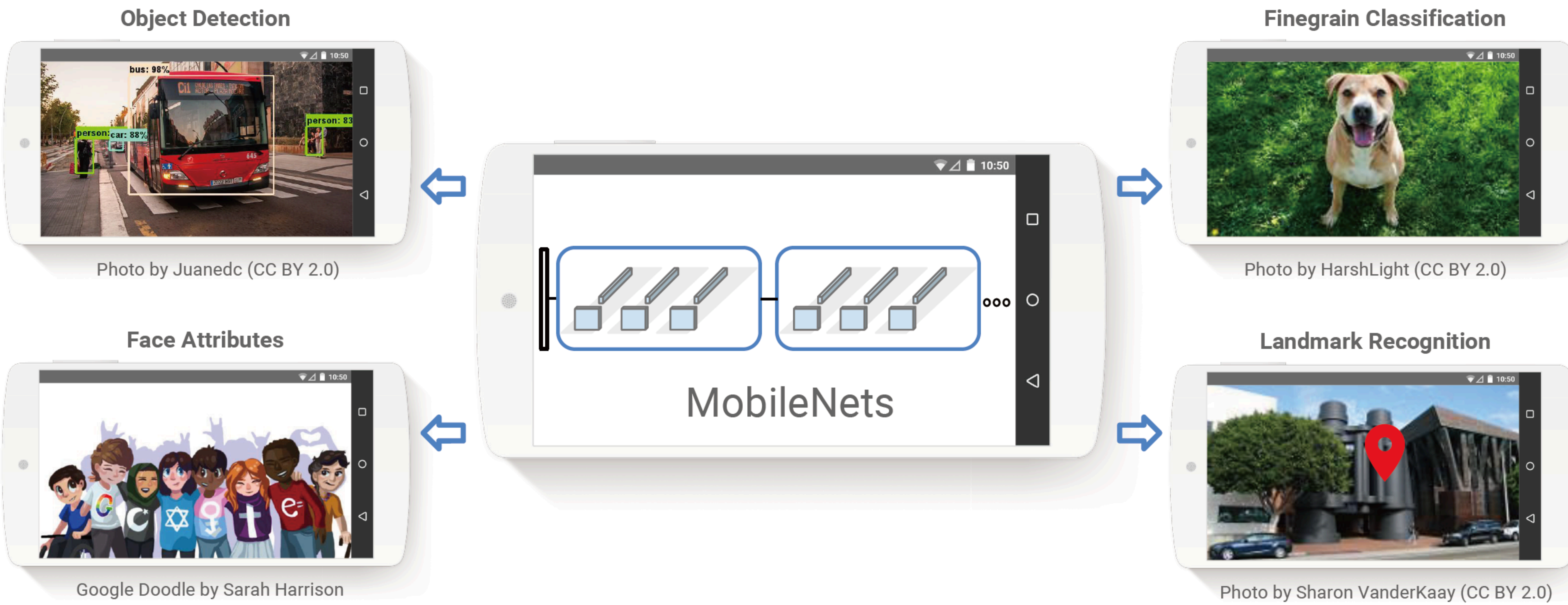
Introduction

- MobileNets is a class of efficient models for **mobile and embedded vision applications**

- Use depthwise separable convolutions to build light weight deep neural networks

- Add two simple global hyperparameters: width multiplier and resolution multiplier that efficiently trade off between latency and accuracy

Introduction



Prior Work

- Compressing pretrained networks

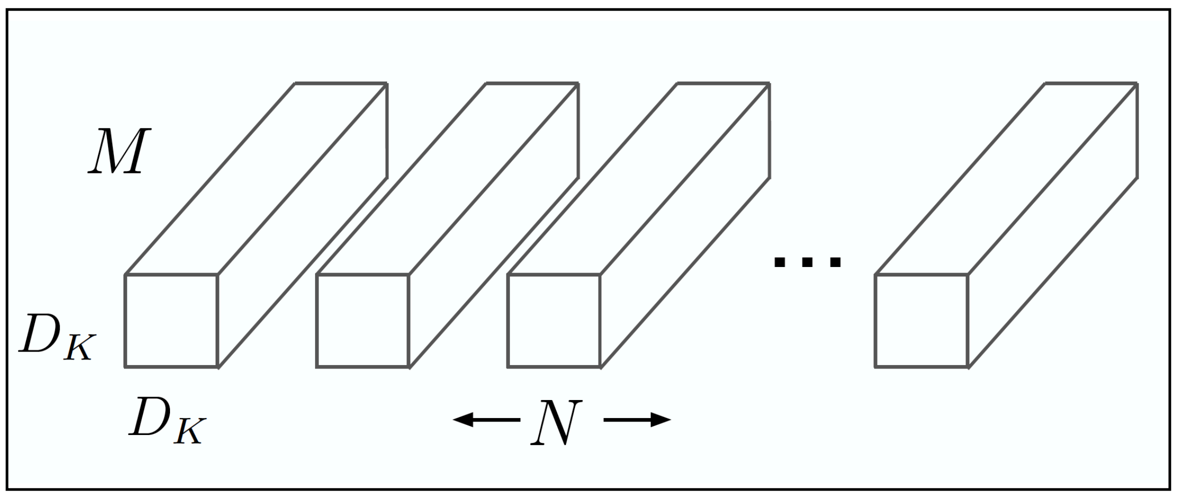
  —Product quantization, Hashing, Pruning, Vector quantization, Huffman coding

  —Factorization

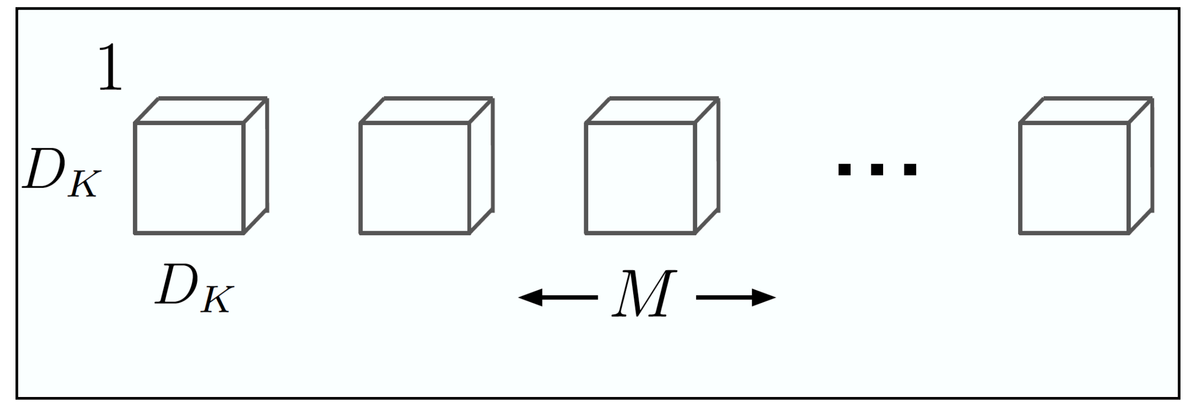
- Training small networks

  — Distillation

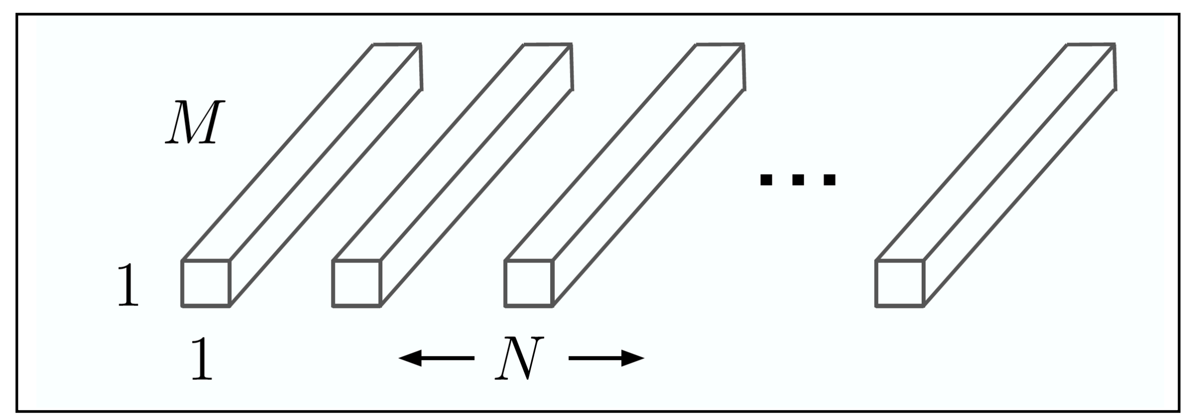
Depthwise Separable Convolution



Standard Convolution Filters



Depthwise Convolutional Filters



Pointwise Convolution

D_K: the spatial dimension of the kernel
M: the number of input channels (input depth)
N: the number of output channel (output depth)

->Reduce computation and model size

Depthwise Separable Convolution

- **Standard convolutional layer:**
  - Input: D_F x D_F x M feature map **F**
  - Output: D_F x D_F x N feature map **G**
  - Convolution kernel: D_K x D_K x M x N kernel **K**

$$\mathbf{G}_{k,l,n} = \sum_{i,j,m} \mathbf{K}_{i,j,m,n} \cdot \mathbf{F}_{k+i-1,l+j-1,m}$$

Cost: D_K x D_K x M x N x D_F x D_F

Depthwise Separable Convolution

- **Depthwise separable convolution:**
  - **Depthwise convolutions**

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \cdot \mathbf{F}_{k+i-1,l+j-1,m}$$

  - **Pointwise convolutions**

Cost: $D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F$

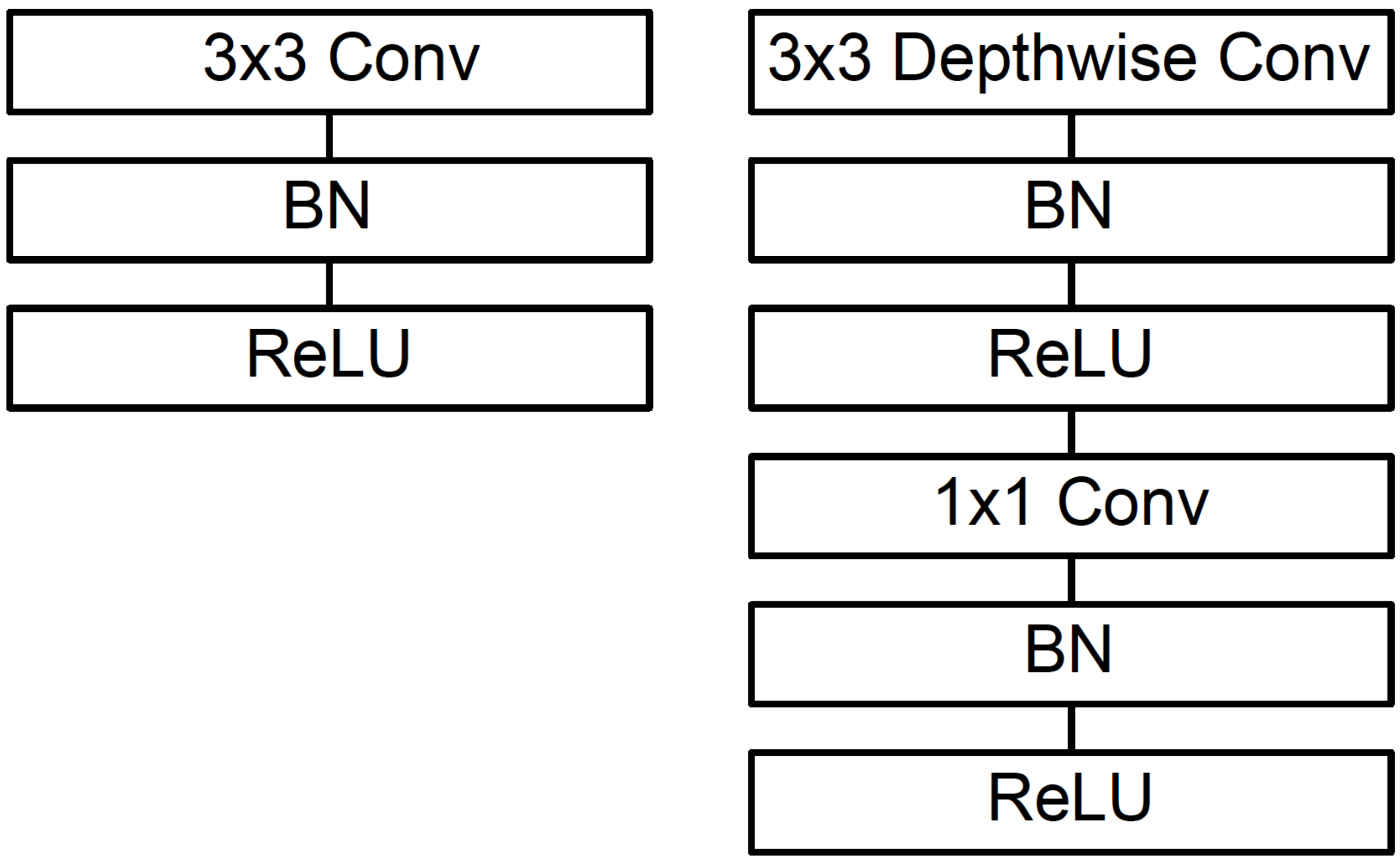
Depthwise Separable Convolution

- **Reduction**

$$
\begin{aligned}
& \frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F} \\
= \quad & \frac{1}{N} + \frac{1}{D_K^2}
\end{aligned}
$$

MobileNet structure

- **28 layers**



Width Multiplier: Thinner Models

- With width multiplier α, the cost become:

$$D_K \cdot D_K \cdot \alpha M \cdot D_F \cdot D_F + \alpha M \cdot \alpha N \cdot D_F \cdot D_F$$

- α∈(0,1] with typical settings of 1, 0.75, 0.5 and 0.25

Resolution Multiplier: Reduced Representation

- With width multiplier ρ, the cost become:

$$D_K \cdot D_K \cdot \alpha M \cdot \rho D_F \cdot \rho D_F + \alpha M \cdot \alpha N \cdot \rho D_F \cdot \rho D_F$$

- $\rho \in (0,1]$ with typical settings ρ to make input resolution of the network be 224, 192, 160 or 128

Depthwise Separable Convolutions v.s. Full Convolutions

Table 4. Depthwise Separable vs Full Convolution MobileNet

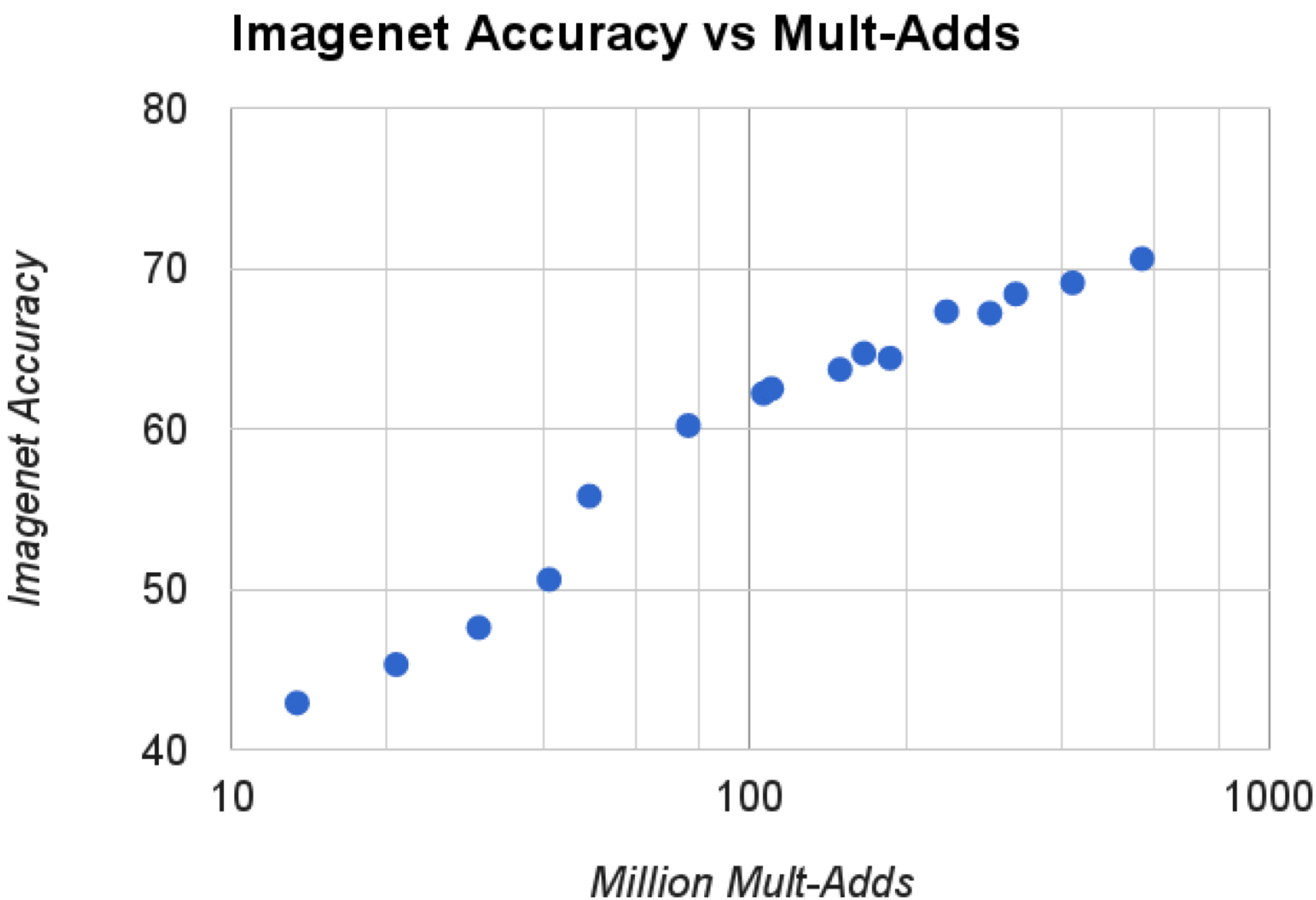
Model	ImageNet Accuracy	Million Mult-Adds	Million Parameters
Conv MobileNet	71.7%	4866	29.3
MobileNet	70.6%	569	4.2

Width Multiplier v.s. Shallow Model

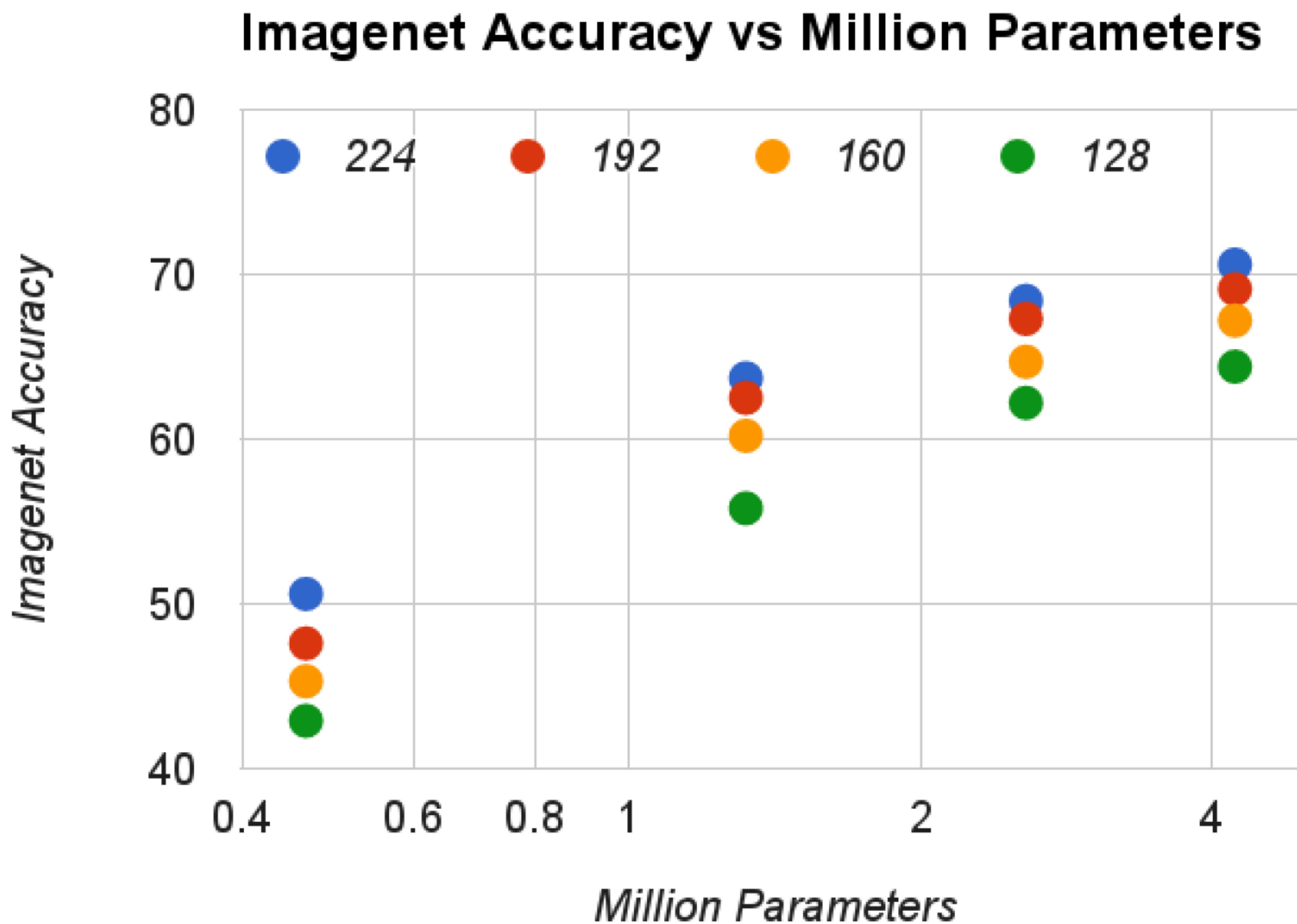
Table 5. Narrow vs Shallow MobileNet

Model	ImageNet Accuracy	Million Mult-Adds	Million Parameters
0.75 MobileNet	68.4%	325	2.6
Shallow MobileNet	65.3%	307	2.9

Trade off between computation (Mult-Adds) and Accuracy



Trade off between Number of and Accuracy



Conclusion

- Propose a new model architecture called MobileNets
- Two Features:
  —Use depthwise separable convolutions
  —Use width multiplier and resolution multiplier