EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks - Paper, Code (TF)

By Google Brain: May 2019 Summary Notes by: Ashvini Jindal, LinkedIn

- **Contribution**:
  - Scaling ConvNets across each of depth, width, and resolution based on novel **Compound Scaling** technique. Existing approaches generally scale along one or more dimensions in non-systematic but not all three.
  - Proposed model is 8x smaller and 6x faster on inference, achieving the same performance as the SOTA GPipe Model.
- **Advantage**: Much smaller models with faster inference and better performance (accuracy).

- Based on NAS, obtain a family of models called EfficientNets.
- EfficientNet variants: EfficientNet-B0 (Smallest model) to EfficientNet-B7 (Largest model)
- How to find EfficientNet-B0:
  - First fix ϕ = 1 and find find α, β, and γ for EfficientNet-B0 architecture using grid search with
- Obtain EfficientNet-B1 to EfficientNet-B7:
  - Change only ϕ value with (above) fixed α, β, and γ parameters. Larger ϕ value corresponds to a model with more parameters leading to various EfficientNet variants: see equation (2) below.
- Intuitively, ϕ acts as user specified coefficient that controls how many resources are available (see equation (3) and (2) below)

- EfficientNet-B0 is simply a scaled-ConvNet where model architecture is **similar** to **MnasNET**.
- Biggest model (EfficientNet-B7) has **66M** parameters compared to the SOTA model (GPipe) with **560M** parameters without any performance dip on ImageNet dataset.
- Model transfers well on other 8 transfer learning classification datasets and achieves SOTA for 5 out of 8.

$$\begin{aligned} \max_{d,w,r} \quad & Accuracy\big(\mathcal{N}(d,w,r)\big) \\ s.t. \quad & \mathcal{N}(d,w,r) = \bigodot_{i=1\ldots s} \hat{\mathcal{F}}_i^{d\cdot\hat{L}_i}\big(X_{\langle r\cdot\hat{H}_i, r\cdot\hat{W}_i, w\cdot\hat{C}_i\rangle}\big) \\ & \text{Memory}(\mathcal{N}) \leq \text{target\_memory} \\ & \text{FLOPS}(\mathcal{N}) \leq \text{target\_flops} \end{aligned} \tag{2}$$

In this paper, we propose a new **compound scaling method**, which use a compound coefficient ϕ to uniformly scales network width, depth, and resolution in a principled way:

$$\begin{aligned} \text{depth: } & d = \alpha^\phi \\ \text{width: } & w = \beta^\phi \\ \text{resolution: } & r = \gamma^\phi \\ \text{s.t. } & \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\ & \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned} \tag{3}$$

Table 5. **EfficientNet Performance Results on Transfer Learning Datasets**. Our scaled EfficientNet models achieve new state-of-the-art accuracy for 5 out of 8 datasets, with 9.6x fewer parameters on average.

	Comparison to best public-available results						Comparison to best reported results					
	Model	Acc.	#Param	Our Model	Acc.	#Param(ratio)	Model	Acc.	#Param	Our Model	Acc.	#Param(ratio)
CIFAR-10	NASNet-A	98.0%	85M	EfficientNet-B0	98.1%	4M (21x)	†Gpipe	**99.0%**	556M	EfficientNet-B7	98.9%	64M (8.7x)
CIFAR-100	NASNet-A	87.5%	85M	EfficientNet-B0	88.1%	4M (21x)	Gpipe	91.3%	556M	EfficientNet-B7	**91.7%**	64M (8.7x)
Birdsnap	Inception-v4	81.8%	41M	EfficientNet-B5	82.0%	28M (1.5x)	GPipe	83.6%	556M	EfficientNet-B7	**84.3%**	64M (8.7x)
Stanford Cars	Inception-v4	93.4%	41M	EfficientNet-B3	93.6%	10M (4.1x)	‡DAT	**94.8%**	-	EfficientNet-B7	94.7%	-
Flowers	Inception-v4	98.5%	41M	EfficientNet-B5	98.5%	28M (1.5x)	DAT	97.7%	-	EfficientNet-B7	**98.8%**	-
FGVC Aircraft	Inception-v4	90.9%	41M	EfficientNet-B3	90.7%	10M (4.1x)	DAT	92.9%	-	EfficientNet-B7	**92.9%**	-
Oxford-IIIT Pets	ResNet-152	94.5%	58M	EfficientNet-B4	94.8%	17M (5.6x)	GPipe	**95.9%**	556M	EfficientNet-B6	95.4%	41M (14x)
Food-101	Inception-v4	90.8%	41M	EfficientNet-B4	91.5%	17M (2.4x)	GPipe	93.0%	556M	EfficientNet-B7	**93.0%**	64M (8.7x)
Geo-Mean						**(4.7x)**						**(9.6x)**

†GPipe (Huang et al., 2018) trains giant models with specialized pipeline parallelism library.
‡DAT denotes domain adaptive transfer learning (Ngiam et al., 2018). Here we only compare ImageNet-based transfer learning results.
Transfer accuracy and #params for NASNet (Zoph et al., 2018), Inception-v4 (Szegedy et al., 2017), ResNet-152 (He et al., 2016) are from (Kornblith et al., 2019).