Cornell Bowers CIS Computer Science

EfficientNet: Rethinking Model Scaling for ConvNets

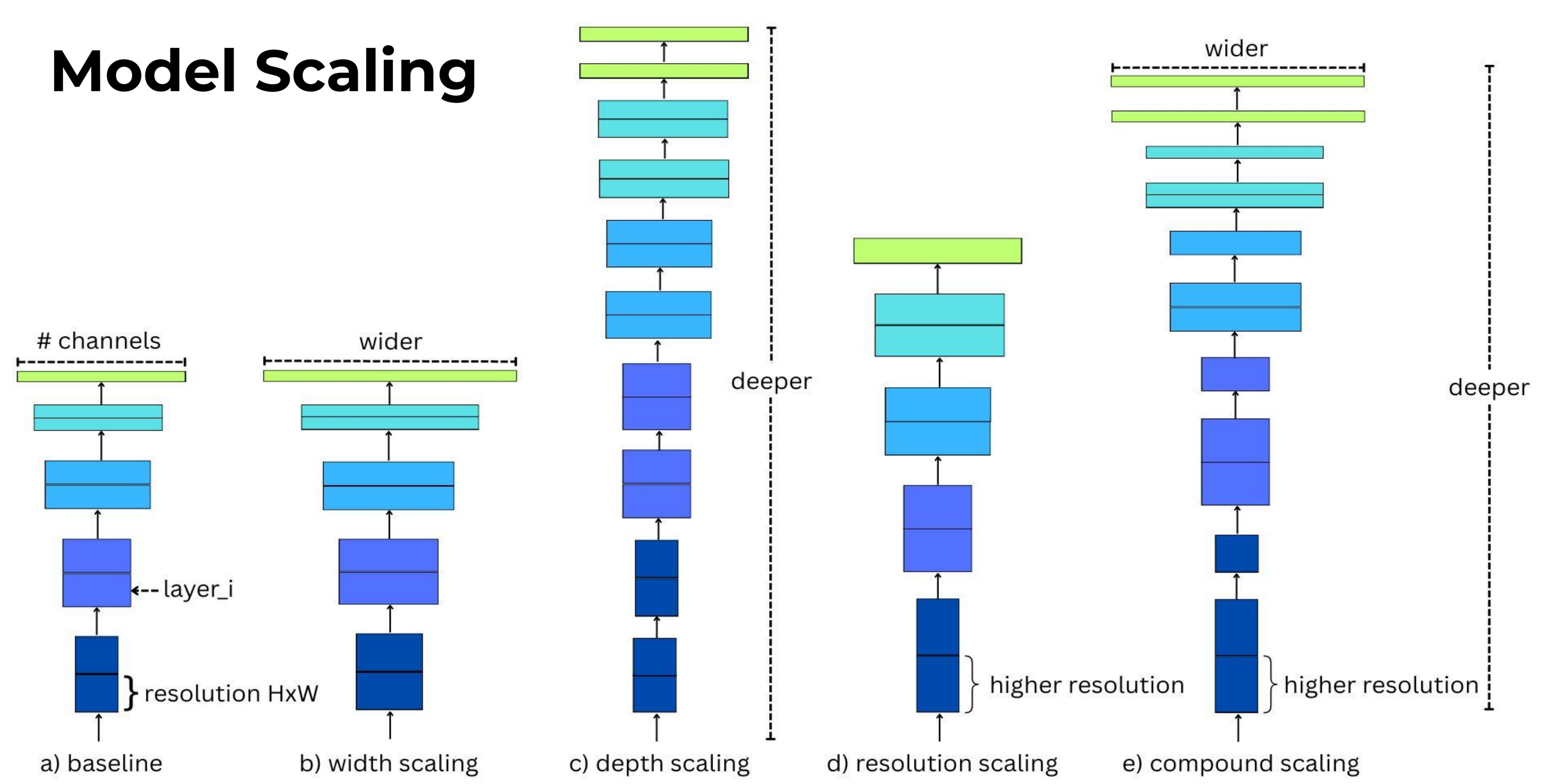
Aaron Baruch (amb565) Mohammad Labadi (maa366) Katerina Popova (kap272) Justin Tien-Smith (jzt5) Peter Zheng (pcz3)

Original Paper

Problem

- **Single-dimension scaling is inefficient**: Scaling depth, width, or input resolution alone yields diminishing accuracy gains as you make models larger.
- **No principled way to balance all three dimensions:** Manually tuning combinations of layers, channels, and image size is tedious and sub-optimal.
- **Huge search space for ConvNet design:** Exploring every mix of depth/width/resolution for a given compute budget is prohibitively expensive.

Model Scaling



Paper's Solution

- **Introduce a new baseline (EfficientNet-B0):** Used neural-architecture search to find an efficient MobileNet-style backbone with MBConv+SE blocks.
- **Compound scaling rule:** Uniformly scale depth, width, and resolution by:

$$d = \alpha^{\phi},\ w = \beta^{\phi},\quad r = \gamma^{\phi}$$

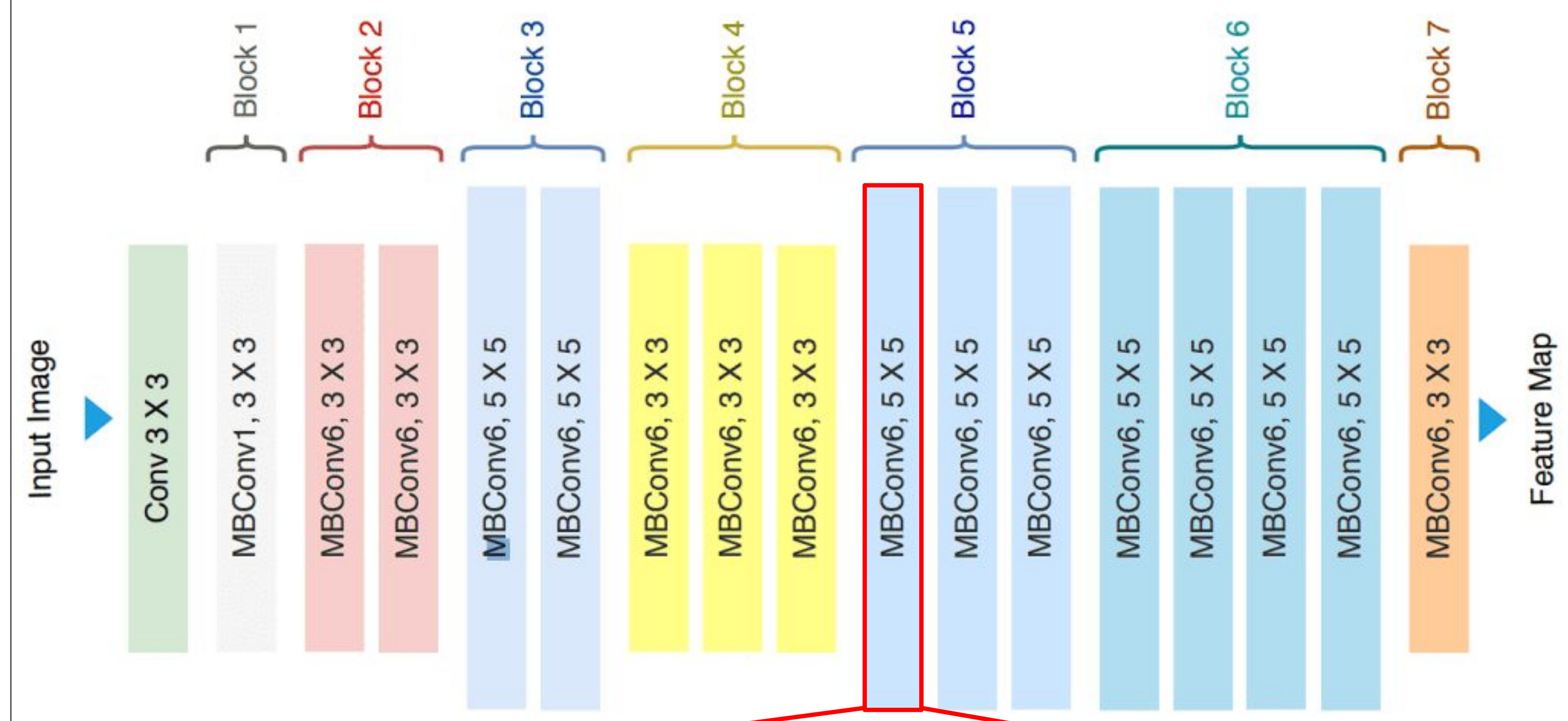
with coefficients $\alpha = 1.2$, $\beta = 1.1$, $\gamma = 1.15$ chosen via a small grid search.

- **Generate a model family (B0–B7):** Vary the compound coefficient ϕ to smoothly trade off compute vs. accuracy.
- **Dramatic efficiency gains:** Achieve up to 8× fewer parameters and 6× faster inference for the same (or better) ImageNet accuracy!

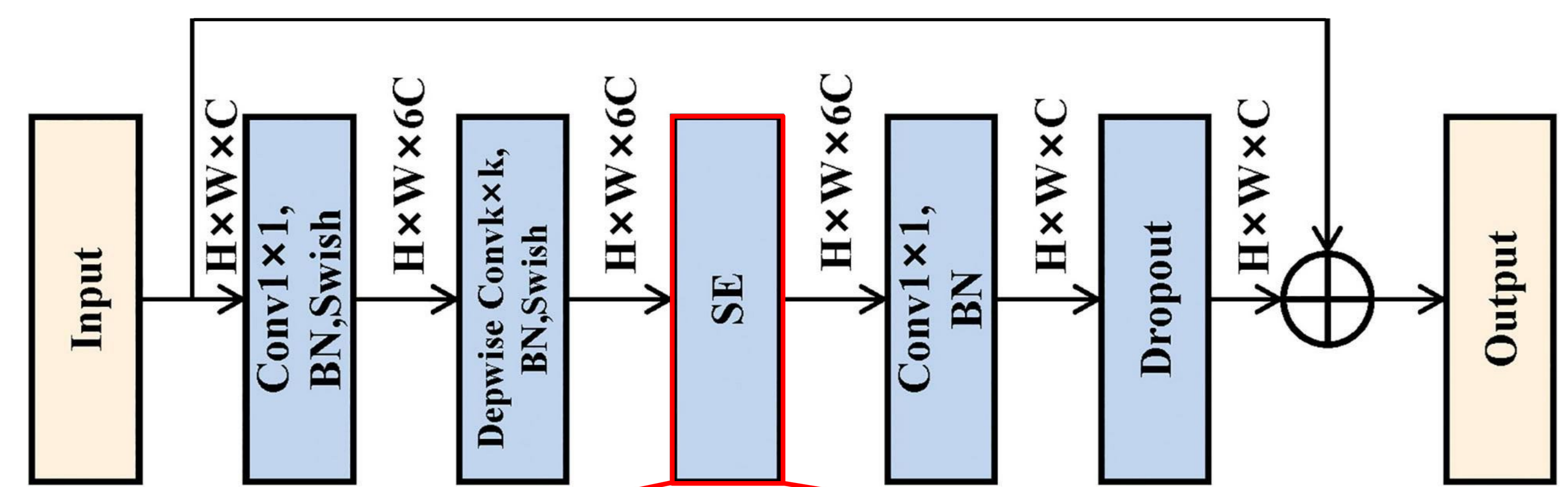
References

- Tan, M., Le, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. arXiv:1905.11946v5, 2020.
- iamtapendu. (2025). Introduction to EfficientNet [Kaggle Notebook]. Kaggle. Retrieved May 5, 2025, from https://www.kaggle.com/code/iamtapendu/introduction-to-efficientnet
- Klingler, N. (2024, March 18). EfficientNet: Optimizing Deep Learning Efficiency. Viso.ai. Retrieved May 5, 2025, from https://viso.ai/deep-learning/efficientnet/

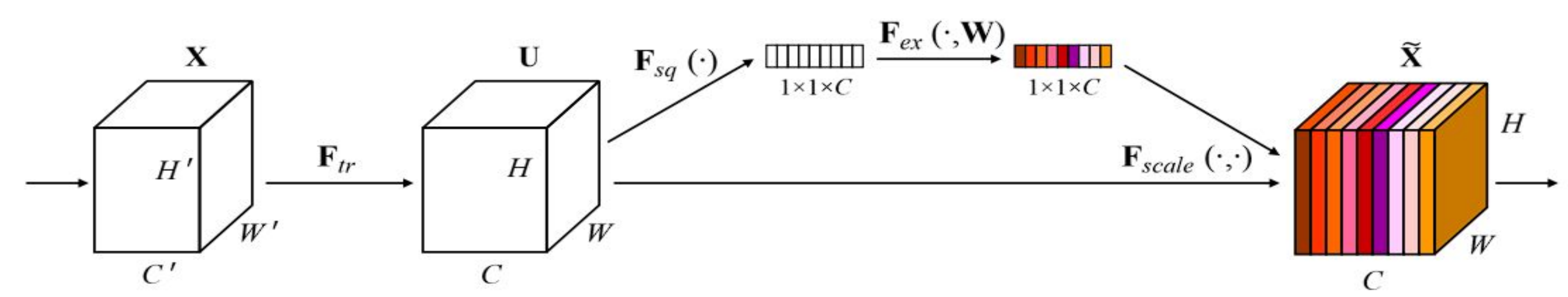
EfficientNet-B0 Model Architecture

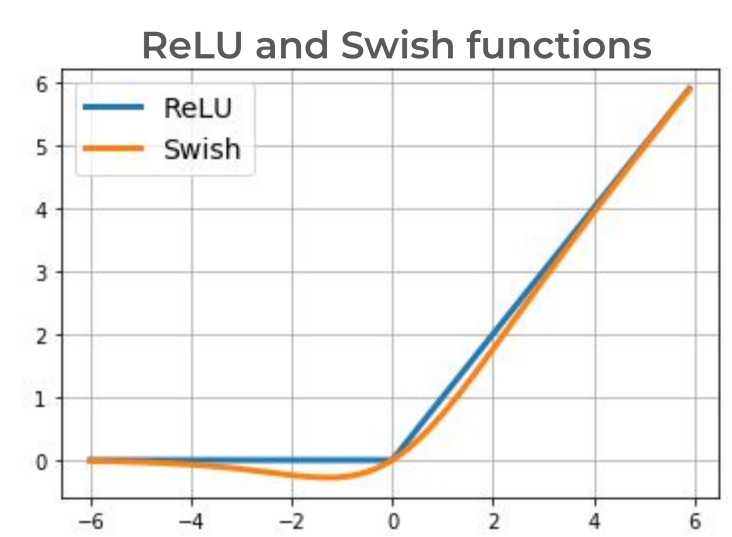


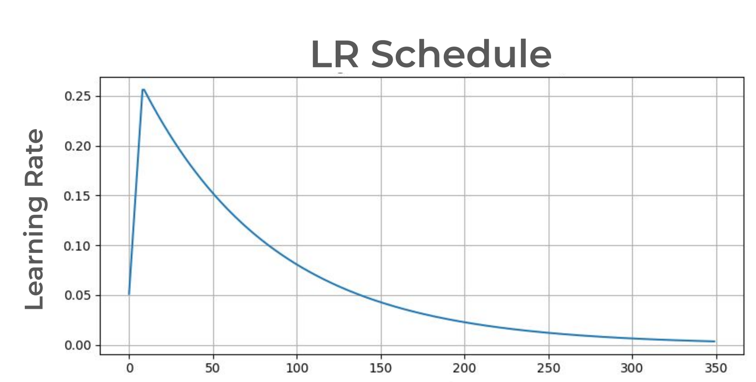
Main Building Block of EfficientNet: Mobile Inverted Bottleneck (MBConv)



Squeeze-and-Excitation



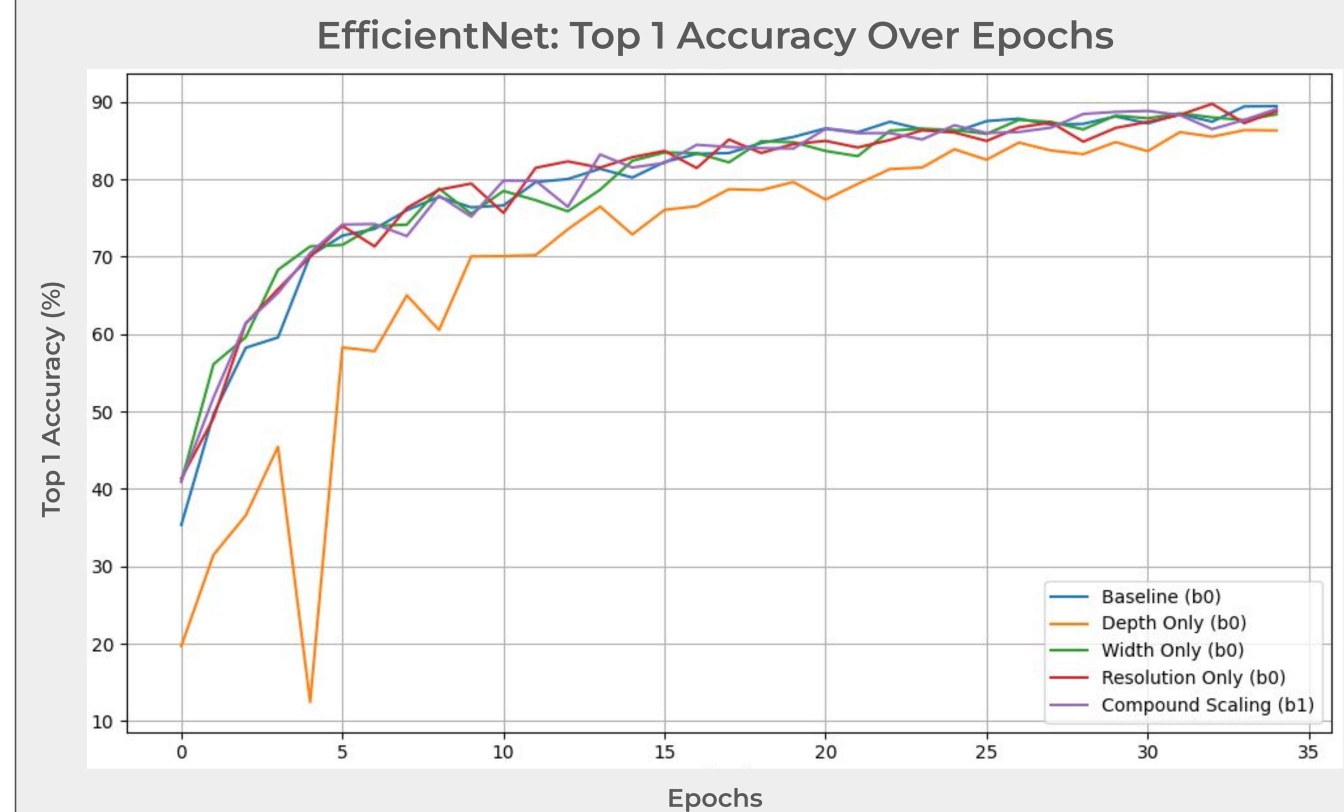




Our Methodology

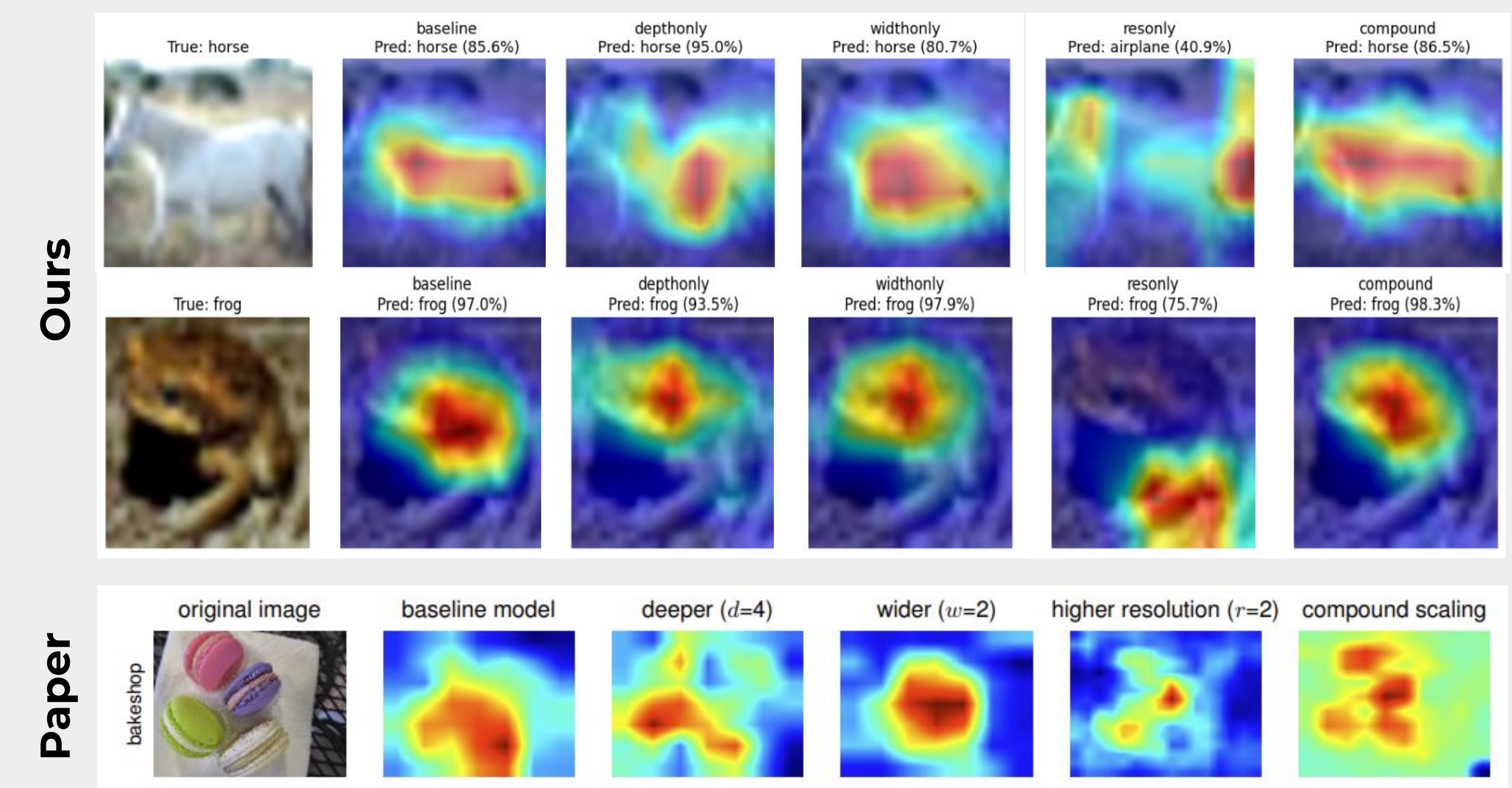
- **Dataset**: CIFAR-10 (60K images, 10 classes).
- **Models**: B0 (baseline), depth-only, width-only, resolution-only, B1 (compound).
- **Optimizer**: *RMSProp* (momentum = 0.9, α = 0.9, ε = 0.1) with *linear → exponential learning rate decay.*
- **Training**: *35 epochs* (vs. 350 in paper).

Results



Over 35 epochs, one compound step (B0 → B1) gave **negligible improvement** over single dimension scaling; larger steps (more compounding) hit **OOM beyond B1.**

Class Activation Maps



Model	Our FLOPS	Our Top 1 Acc	Paper FLOPS	Paper Top 1
Baseline B0 model	0.82B	89.45%	0.4B	77.3%
Scale by depth (B0)	3.35B	86.31%	1.8B	79.0%
Scale by width (B0)	3.02B	88.40%	1.8B	78.9%
Scale by resolution (B0)	3.27B	88.80%	1.9B	79.1%
Compound scale (B1)	1.82B	85.21%	1.8B	81.1%

Table 1. Scaling effect on FLOPS and Top-1 Accuracy

Conclusion/Future Work:

- **Conclusion:** Our findings did not directly align with the paper's key insight. Compound scaling did not seem to deliver noticeable gains than scaling any single dimension alone.
- **Future Work:**
  - Extend experiments across the full EfficientNet family (B0–B7)
  - Match the original training schedule (350 epochs) for fair comparison
  - Scale up to and evaluate on the full ImageNet dataset instead of just CIFAR-10