

# ConvNets Match Vision Transformers at Scale

Samuel L Smith<sup>1</sup>, Andrew Brock<sup>1</sup>, Leonard Berrada<sup>1</sup> and Soham De<sup>1</sup>

<sup>1</sup>Google DeepMind

Many researchers believe that ConvNets perform well on small or moderately sized datasets, but are not competitive with Vision Transformers when given access to datasets on the web-scale. We challenge this belief by evaluating a performant ConvNet architecture pre-trained on JFT-4B, a large labelled dataset of images often used for training foundation models. We consider pre-training compute budgets between 0.4k and 110k TPU-v4 core compute hours, and train a series of networks of increasing depth and width from the NFNet model family. We observe a log-log scaling law between held out loss and compute budget. After fine-tuning on ImageNet, NFNets match the reported performance of Vision Transformers with comparable compute budgets. Our strongest fine-tuned model achieves a Top-1 accuracy of 90.4%.

*Keywords: ConvNets, CNN, Convolution, Transformer, Vision, ViTs, NFNets, JFT, Scaling, Image*

## Introduction

Convolutional Neural Networks (ConvNets) were responsible for many of the early successes of deep learning. Deep ConvNets were first deployed commercially over 20 years ago (LeCun et al., 1998), while the success of AlexNet on the ImageNet challenge in 2012 re-ignited widespread interest in the field (Krizhevsky et al., 2017). For almost a decade ConvNets (typically ResNets (He et al., 2016a,b)) dominated computer vision benchmarks. However in recent years they have increasingly been replaced by Vision Transformers (ViTs) (Dosovitskiy et al., 2020).

Simultaneously, the computer vision community has shifted from primarily evaluating the performance of randomly initialized networks on specific datasets like ImageNet, to evaluating the performance of networks pre-trained on large general purpose datasets collected from the web. This raises an important question; do Vision Transformers outperform ConvNet architectures pre-trained with similar computational budgets?

Although most researchers in the community believe Vision Transformers show better scaling properties than ConvNets, there is surprisingly little evidence to support this claim. Many papers studying ViTs compare to weak ConvNet baselines (typically the original ResNet architecture (He et al., 2016a)). Additionally, the strongest ViT models have been pre-trained using large com-

pute budgets beyond 500k TPU-v3 core hours (Zhai et al., 2022), which significantly exceeds the compute used to pre-train ConvNets.

We evaluate the scaling properties of the NFNet model family (Brock et al., 2021), a pure convolutional architecture published concurrently with the first ViT papers, and the last ConvNet to set a new SOTA on ImageNet. We do not make any changes to the model architecture or the training procedure (beyond tuning simple hyper-parameters such as the learning rate or epoch budget). We consider compute budgets up to a maximum of 110k TPU-v4 core hours,<sup>1</sup> and pre-train on the JFT-4B dataset which contains roughly 4 billion labelled images from 30k classes (Sun et al., 2017). We observe a log-log scaling law between validation loss and the compute budget used to pre-train the model. After fine-tuning on ImageNet, our networks match the performance of pre-trained ViTs with comparable compute budgets (Alabdulmohsin et al., 2023; Zhai et al., 2022), as shown in Figure 1.

## Pre-trained NFNets obey scaling laws

We train a range of NFNet models of varying depth and width on JFT-4B. Each model is trained for a range of epoch budgets between 0.25 and 8, using a cosine decay learning rate schedule.

<sup>1</sup>TPU-v4 cores have roughly double the theoretical flops of TPU-v3 cores, however both cores have similar memory.

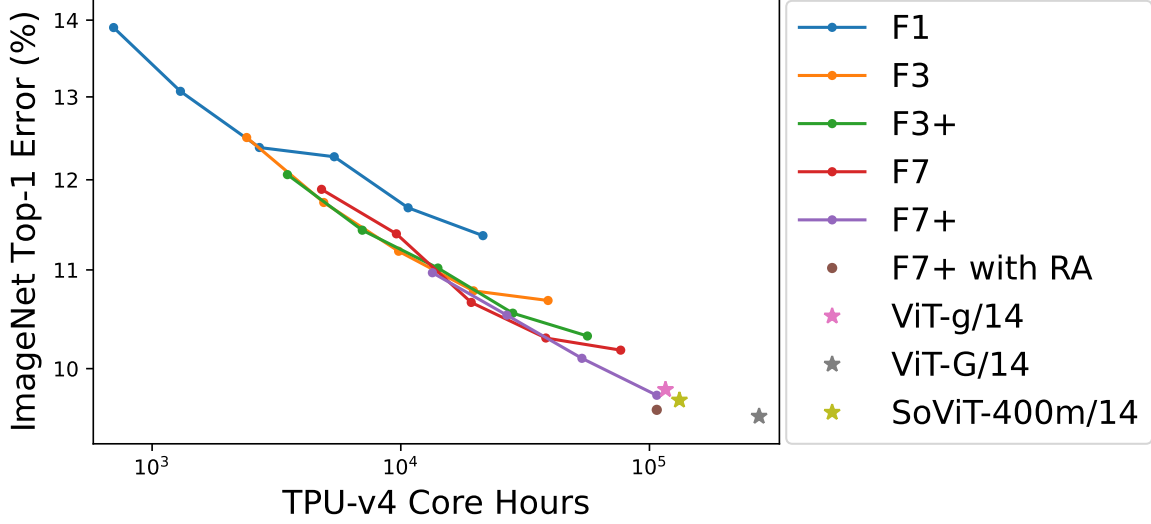


Figure 1 | ImageNet Top-1 error, after fine-tuning pre-trained NFNet models for 50 epochs. Both axes are log-scaled. Performance improves consistently as the compute used during pre-training increases. Our largest model (F7+) achieves comparable performance to that reported for pre-trained ViTs with a similar compute budget (Alabdulmohsin et al., 2023; Zhai et al., 2022). The performance of this model improved further when fine-tuned with repeated augmentation (RA) (Hoffer et al., 2019).

The base learning rate is tuned separately for each epoch budget on a small logarithmic grid. In Figure 2, we provide the validation loss at the end of training on a held out set of 130k images, plotted against the compute budget required to train each model<sup>2</sup>. We note that F7 has the same width as F3, but is double the depth. Similarly F3 is double the depth of F1, and F1 is double the depth of F0. F3+ and F7+ have the same depths as F3 and F7 but larger width. We train using SGD with Momentum and Adaptive Gradient Clipping (AGC) at batch size 4096, and we use an image resolution of  $224 \times 224$  during training and  $256 \times 256$  at evaluation. For additional details describing the NFNet architecture and training pipeline we refer the reader to the original paper (Brock et al., 2021), including the pre-training framework for JFT described in Section 6.2. Note that we removed near-duplicates of images in the training and validation sets of ImageNet from JFT-4B before training (Kolesnikov et al., 2020).

Figure 2 shows a clear linear trend, consistent with a log-log scaling law between validation loss and pre-training compute. This matches the log-log scaling laws previously observed when per-

forming language modelling with transformers (Brown et al., 2020; Hoffmann et al., 2022).

The optimal model size and the optimal epoch budget (which achieve the lowest validation loss) both increase in size as the compute budget increases. We found that a reliable rule of thumb is to scale the model size and the number of training epochs at the same rate, as previously observed for language modelling by Hoffmann et al. (2022). We note that the optimal epoch budget was greater than 1 for overall compute budgets greater than roughly 5k TPU-v4 core hours.

In Figure 3 we plot the observed optimal learning rate (which minimizes validation loss), for 3 of our models, across a range of epoch budgets.<sup>3</sup> Note that we tuned the learning rate on a logarithmic grid spaced by factors of 2. We find that all models in the NFNet family show a similar optimal learning rate  $\alpha \approx 1.6$  for small epoch budgets. However the optimal learning rate falls as the epoch budget rises, and for large models the optimal learning rate falls more quickly. In practice one can efficiently tune the learning rate within 2 trials by assuming that the optimal learning rate falls slowly but monotonically as both the model size and the epoch budget increases.

<sup>2</sup>We estimate the compute required to train each model by eye from the typical steps per second achieved by each model during training (when not pre-empted).

<sup>3</sup>The optimal learning rate showed very similar trends for all models. We select 3 models here for visual clarity.

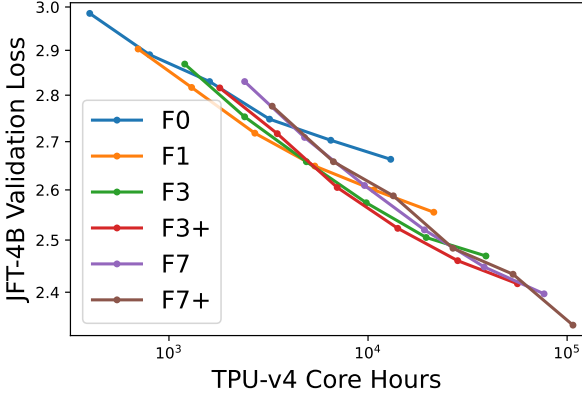


Figure 2 | Held out loss of NFNets on JFT-4B, plotted against the compute used during training. Both axes are log-scaled, and each curve denotes a different model trained for a range of epoch budgets. We observe a linear trend, matching the scaling laws observed for language modelling.

Finally, we note that some pre-trained models in Figure 2 perform less well than expected. For example, the curve for NFNet-F7+ models at different pre-training budgets is not smooth. We believe this arises because our data loading pipeline did not guarantee that each training example would be sampled once per epoch if the training run was pre-empted/restarted, potentially causing some training examples to be under-sampled if a training run was restarted multiple times.

## Fine-tuned NFNets are competitive with Vision Transformers on ImageNet

In Figure 1, we fine-tune our pre-trained NFNets on ImageNet, and plot the Top-1 error against the compute used during pre-training. We fine-tune each model for 50 epochs using sharpness aware minimization (SAM) (Foret et al., 2020) with stochastic depth and dropout. We train at resolution  $384 \times 384$  and evaluate at  $480 \times 480$ .

The ImageNet Top-1 accuracy consistently improves as the compute budget increases. Our most expensive pre-trained model, an NFNet-F7+ pre-trained for 8 epochs, achieves an ImageNet Top-1 accuracy of 90.3% while requiring roughly 110k TPU-v4 core hours to pre-train and 1.6k TPU-v4 core hours to fine-tune. Furthermore, we achieve 90.4% Top-1 accuracy if we additionally introduce repeated augmentation during fine-tuning (Fort et al., 2021; Hoffer et al., 2019) with aug-

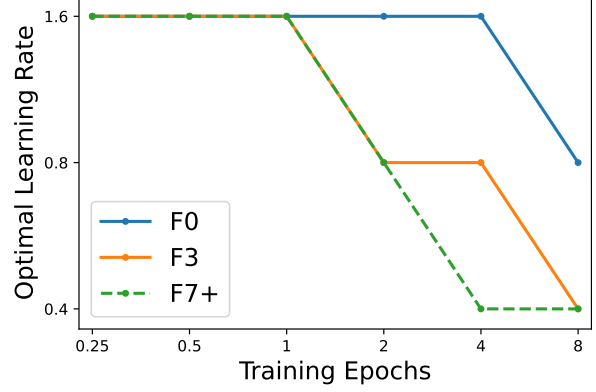


Figure 3 | The optimal learning rate behaves predictably and is easy to tune. All models show similar optimal learning rates  $\alpha \sim 1.6$  when the epoch budget is small. The learning rate falls slowly as model size and epoch budget increases.

mentation multiplicity 4.<sup>4</sup> For comparison, the best reported Top-1 accuracy of an NFNet on ImageNet without extra data is 86.8% (Fort et al., 2021), achieved by an NFNet-F5 with repeated augmentation. This demonstrates that NFNets benefit substantially from large scale pre-training.

Despite the substantial differences between the two model architectures, the performance of pre-trained NFNets at scale is remarkably similar to the performance of pre-trained Vision Transformers. For example, Zhai et al. (2022) achieve 90.2% Top-1 on ImageNet with a ViT-g/14, after pre-training on JFT-3B for 210k TPU-v3 core hours, and 90.45% with a ViT-G/14 after pre-training on JFT-3B for over 500k TPU-v3 core hours. In a recent work, Alabdulmohsin et al. (2023) optimize the ViT architecture and achieve 90.3% Top-1 with a SoViT-400m/14 after pre-training on JFT-3B for 230k TPU-v3 hours.

We evaluated the pre-training speed for these models on TPU-v4 (using the original authors’ codebase), and estimate that ViT-g/14 would take 120k TPU-v4 core hours to pre-train, while ViT-G/14 would take 280k TPU-v4 core hours and SoViT-400m/14 would take 130k TPU-v4 core hours. We use these estimates to compare the pre-training efficiency of ViTs and NFNets in Figure 1. We note however that NFNets were optimized for TPU-v4, and perform less well when evaluated

<sup>4</sup>When using repeated augmentation, we reduce the number of passes through the data such that the total computational cost of fine-tuning is constant.

on other devices. For example, we estimate that NFNet-F7+ would require 250 TPU-v3 core hours to pre-train for 8 epochs in our codebase.

Finally, we note that the pre-trained checkpoints achieving the lowest validation loss on JFT-4B did not always achieve the highest Top-1 accuracy on ImageNet after fine-tuning. In particular, we found that, under a fixed pre-training compute budget, the fine-tuning regime consistently favoured slightly larger models and slightly smaller epoch budgets. Intuitively, larger models have more capacity and are therefore better able to adapt to the new task. In some cases, slightly larger learning rates (during pre-training) also achieved better performance after fine-tuning.

## Discussion

Our work reinforces the bitter lesson. The most important factors determining the performance of a sensibly designed model are the compute and data available for training<sup>5</sup> (Tolstikhin et al., 2021). Although the success of ViTs in computer vision is extremely impressive, in our view there is no strong evidence to suggest that pre-trained ViTs outperform pre-trained ConvNets when evaluated fairly. We note however that ViTs may have practical advantages in specific contexts, such as the ability to use similar model components across multiple modalities (Bavishi et al., 2023).

## Acknowledgements

We thank Lucas Beyer and Olivier Henaff for feedback on an earlier draft of this note. We also thank Lucas Beyer for providing training speed estimates for ViT models on TPU-v4 devices.

## References

- I. Alabdulmohsin, X. Zhai, A. Kolesnikov, and L. Beyer. Getting vit in shape: Scaling laws for compute-optimal model design. *arXiv preprint arXiv:2305.13035*, 2023.
- R. Bavishi, E. Elsen, C. Hawthorne, M. Nye, A. Odena, A. Somani, and S. Taşırlar. Introducing our multimodal models, 2023. URL <https://www.adept.ai/blog/fuyu-8b>.
- A. Brock, S. De, S. L. Smith, and K. Simonyan. High-performance large-scale image recognition without normalization. In *International Conference on Machine Learning*, pages 1059–1071. PMLR, 2021.
- T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.
- S. Fort, A. Brock, R. Pascanu, S. De, and S. L. Smith. Drawing multiple augmentation samples per image during training efficiently decreases test error. *arXiv preprint arXiv:2105.13343*, 2021.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016a.
- K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016b.
- E. Hoffer, T. Ben-Nun, I. Hubara, N. Giladi, T. Hoefler, and D. Soudry. Augment your batch: better training with larger batches. *arXiv preprint arXiv:1901.09335*, 2019.
- J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L.

<sup>5</sup>By sensibly designed, we mean models that are sufficiently expressive and have stable gradient propagation.

- Casas, L. A. Hendricks, J. Welbl, A. Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.
- A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, and N. Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020.
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.
- I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.
- X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022.