Vision Transformers on CIFAR–100: Design From Scratch and Fine–Tuning Pretrained Swin Models

Daniel Jolin
801282735
Homework 6: Vision Transformers
https://github.com/RocketDan11/dl/homework/homework6/
04-18-25

***Abstract*—This report investigates Vision Transformer (ViT) architectures for image classification on the CIFAR–100 dataset. In Problem 1, we design and train multiple ViT configurations from scratch, comparing them against a ResNet-18 baseline in terms of accuracy, model size and training time. In Problem 2, we fine-tune pretrained Swin-Tiny and Swin-Small models and contrast their performance with a scratch-trained Swin-Tiny. Our findings highlight trade-offs among model complexity, compute cost, and classification accuracy on low-resolution imagery.**

***Index Terms*—Vision Transformer, Swin Transformer, CIFAR-100, Image Classification, Computational Complexity, Transfer Learning**

I. Introduction

Vision Transformers (ViT) reinterpret images as sequences of patches and apply self-attention to capture global context. While ViTs excel on large-scale datasets, their behavior on small-resolution benchmarks like CIFAR-100 (32×32 RGB) warrants systematic study. Additionally, pretrained hierarchical transformers such as Swin Transformers promise rapid fine-tuning on limited data. We explore two questions: (1) how ViT hyperparameters (patch size, embedding dimension, depth, attention heads) affect accuracy, model size, and compute; (2) the benefits of Swin transfer learning versus training from scratch.

II. Methodology

All experiments employ PyTorch on an NVIDIA RTX 3080ti GPU with CPU fallback. We adopt standard CIFAR-100 augmentations (4-pixel padding, random crop, horizontal flip) and normalization.

A. Problem 1: ViT From Scratch

Design space. We train four ViT variants (Table **??**) with patch sizes $P \in \{4, 8\}$, embedding dims $D \in \{128, 256, 512\}$, depths $L \in \{4, 8\}$, and heads $H \in \{2, 4, 8\}$. The MLP hidden size is $2D$. Each model uses positional embeddings. Training runs for 50 epochs, batch size 64, AdamW with $learning\_rate$=1e-3 and $weight\_decay$=1e-2.

heads	layers	patch size	hidden	mlp	size (mb)	training time (s)	accuracy (%)
4	4	4x4	256	512	333	779.5	23.52
4	4	8x8	256	512	333	752	35.76
2	4	4x4	256	512	94.12	671	35.12
8	8	8x8	128	256	333.04	745.5	24.8
4	4	8x8	256	512	87.61	823.5	39.43
2	8	4x4	256	512	95.5	681.5	37.89
4	8	4x4	256	512	648.19	1008	12.54
4	4	4x4	512	1024	648.19	1068	32.61

Fig. 1: Vit Comparison Metrics

Baseline. A ResNet-18 from torchvision is retrained for 30 epochs under identical settings.

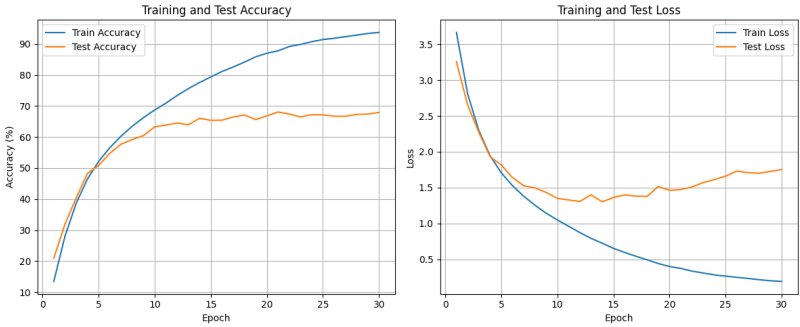

Fig. 2: Resnet Baseline Metrics

Metrics. The baseline ResNet model trains significantly faster, (508s), and is smaller (54.4mb), while achieving a higher accuracy than all ViT models (67.92%)

B. Problem 2: Swin Fine-Tuning

We load pretrained `microsoft/swin-tiny-patch4-window7-2` and `microsoft/swin-small-patch4-window7-224`, replace their heads with a 512→100 MLP, and freeze the backbone. Fine-tuning uses batch size 32, Adam LR=2e-5, 5 epochs. Additionally, we train a Swin-Tiny from scratch for 5 epochs under the same augmentations.

TABLE I: Swin Scratch vs. Finetuning

Model	Time/epoch (s)	Acc (%)
swin-tiny-finetuned	81.83	66.33
swin-small-finetuned	133.4	70.48
swin-tiny-scratch	208.2	37.09

Key Findings:
Fine-tuning vs. Training from Scratch:
- Accuracy difference: 29.24%
Swin-Tiny vs. Swin-Small: - Accuracy difference: 4.15%
Training Time Comparison:
- Swin-Tiny (pretrained): 81.83 seconds/epoch
- Swin-Small (pretrained): 133.71 seconds/epoch
- Swin-Tiny (scratch): 208.22 seconds/epoch

III. Discussion

A. ViT Hyperparameter Trade-offs

Small-patch, shallow ViTs achieve closer to ResNet accuracy with fewer parameters but larger model size due to longer token sequences. Deeper or higher-dim variants improve accuracy at substantial computational cost. Larger patches reduce sequence length and model size but may lose fine spatial details, yielding diminishing returns.

B. Transfer Learning Benefits

Pretrained Swin converges in 5 epochs to higher accuracy than scratch-trained variants, demonstrating a superior compute-to-accuracy ratio. Swin-Small further boosts accuracy at 2× parameters and 1.4× training time.

IV. Conclusion

ViTs can rival ResNet on CIFAR-100 with proper design, though their compute scales rapidly with depth and token count. Pretrained Swin Transformers offer the most efficient path to high accuracy on data-limited tasks through transfer learning.