CS 182 CV Project: Cross Vision Transformers and Sparse Attention for the Tiny Imagenet Challenge

Eric Tang
erictang000@berkeley.edu

Calvin Grewal
calvingrewal@berkeley.edu

Karthik Vegesna
kvegesna@berkeley.edu

Abstract

In this project, we attempt a number of different techniques for building a robust classifier for the Tiny ImageNet challenge. We focus on the latest models using Vision Transformers, which have been shown to help improve robustness against out of distribution examples from both white box and black box attackers. We report a top 1 validation accuracy of 81% on our architecture from fine tuning on Tiny ImageNet, using vision transformer blocks that were pretrained with ImageNet 1k, and using standard data augmentations along with AugMix. Our architecture shows performance improvements over a standard ViT model via parallel vision transformers attending to different image patch sizes combined with cross attention and an MLP head. We also observe faster training and higher clean accuracy compared with deeper stacked ViT architectures with similar numbers of parameters. We benchmark robustness and accuracy of our model against a variety of ViT and ResNet based models on Tiny Imagenet-C and with adversarial attacks from Foolbox, and evaluate the addition of cross attention and varying patch sizes, as well as the use of sparse attention, to classifying out of distribution images.

1. Introduction

Correctly classifying out of distribution images with deep neural networks has been a problem of interest in Computer Vision, as it has been noted that current models tend to overfit to their training distributions [5, 11]. This has been especially studied for Convolutional Neural Networks, which have formed the backbone of solutions to the ImageNet challenge [13], with well known architectures like ResNet and InceptionNet [4, 15], and more recent variants of these being among the most common in use today.

The Vision Transformer, introduced by Dosovitkiy et. al, is a model that uses stacked transformer encoder blocks for ImageNet classification. It has been suggested that use of such transformers can help improve robustness of models on vision tasks, especially given prior research into self attention for improving robustness for NLP tasks [7, 10]. Recent work has suggested that the higher level features learned by Vision Transformers compared to CNNs are more generalizable [14], and that Vision Transformers are more robust to black box adversarial attacks [12]. In addition, many variants of the original Vision Transformer have shown clean accuracy surpassing that of most current CNN models, with EfficientNet variants being the only remaining CNN models that still outperform Vision Transformers that have been pretrained with large external datasets [16].

Motivated by this recent work, in this project, we attempt to benchmark Tiny ImageNet performance with Vision Transformers using the original Vision Transformer (ViT) and Pooling Vision Transformers (PiT). We find that PiT from Heo et. al performs the best in terms of clean accuracy, with an 85% validation accuracy, but has increased sensitivity to adversarial perturbations. In addition to benchmarking performance of existing models with small tweaks, we propose an improvement on the base Vision Transformer architecture by using parallel vision transformers with cross attention layers that is inspired by the CrossViT architecture from Chen et. al, but adapted to take promote faster training, and to take advantage of pretrained weights from the base Vision Transformer that are available via the Pytorch Image Models Github repository [18]. We find that our architecture is able to increase validation accuracy on top of a base ViT model of a larger size, but that this comes at the cost of robustness, where the original ViT models were found to perform the best. Additionally, we experimented with applying sparse attention and batch norm adaptation to various existing models for evaluating their impact on robustness and show these results below.

2. Literature Survey and Related Work

2.1. Vision Transformers (ViT)

Transformers [17] are mainstream in NLP for many tasks, where they achieve state of the art results in tasks like text generation and question answering [1]. The original transformer architecture from Vaswani et al. consists of an encoder block that feeds into a multi head attention block in a separate decoder block. Vision transformers, as

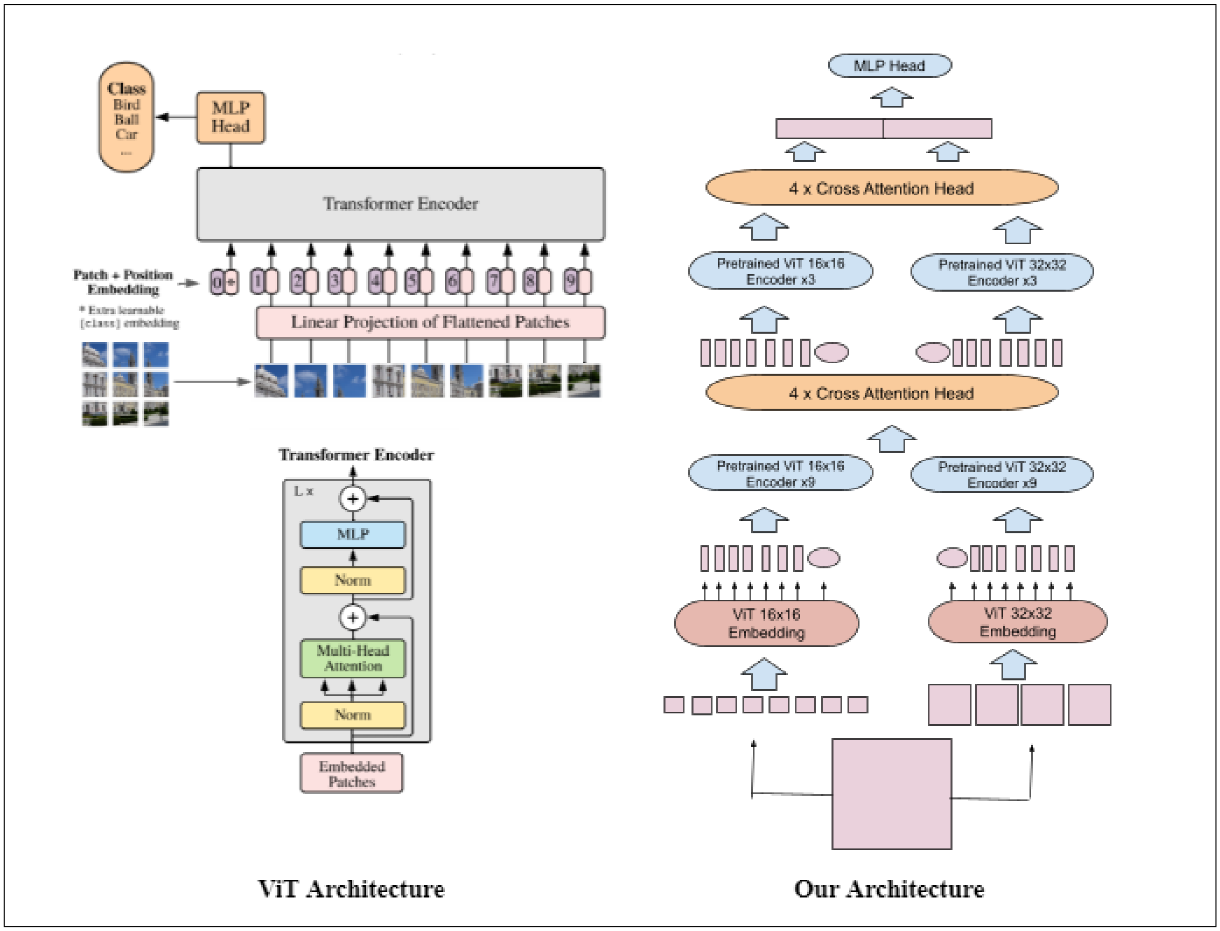

Figure 1. **Left:** Original vision transformer architecture with detailed Transformer Encoder Block. **Right:** Our architecture based off of CrossViT adapted to using pretrained encoder block modules with 2 cross attention layers and a concat before passing into an MLP head.

introduced by Dosovitkiy et al, consist solely of stacks of 12-24 transformer encoder blocks, with images that are first split into fixed-size patches (usually 16x16 or 32x32) then flattened, passed through a linear projection layer, and then passed through a positional embedding layer. Finally, the resulting sequence is fed through the transformer encoder blocks, with the final output then going through an MLP with the number of classes as the final output. This model achieved state of the art results on ImageNet and showed that fully attention based architectures could be used for vision tasks.

2.2. ViT Variants (PiT and CrossViT)

The Pooling-based Vision Transformer (PiT), proposed by Heo et al. builds on the work of ViT and uses additional pooling operations to improve baseline model performance and reduce reliance on large pretraining sets like JFT-300M. In addition, PiT outperforms ViT on several robustness benchmarks, including resistance to FGSM attacks, ImageNet-A performance, and occlusion benchmarks [9]. The Cross-attention Vision Transformer (CrossViT), proposed by Chen et al. combines image patches of different sizes in a transformer to produce stronger image features. The CrossViT uses transformer encoders made up of a large primary branch with coarse-grained patch sizes and a small complementary branch that uses fine-grained patch sizes. The branches consist of stacked transformer encoder blocks which attend to the other branch periodically via cross attention blocks. CrossViT also showed improvements over the ViT baseline.

2.3. Tiny ImageNet-C

ImageNet-C and Tiny ImageNet-C are datasets generated by applying 15 algorithmically generated corruptions to the ImageNet and Tiny ImageNet datasets [6]. Since this project is focused on developing a robust model against unknown test-time perturbations, the Tiny ImageNet-C dataset provides a very useful benchmark for our model. For this project, we randomly chose 5 of the corruptions to test our model with: elastic transform, motion blur, snow, noise impulse, and JPEG compression.

3. Methods and Approach

3.1. Data Augmentations and Preprocessing

While training our models, we experimented with various data augmentation techniques. First, we used random cropping, random horizontal/vertical flipping, random rotation, color jittering, and Cutout [3]. After experimenting with various combinations of these transformations, we discovered that models trained with AugMix [8] consistently outperformed any of the other combinations of transformations we tried, and so we used it for the final robustness evaluations of all of our models, in addition to keeping horizontal flips and random cropping. The implementation of AugMix we used from the Pytorch Image Models repository also uses contrast, color, brightness, and sharpness in addition to the AugMix transforms from the original paper. Lastly, we resize all our images to 224 by 224 before feeding it into our model to fit the trained input sizes for the models we selected.

3.2. Cross Attention and Parallel Transformer Blocks

Our main architectural change that we present is using cross attention across otherwise parallel vision transformers in order to improve classification performance. The novelty of our model architecture is in it's use of pretrained ViT encoder blocks, combined with the idea of using cross attention across encoder blocks processing different sized patches in order to capture a larger variety of spatial dependencies in the image than a convolutional model or a single ViT model would. It draws inspiration from similar work done for CrossViT [2], but is distinct in that it uses cross attention only towards the end of the sequence of transformer encoder blocks in order to help the models learn features separately, and to separate the gradient flow for faster training.

3.3. Sparse Attention and Batch Norm Adaptation

One idea that we experimented with before leaving it behind due to drops in accuracy, was applying Sparse Attention from Zhao et al [19] to pretrained models in order for the model to be able to focus on extracting information from other more spatially relevant tokens by explicitly masking out attention values. We report exact numerical results below. Overall, we found that for the values Zhao et al. reported (k = 8, 16, 32), our accuracy decreased with no gain in robustness to adversarial attacks. We did find that models trained with Sparse Attention performed much better when inference was also run with Sparse Attention, and that allowing for the full attention mechanism at test time generally decreased both clean and robust accuracy, which we believe is due to the learned weights assuming that the attention is sparse, and only attends to the image patches with relatively stronger signal.

An additional idea we considered was using Batch Norm Adaptation, a technique used to try and update the batch norm layers to use test time statistics rather than training time statistics to try and account for test time distribution shift. Although this generally improved results, we avoided using it in our final implementation due to the suggestion of a TA to stick with strictly running inference and not adapting at test time.

3.4. Training and Transfer Learning

A key component of our method is that we use transformer blocks with pre-trained weights on ImageNet. We freeze all but the last 90 layers of the transformer and then attach a classification head composed of Linear, ReLU, and Dropout layers. Finally, we use AugMix on the TinyImageNet data and fine-tine the model.

For our training, we use the Adam optimizer with two different initial learning rates: 1e-4 for the pre-trained transformer and 1e-3 for the classification head. We also included a cosine annealing scheduler to adjust the learning rates during training. Lastly, we used a batch size of 32 and trained for 10 epochs on most models. A challenge that we encountered while trying to make architectural changes was in making sure that pretrained weights were not perturbed which led us to make the above architectural decisions.

4. Results

4.1. Comparison to Larger ViT

Empirically, our parallel architecture with a similar number of model parameters to a larger vision transformer was able to outperform the deeper stacked serial model, and greatly improved speed of training, with each epoch taking roughly 50 minutes on a Tesla V100 GPU to train for the ViT_L_patch16_224_in21k model, consisting of 24 transformer encoder layers, vs each epoch taking roughly 20 minutes on the same GPU for the parallel model consisting of the smaller ViT_B_patch16_224 and ViT_B_patch32_224 models, each with 12 transformer encoder layers, with outputs simply concatenated and passed to an MLP for output. Additionally, we observed an error rate of 18.9% for our model, which topped the performance of the single larger vision transformer, with an observed error rate of 20.1%.

4.2. Error Analysis

To evaluate our model's performance, we look at the top 1 accuracy on plain Tiny ImageNet as well as the average accuracy on each of the corrupted datasets. The table below gives a summary of each model's performance.

Robustness

Model	Baseline	Impulse Noise	Snow	Motion Blur	Elastic Transform	Jpeg	Average	% Change
ViT	79.2	68.0	68.0	70.63	67.2	70.5	68.9	**-13.0**
PiT	**85.3**	67.6	73.0	72.6	70.6	73.1	71.4	-16.3
DoubleViT	81.1	67.7	67.6	70.5	67.6	71.1	68.9	-15.0
DoubleViT+Cross	81.2	67.1	68.7	71.2	67.8	72.1	69.4	-14.6
DoubleViT+2Cross	80.7	65.9	68.4	71.2	68.2	71.9	69.1	-14.4

Table 1. Model accuracies across different Tiny ImageNet-C perturbations for Vision Transformer models. Each model was evaluated on the base TinyImageNet, as well as each of the corrupted datasets.

Clean Accuracy

Model	Top 1 Val Accuracy	Top 3 Val Accuracy
PiT	**85.5**	93.9
PiT+Aug	85.3	93.7
PiT+Aug+Sparse	79.7	90.3
ViT	78.4	88.1
ViT+Aug	79.1	88.8
ViT+Aug+Sparse	75	86.9
InceptionResnetV2	67.3	80.5
InceptionResnetV2+Aug	67.9	80.3
DoubleViT	81.1	90.8
DoubleViT-Cross	81.2	91.6
DoubleViT-2Cross	80.7	91.9

Table 2. Top 1 and 3 validation accuracies. DoubleViT is our model.

We see that PiT achieved the highest top 1 accuracy on TinyImageNet. DoubleViT is next best, achieving 81.1%, 81.2%, and 80.7% accuracy on DoubleViT with 0, 1 and 2 cross attention layers, respectively. Finally, the base ViT performed the worst out of the models tested on standard Tiny ImageNet. We observe that the addition of pooling layers contributes strongly to the overall performance of the model on Tiny ImageNet, and that the addition of cross attention layers to our own model architecture seems to correlate to an increase in top 3 validation accuracy. We hypothesize that the reason that the 2 cross attention layer DoubleViT model has a lower top 1 validation accuracy is due to the larger number of weights that needed to be updated with the addition of more layers, which is a problem with purely using pretrained weights to build our model. This is a constraint in terms of compute that cannot be overcome without additional funding, and so we leave the task of training our model from scratch with external data to future work.

4.3. Ablations and Robustness Benchmarking

For robustness analysis, we measure robustness by the % change between the top 1 accuracy on Tiny ImageNet and the average top 1 accuracy on the corrupted data.

As we can see, although PiT achieved the highest top-1 accuracy on TinyImageNet, it was the least robust against our corruptions. We hypothesize that this is due to the fact that the pooling layers create dependencies between encoder layers that may not otherwise exist. We observe that the vanilla ViT recorded the lowest accuracy on vanilla TinyImageNet but was the most robust in our experiment.

Our model offers a compromise between the two – we offer a TinyImageNet accuracy higher than ViT combined with better robustness than PiT. Furthermore, we see that as the number of cross attention layers increases from 0 to 2, the robustness also increases. Thus we tentatively conclude that the addition of cross attention layers can improve the robustness in vision transformer based models.

5. Conclusion/Lesson Learned

Our vision transformer model with parallel ViT blocks attending to different patch sizes combined with cross attention demonstrated Top-1 accuracy of 81% on Tiny ImageNet after being trained for just 2 epochs with AugMix. We found that there is a potential benefit in using additional cross attention for parallel vision transformer models, both for clean accuracy, as well as robustness, and show a way in which models like CrossViT can easily be recreated from pretrained transformer encoder blocks for faster training in transfer learning settings like this one.

6. Team Contributions

Eric worked on modeling and evaluation, Calvin on robustness evaluation, and Karthik on drafting the report. 60:30:10%

References

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.

[2] C. Chen, Q. Fan, and R. Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. *CoRR*, abs/2103.14899, 2021.

[3] T. DeVries and G. W. Taylor. Improved regularization of convolutional neural networks with cutout, 2017.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[5] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *CoRR*, abs/2006.16241, 2020.

[6] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[7] D. Hendrycks, X. Liu, E. Wallace, A. Dziedzic, R. Krishnan, and D. Song. Pretrained transformers improve out-of-distribution robustness. *CoRR*, abs/2004.06100, 2020.

[8] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty, 2020.

[9] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh. Rethinking spatial dimensions of vision transformers, 2021.

[10] Y.-L. Hsieh, M. Cheng, D.-C. Juan, W. Wei, W.-L. Hsu, and C.-J. Hsieh. On the robustness of self-attentive models. July 2019.

[11] P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, S. Beery, J. Leskovec, A. Kundaje, E. Pierson, S. Levine, C. Finn, and P. Liang. WILDS: A benchmark of in-the-wild distribution shifts. *CoRR*, abs/2012.07421, 2020.

[12] K. Mahmood, R. Mahmood, and M. van Dijk. On the robustness of vision transformers to adversarial examples. *CoRR*, abs/2104.02610, 2021.

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[14] R. Shao, Z. Shi, J. Yi, P. Chen, and C. Hsieh. On the adversarial robustness of visual transformers. *CoRR*, abs/2103.15670, 2021.

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014.

[16] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2017.

[18] R. Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

[19] G. Zhao, J. Lin, Z. Zhang, X. Ren, Q. Su, and X. Sun. Explicit sparse transformer: Concentrated attention through explicit selection. *CoRR*, abs/1912.11637, 2019.