CONVOLUTIONS OR TRANSFORMATIONS: A COMPARISON OF CNN NETWORKS WITH VISION TRANSFORMER NETWORKS

Alec Braynen

ABSTRACT

In this work, the author explores the performance of basic Convolutional Neural Networks (CNN) in comparison to basic Vision Transformers (VTs). The author created three CNNs and three VT models of different sizes. The CNN networks are composed of a differing layouts of convolutional, pooling and dense layers, while the VT networks are composed of encoder layers, multi-layered perceptrons and attention layers. The networks were trained and tested on the CIFAR100, CIFAR10, and Fashion MNIST datasets. On the CIFAR-10 and CIFAR-100 dataset, the VT model outperforms the CNN model, however on the MNIST dataset the reverse occurs. The models' architectures are discussed, and the results are compared against the ViT and CvT architectures. We conclude that Vision Transformers seem to improve performance over CNNs on image classification tasks, however this improvement seems to come at the cost of increased training time.

1. INTRODUCTION

Transformers were a game-changing discovery for Artificial Intelligence and machine learning. They underpin a great deal of the progress made in Natural Language Processing; allowing us to talk to virtual assistants on our phone and computers.

The influential paper, [1] described how transformers could be used to replace or reinforce Recurrent Neural Network architectures. Recently, it has been shown that transformers can be applied to vision problems.[2] These transformers, called vision transformers, are currently deployed in some state-of-the-art models on benchmarks of image datasets. Historically, CNNs have been the dominant network architecture for image classification tasks. But this may change if VTs are able to consistently surpass them in state-of-the-art performance.

A basic CNN architecture would contain a convolutional layer, a pooling layer, a dense layer and an output layer. The convolutional layer filters the images, the pooling layer abstracts the important features, and the dense layer and output layer are fully connected layers to the classification neurons.

A basic vision transformer contains an encoder that is comprised of a multi-head attention layer, a normalization layer and a multi-layered perceptron.

In this work, the author presents experiments comparing the performance of three different VTs with three different CNNs. The models are trained and tested on the CIFAR10, CIFAR100 and Fashion MNIST dataset. The three different VT models had better performance than the equivalent CNN models on every dataset except the Fashion MNIST dataset, where the CNNs outperformed the VTs.

This work indicates that Vision Transformers may be a fruitful direction of research towards better performance on image recognition.

2. RELATED WORKS

In [2], Dosovskiy, Alexey, et al. proposed a novel vision transformer (ViT) architecture that showed VTs could be applied to image recognition problems. They apply a transformer architecture directly to image data and achieve excellent results on image classification. Furthermore, results are improved by pre-training the model large datasets and testing it on ImageNet, CIFAR-100 or CIFAR-10 datasets, the model approaches state of the art performance. The authors were able to adapt transformers to image recognition, by splitting image data into fixed-size patches, arranging the data linearly, and adding position vectors and a classification token to the dataset. This allowed them to train the transformer on image data.

In [3], Haiping et al. present a novel Convolutional Vision Transformer architecture. Their architecture, CvT, extends the ViT transformer discussed above. They extend the ViT model by introducing convolutional token embedding, and a convolutional transformer block that leverages a convolutional projection. This allowed them to introduce features from CNN models into their CvT model. Additionally, the authors 1) partitioned the transformer compoent into multiple stages to form a

heirarchal structure of transformers. Each stage begins with a convolutional token embedding that performs overlapping convolution operations. And 2) replaced the linear projection with a convolutional projection.

In [4], Fan et al. presented a Multiscale Vision Transformer (MViT) model. The MViT model is a combination of multiscale feature heirarchies and transformer models. Each stage in the transformer has a different resolution capacity that down-scales the image spatial resolution while increasing the channel capacity. The authors propose that this architecture give the model an implicit temporal bias and allows the early layers to operate at high spatial resolution to model low-level information and the deeper layers focus on complex high-level features.

In [5], Yawei et al. introduce LocalViT, a vision transformer with locality mechanisms. The authors state that in ViT, the model lacks a locality mechanism. They state that by adding a locality mechanism to ViT, the new model is able to better understand feature ssuch as lines, edges, shapes and objects. They add this locality to the model via a depth-wise convolutional layer. They go on to further explain, that adding locality opens up design choices to further improve performance.

In [6], Rao et al. propose the model Dynamicvit, a model with dynamic token sparsificition to increase the amount of attention in the vision transformer to the image datasets. Dynamic token sparsification prunes redundant tokens dynamically based on the input. The authors achieve this by implementing a prediction module to estimate the importance of each token given the current features in the model. This prediction module is added in different layers to remove tokens hierarchically. By pruning tokens, Rao et al. were able reduce 31-37% FLOPs and improve throughput by 40% while only sacrificing 0.5% of accuracy.

In [7] Qing-Long et al. implement a CNN with a Shuffle Attention module to improve performance, without the overhead of combining a spatial attention module and a channel attention module. This Shuffle Attention module seems to enhance the performance of CNN networks. The module constructs channel attention and spatial attention simultaneously. This implementation allowed them to achieve state of the art performance with fewer parameters than other state-of-the-art models at the time.

The authors in [8], propose a new activation function called “Fast Exponentially Linear Unit Activation Function (FELU),” to improve on the problems of gradient disappearance, neuron death, output offset, etc. Qiumei et al. state that FELU improves classification accuracy and calculation speeds. The authors implement FELU on a simple CNN architecture with two convolutional layers, two pooling layers and two fully connected layers. In their experiments FELU improved the accuracy of their model over RELU by 2-6%.

Deboleena et al. propose Tree-CNN in [9]. Tree-CNN is a continuous learning model that grows in a “tree-like” fashion with new data. Their model aims to address the “catastrophic forgetting” issue in current deep models. Tree-CNN is a network comprised of CNNs that grows as new classes are learned. The network branches the new classes based on the similarity of the new classes to the old ones. The initial nodes hold coarse image features, while the newer nodes hold finer features. The network achieves an accuracy of 69% on the CIFAR-100 dataset.

In [10], the authors propose a “defense layer,” for convolutional neural networks, to protect against adversarial attacks and to improve robustness against them. The defense layer is a parameter-free layer that protects against FGSM, L_2, and DeepFool attacks. Akhil Goel et al, implement their defense layer on VGG, ResNet and DenseNet. The models were tested with this defense layer on the MNIST, CIFAR-10 and PaSC datasetm and results showed that the defense-layer was able to preserve accuracy on these benchmarks, despite the adversarial attack.

Xiangyu et al. proposes a novel architecture called WA-CNN in [11]. WA-CNN has a Wavelet-Attention block that implements attention only in the high-frequency domain. Low frequency and high frequency components are stored along with detailed information and noise for the components. In the CIFAR-10 and CIFAR-100 datasets, WA-CNN achieves a 1.26% and 1.54% accuracy improvement respectively.

3. METHOD

In developing our three models of VTs and CNNs, we used the number of layers as the measure of size for each network. The small architectures have the least number of layers, while the base architectures have the most. Additionally, this means that the different architectures have differing amounts of parameters as well. However, this was a secondary design criteria in

determining size.

3.1 TINY VT

The tiny visual transformer architecture is comprised of 4 blocks of a normalization layer, multi-head attention layer and multi-layer perceptron layer. After these 4 blocks, there are three additional layers: 1) a normalization layer, 2) a flatten layer and 3) a dropout layer.

3.2 TINY CNN

The tiny CNN is comprised of a 2D convolutional layer, a max pooling layer, another 2D convolutional layer followed by a max pooling layer, then a flatten layer, a dense layer and then the output layers.

3.3 SMALL VT

The small visual transformer architecture is comprised of 6 blocks of a normalization layer, multi-head attention layer and multi-layer perceptron layer. After these 4 blocks, there are three additional layers: 1) a normalization layer, 2) a flatten layer and 3) a dropout layer.

3.4 SMALL CNN

The small CNN is comprised of a 2D convolutional layer, followed by a max pooling layer, then another 2D convolutional layer, followed by a max pooling layer, then another 2D convolutional layer, followed by a pooling layer, a flattening layer, two dense layers and then the output layer.

3.5 BASE VT

The base visual transformer architecture is comprised of 8 blocks of a normalization layer, multi-head attention layer and multi-layer perceptron layer. After these 4 blocks, there are three additional layers: 1) a normalization layer, 2) a flatten layer and 3) a dropout layer.

3.6 BASE CNN

The base CNN is comprised of a 2D convolutional layer, followed by a max pooling layer, then another 2D convolutional layer, followed by a max pooling layer, then two 2D convolutional layers, followed by a flattening layer, three dense layers and then the output layer.

3.7 HYPERPARAMETERS AND TRAINING

Both the CNNs and VTs were trained for fifteen epochs, and used a batch size of 256, and a loss function of Sparse Categorical Cross Entropy. The VTs use an AdamW optimizer and the CNNs use an Adam optimizer. Both optimizers had a learning rate of 0.001. The VTs also, have a weight decay of 0.0001 and an image patch size of six. The CNNs use the ReLU activation function in the convolutional layers and a SoftMax activation function on the output layers. Each convolutional layer has a kernel size of 3x3, and the pooling layers have a size of 2x2.

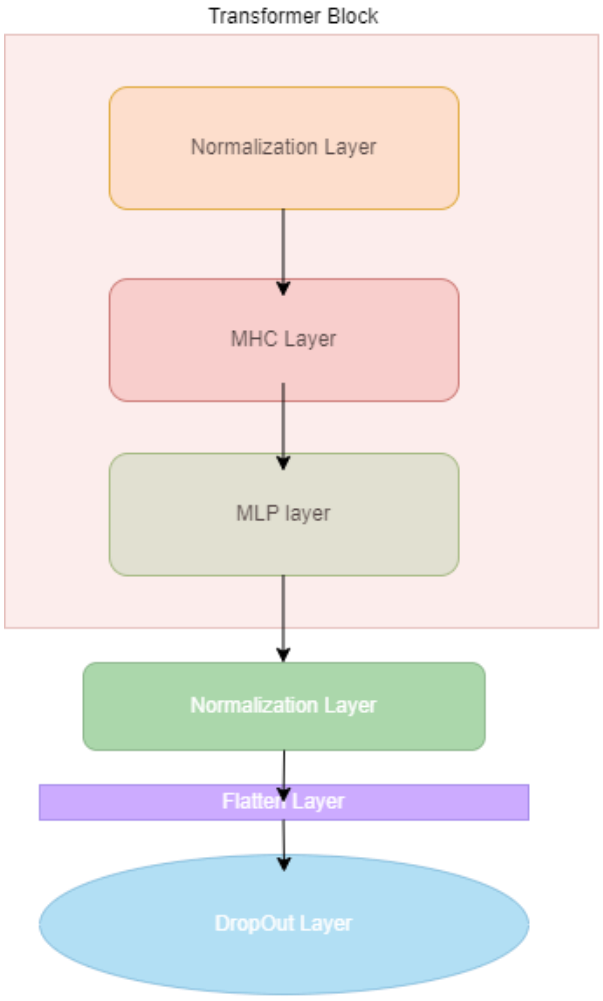

Figure 1. Transformer Architecture

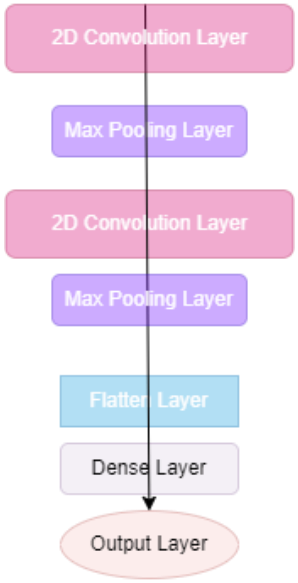

Figure 2. CNN Architecture

4. EXPERIMENTS, DATASETS AND RESULTS

We tested the models on the CIFAR-10, CIFAR-100 and Fashion MNIST datasets.

The dataset descriptions and our results are shown below.

4.1 CIFAR-10

The CIFAR-10 dataset is an image dataset with ten classes. The models were trained on fifty thousand training image samples and tested on 10000 test image samples.

The results are shown in Table 1.

4.2 CIFAR-100

The CIFAR-100 dataset is a more difficult image dataset with 100 classes. The models were trained on fifty thousand training image samples and tested on 10000 test image samples.

The results are shown in Table 2.

4.3 FASHION MNIST

The Fashion MNIST dataset is a grayscale image dataset with ten classes. The models were trained on sixty thousand training image samples and tested on 10000 test image samples.

The results are shown in Table 3.

4.4 EXPERIMENTS AND RESULTS

We tested and compared the VTs and the CNNs on the CIFAR-10, CIFAR-100 and Fashion MNIST datasets. Our results showed that the VTs outperformed the CNNs in accuracy on the CIFAR-10 and CIFAR-100 datasets. These results were consistent at every comparable model size. However, the CNNs outperformed the VTs on the Fashion MNIST dataset in all sizes.

We propose that these results indicate that the Vision Transformers were better able to acquire features from the more complicated CIFAR-10 and CIFAR-100 datasets. However, on the grayscale and lower resolution Fashion MNIST dataset, the CNN models were able to extract the same features as the Vision Transformers and subsequently outperform the VTs.

CIFAR 100	Tiny VT	Tiny CNN	Small VT	Small CNN	Base VT	Base CNN	ViT [2]	CvT [3]
Macro Precision	0.41639	0.3670427	0.427684	0.35506439	0.406237	0.3075451		
Micro Precision	0.3905	0.3557	0.4081	0.3474	0.386	0.2988		
Macro Recall	0.3905	0.3557	0.4081	0.3474	0.386	0.2988		
Micro Recall	0.3905	0.3557	0.4081	0.3474	0.386	0.2988		
F1 Score	0.385433	0.3533879	0.401599	0.34554376	0.380193	0.2910662		
Accuracy	39.05%	35.57%	40.81%	34.74%	38.60%	29.88%	94.55%	94.09%
Top 5 Accuracy	69.61%	63.45%	71.19%	62.65%	68.96%	58.37%		
Loss	2.3546	6.1599	2.262	5.0455	2.3917	3.0478		

TABLE 1: CIFAR 100 RESULTS

CIFAR 10	Tiny VT	Tiny CNN	Small VT	Small CNN	Base VT	Base CNN	ViT [2]	CvT [3]
Macro Precision	0.738279	0.7002881	0.748758	0.70717233	0.758317	0.7122428		
Micro Precision	0.7397	0.6975	0.7423	0.7094	0.7475	0.7061		
Macro Recall	0.7397	0.6975	0.7423	0.7094	0.7475	0.7061		
Micro Recall	0.7397	0.6975	0.7423	0.7094	0.7475	0.7061		
F1 Score	0.736836	0.6959802	0.741229	0.70733204	0.746707	0.7065702		
Accuracy	73.97%	69.75%	74.23%	70.94%	74.75%	70.61%	99.50%	99.39%
Top 5 Accuracy	98.34%	96.63%	98.42%	96.89%	98.35%	96.86%		
Loss	0.7356	1.9659	0.734	1.5375	0.7147	1.1148		

TABLE 2: CIFAR-10 RESULTS

Fashion MNIST	Tiny VT	Tiny CNN	Small VT	Small CNN	Base VT	Base CNN
Macro Precision	0.892157	0.9047017	0.893119	0.89223606	0.89642	0.9123916
Micro Precision	0.8896	0.9045	0.8879	0.8899	0.895	0.9102
Macro Recall	0.8896	0.9045	0.8879	0.8899	0.895	0.9102
Micro Recall	0.8896	0.9045	0.8879	0.8899	0.895	0.9102
F1 Score	0.89026	0.9034669	0.888571	0.89040485	0.894753	0.9107914
Accuracy	88.96%	90.45%	88.79%	88.99%	89.50%	91.02%
Top 5 Accuracy	99.85%	99.82%	99.83%	99.78%	99.85%	99.73%
Loss	0.3036	0.5447	0.2956	0.4409	0.3117	0.361

TABLE 3: FASHION MNIST RESULTS

5. DISCUSSION AND CONCLUSION

In this work, we tested and compared three VTs of differing sizes with three CNNs of differing sizes. The models were tested on the CIFAR-10, CIFAR-100 and Fashion MNIST datasets respectively.

Results showed that the VTs outperformed the CNNs on the CIFAR datasets. However, the CNNs outperformed the VTs on the MNIST dataset. We propose that these results reflect the capabilities of the networks to extract features from complex and simple images. The VTs did a better job of extracting features from the more complicated CIFAR datasets than did the CNNs. However, on the simpler MNIST dataset, the CNNs outperformed the VTs.

This indicates that perhaps if we preprocessed the data further on the CIFAR datasets that the CNNs may outperform the VTs. However, we would need to perform further experiments and tests to be sure.

6. REFERENCES

[1] A. e. a. Vaswani, "Attention is all you need.," *Advances in neural information processing systems*, 2017.

[2] A. e. a. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale.," *arXiv preprint arXiv*, vol. 2010.11929, 2020.

[3] H. e. a. Wu, "Cvt: Introducing convolutions to vision transformers.," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[4] H. e. a. Fan, "Multiscale vision transformers.," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[5] Y. e. a. Li, "Localvit: Bringing locality to vision transformers.," *arXiv preprint arXiv*, vol. 2104.05707, 2021.

[6] Y. e. a. Rao, "Dynamicvit: Efficient vision transformers with dynamic token sparsification.," *Advances in neural information processing systems* , 2021.

[7] Q.-L. a. Y.-B. Y. Zhang, "Sa-net: Shuffle attention for deep convolutional neural networks.," in *CASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

[8] Z. T. D. a. W. F. Qiumei, "Improved convolutional neural network based on fast exponentially linear unit activation function.," *Ieee Access 7*, 2019.

[9] D. P. P. a. K. R. Roy, "Tree-CNN: a hierarchical deep convolutional neural network for incremental learning.," *Neural Networks* , vol. 121 , 2020.

[10] A. e. a. Goel, "DNDNet: Reconfiguring CNN for adversarial robustness.," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[11] X. P. H. a. X. S. Zhao, "Wavelet-Attention CNN for image classification.," *Multimedia Systems*, 2022.