VISION TRANSFORMERS VS. CONVOLUTIONAL NEURAL NETWORKS

Xiaowen Li

ABSTRACT

Image classification is one of the most important and the most extensive topics in deep learning neural networks. Convolutional neural networks(CNNs) are known as a common method in computer vision for complex tasks. Lots of state-of-art works were based on CNNs until the vision transformer method was brought to the field.

In this paper, we are going to examine the performance of both the CNNs and vision transformer models with different model sizes. Three publicly available datasets are used for the experiment - CIFAR-10, CIAFAR-100, and Fashion-MNIST.

The experimental analysis evaluated the performance of the image classification model by both the test accuracy and loss value. And comparisons are made not only between CNNs and vision transformer models but also between the same model with different input or model sizes.

We observe that both CNN models and vision transformer models give better test accuracy when the size sets to tiny or small. And comparing performances between the two models, we see that in terms of test accuracy, CNN beats vision transformers in 50 epochs, but in terms of loss values, the vision transformer has a better performance when the size of the model sets to base.

1. INTRODUCTION

The original transformer is a deep learning model that is used in the field of natural language processing. Internally, the transformer learns by measuring the relationship between input token pairs. When transferring it to the computer vision field, we can take patches of images as the token. This method was well explained in the paper [1]. The most difficulty people face to make transformer work for image classification is the cost of the attention mechanism, while local attention is limited by hardware, people found global attention was the solution. As attention was on image patches, positional embedding is necessary to let the transformers know where each patch will fit. All patches with their individual positional embedding will then feed into the transformer encoder. There is an extra learnable embedding to output the final classification of the entire image.

And on the other hand, convolutional neural networks have been practiced for image classification and proved its powerfulness in this field. In this paper, we take the classical convolutional 2D model as the benchmark to evaluate the performance of vision transformers. We also explore how both models will be affected by the model size and by different datasets taken as their inputs.

2. RELATED WORKS

Before we introduce the paper [1] that opens up the mind of transformers in the field of computer vision, all stories of transformers start with Vaswani, etc.'s paper *Attention Is All You Need* [2] - they came up with a brand new network architecture, the transformer, that completely dispensed the use of recurrence and convolutions and take the advantage of the attention mechanism that connects the encoder and decoder.

This new architecture was first practiced in the natural language processing field. And people start to extend its usage. During this time, people combine it with convolutional layers for image classification. And finally in the year of 2020, Dosovitskiy, etc published the paper [1], said that the reliance on CNNs is not necessary. They introduced vision transformer which can directly takes sequences of images and showed that this architecture had excellent performance even compares to the state-of-art CNN models.

Comparing the results from the vision transformer with the high accuracy well trained CNN models have become some standard to evaluate the ability of the vision transformer. In the paper [3], Cuenat and Couturier did the comparison on Digital Holography data, and found that while Vision Transformer reaches similar accuracy as CNNs but it is also more robust than CNNs.

Another comparison was done with High-resolution (HR) synthetic aperture radar (SAR) image data. To find a solution to the issue of CNNs' weakness on capture global information of images. Liu, etc. in their paper [4] proposed an end-to-end network global-local network structure (GLNS) that combines a lightweight CNN and a compact vision transformer and achieved the highest accuracy on their data than the pure CNN.

Do Vision Transformers See Like Convolutional Neural Networks? [5], Raghu and his fellows questioned how exactly vision transformers solve those complex tasks? Was it somewhere similar to CNNs or totally different? They analyzed the internal representation structure of vision transformers and CNNs and found that they are totally different in terms of structures and due to this, vision transformers are able to successfully preserve input spatial information.

On the other hand, convolutional neural networks are well introduced in Albawi, etc.'s paper [6]. CNNs takes its name from the mathematical linear operation between matrices called convolution. A complete CNN involves convolutional layers, non-linearity layers, and fully connected layers. CNNs represent a huge breakthrough in image classifications.

Some benchmarks done by CNNs are: AlexNet [7] that uses grouped convolutions and allows the model to run parallelism over two GPUs; Inception-V1 [8] that Uses the dimension reduced inception module; VGG [9], an innovative object-recognition model that supports up to 19 layers; ResNet [10] that solves the "vanishing gradient" problem, etc.

3. METHOD

In this paper, we perform training and testing on three different image datasets - CIFAR-10, CIFAR-100, and Fashion-MNIST with both a CNN structured model and the vision transformer model that works on different sizes. The names of the size we are using in this paper are tiny, small, and base, with tiny being the smallest and base to determine the largest possible size for the experiment. The exact meaning of the size is defined differently on CNN and vision transformer models: for CNN, size was defined by how many convolutional layers are involved in the model. The architecture of the CNN model was given below:

One 64 filter kernel size 3 2D convolutional layer with $relu$ activation function to take the input then a size 2 max-pooling followed right after it. With size tiny, this is all convolutional layer involved in this model. If the size is small we add one more 54 filter kernel size 3 2d convolutional layers with $relu$ activation function and a size 2 max-pooling layer. If the size is base, a third convolutional layer and max-pooling layers will be added to the model with this time the convolutional layer has 32 filters. Then followed with a flatten layer and then a 512 neurons densely connected layer, and finally a softmax activation output layer with 10 or 100 possibly classes depending on which dataset we use as input. A figure [1] is also generated showing the architecture for CNN model.

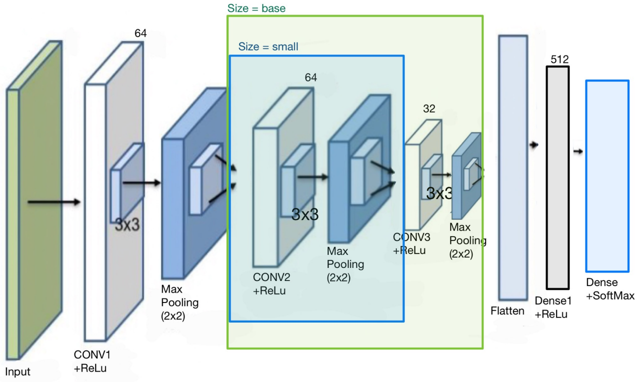

Fig. 1: CNN model Architecture

And for the vision transformer model, size was determined by how many layers of the transformer block was computed in the training process with tiny being 4, small being 6, and base being 8. This transformer block contains first one layer normalizationand then a 4-head attention layer, skipping connection 1 and process to the second layer normalization then a multilayer perceptron is called then connection 2 is skipped. A detailed explanation on vision transformer from *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*[1] is attached [2].

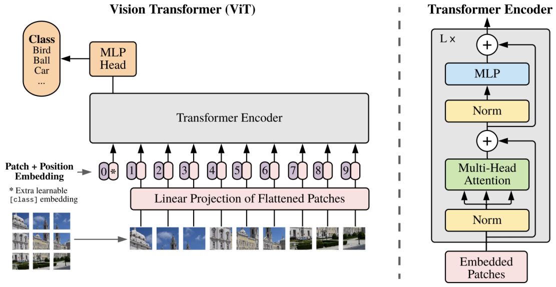

Fig. 2: Diagram explaining vision transformer [1]

From the paper [1], the authors made it possible to use the already existing training model Transformers - which is commonly used in the Natural language processing (NLP) field - to image patches and showed this method works well in large datasets on image classification.

And in our experiment we are going to perform such vision transformer model on 3 different image datasets and compare its result with same experiment but done with the CNN model[1].

4. EXPERIMENTS AND RESULTS

4.1. Datasets

3 different publicly available image datasets are used in this experiment. All can be directly imported with TensorFlow

datasets.

4.1.1. *CIFAR-10*

The CIFAR-10 datasets [3] consist of 60000 32 × 32 color images that are equally distributed into 10 classes. Dataset can be split into the training set and test set, for 50000 and 10000 images correspondingly. All classes are mutually exclusive and no overlaps between any 2 classes, this leads to 6000 individual images for each class.

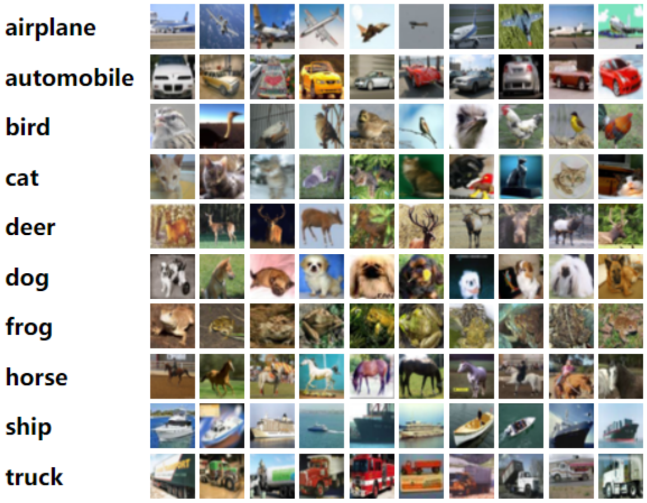

Fig. 3: example of sample data from CIFAR-10 [11]

4.1.2. *CIFAR-100*

The CIFAR-100 datasets also consist of 60000 32 × 32 color images, but are equally distributed into 100 classes. With no over lap between any two classifications, this leads to 600 images per class. This 100 classes can be grouped into 20 super classes. Both the ’fine’ classes and super classes can be used as decision targets for training, in our experiment, we consider all 100 ’fine’ classes as our targets. This datasets is split for training and testing purpose, with 50000 images for training and 10000 for testing.

4.1.3. *Fashion-MNIST*

The Fashion-MNIST [4] is a dataset of Zalando’s article images, consisting of 60000 training images and 10000 testing iamges but with image shape being 28 × 28 and are all grayscale. Each image is associated with a label from one of the 10 classes. Also, no class is overlapped.

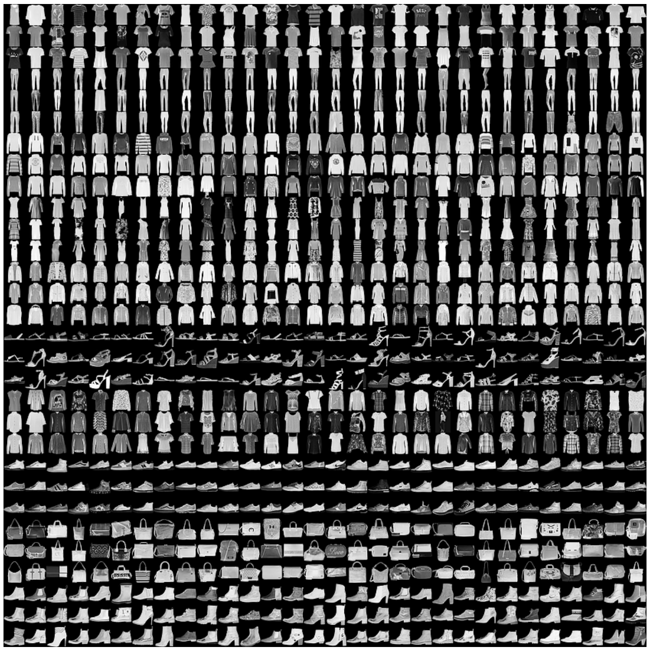

Fig. 4: example of sample data from Fashion-MNIST [12]

4.2. Implementation and Required Packages

We run our experiments on TensorFlow 2 with python version 3.9. Keras API is used in experiments.

4.3. Experiment Methods

Our experiment is aimed to build a vision transformer model as explained in paper [1] and compare its training performance with a convolutional neural network model for image classification. Comparision is taken between 2 models with different dataset inputs and the different sizes of models explained in section [3]. The following metrics are taken for the evaluation:

- Accuracy
- Recall (macro and micro)
- Precision (macro and micro)
- F1 score (macro and micro)
- Confusion Matrix

To control variables for comparisons we fix all hyperparameters by balancing training performance with time efficiency for each model. For CNN, we set the number of epochs to be 50, and the number of batches to 128. We pick Adam as our optimizer with a learning rate of 0.0001 and sparse categorical cross-entropy as the loss function since we are dealing with multi-classes image classification. For the vision transformer model, we pick the number of epochs to be 50 and the number of batches to be 256, notice that these will terminate the training process before the accuracy and loss converge to a certain number, this setup was chosen to cut down the runtime.

Checkpoint is used with picking the largest validation accuracy to save the model.

4.4. Experiment Observations

We process the training with both convolutional neural network and vision transformer models as described in previous section. We fixed number of epochs as 50 for both models.

For the experiment, we run 3 different sizes[3] of each models and with 3 different datasets [4.1] as the input. $3(sizes) \times 3(datasets) \times 2(models)$ combinations has been tested.

CNN-tiny-cifar10	ViT-tiny-cifar10
CNN-tiny-cifar100	ViT-tiny-cifar100
CNN-tiny-fashion_mnist	ViT-tiny-fashion_mnist
CNN-small-cifar10	ViT-small-cifar10
CNN-small-cifar100	ViT-small-cifar100
CNN-small-fashion_mnist	ViT-small-fashion_mnist
CNN-base-cifar10	ViT-base-cifar10
CNN-base-cifar100	ViT-base-cifar100
CNN-base-fashion_mnist	ViT-base-fashion_mnist

Table 1: All combinations ran for experiment

Size effects in Vision Transformer model The size tiny, small and base are defined by the number of layers of the transformer block involved in the training process. Tiny takes 4 iterations, small takes 6 and base takes 8. As we can see from figure [5], Surprisingly 6 iterations of the transformer block provides best test accuracy among all datasets we used in the experiment. And size tiny model gives better test accuracy than the size base model.

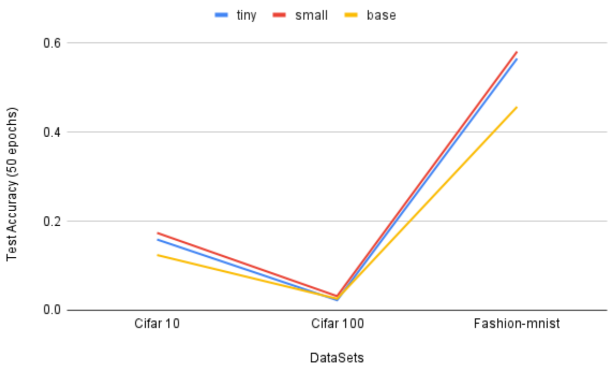

Fig. 5: Size effects in Vision Transformer model

Size effects in CNN model Also a quick comparison between different size settings of the CNN model, from figure [6], we see that size tiny and small has almost the same performance on test accuracy but with size base, the CNN model generates lower accuracy. Recall that we determined the size of the CNN model by how many convolutional layers were added to the model, the tiny model has only one convolutional layer with 64 filters and the small model will add another 64 filters conv2D layer. For the base model, the third conv2D was added and it came with 32 filters. Which is the opposite direction as normally people will increase the number of filters for each additional layer to obtain more complex features based on previous ones. So this explains why the base model with the most layers gets the worst accuracy.

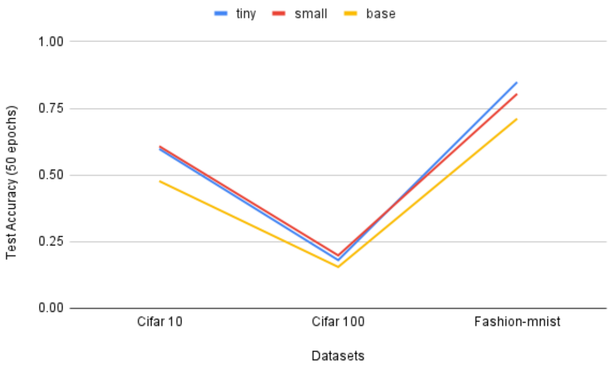

Fig. 6: Size effects in CNN model

Dataset effects in Vision Transformer model and CNN model As we learned from the paper [1], the performance of the vision transformer is highly dependent on the data size, and since all three datasets we are using here is the same size, we fix this variable while testing. From the figure [5] we can tell that the dataset CIFAR-100 gets the worst test accuracy while Fashion-MNIST gets the best test accuracy no matter what size we are using for the vision transformer model and even the same for the CNN models. This may be due to the fact the CIFAR-100 consists of way more classes than the rest datasets which causes difficulty finding the true label. And for Fashion-MNIST, this dataset contains smaller size images and they are also grayscaled, which makes them easier to classify.

CNN vs. Vision Transformer From figure [7], we see that in terms of testing accuracy, the vision transformer model never beats the classical CNN model in our experiment. But this does not mean vision transformer is not as good as CNNs for image classification, actually, as explored by paper [1], vision transformer can reach and beat the state-of-art CNN accuracy with many datasets. The result we are getting from the experiment is limited by the run-time limit, we have to limit both CNN and vision transformer models only run for 50 epochs. So all we can conclude from our experiment is, within only 50 epochs, the CNN model has an overall better test accuracy than the vision transformer.

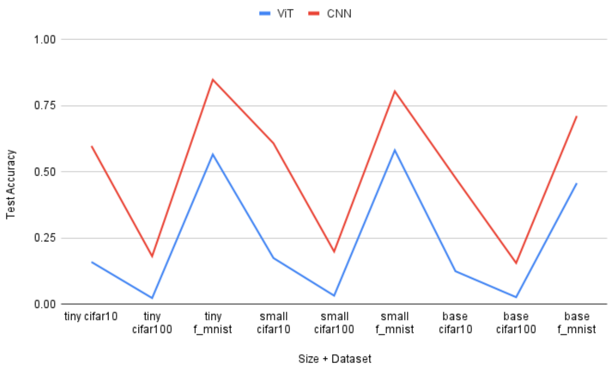

Fig. 7: CNN vs. Vision Transformer in Test Accuracy

And if we take a look at the loss value, from figure [8], we can see that when the size is tiny or small, the loss value of the CNN models is less than vision transformer models, but when the size sets as base, vision transformer models produce smaller loss values than CNNs. So in terms of loss values, our experiments show that with base size, the vision transformer works better than CNNs.

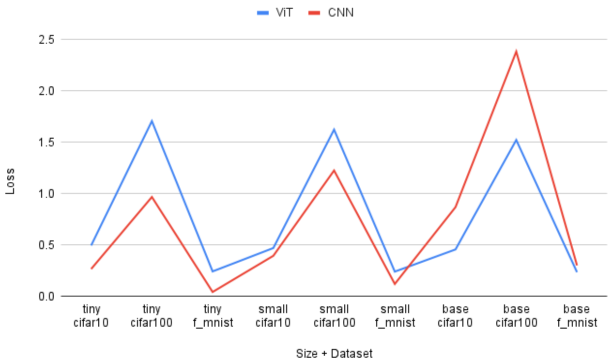

Fig. 8: CNN vs. Vision Transformer in Loss

5. CONCLUSION

In this paper, we evaluated convolutional neural network models and vision transformer models with 3 different sizes of the model and 3 different datasets - CIFAR-10, CIFAR-100, and Fashion-MNIST. From our experiment observations, we see that both CNN models and vision transformer models give better test accuracy when the size sets to tiny or small and perform relatively poorly when the size sets to base. And comparing performances between the two models, we see that in terms of test accuracy, CNN beats vision transformers in 50 epochs, but in terms of loss values, the vision transformer has a better performance when the size of the model sets to base.

6. REFERENCES

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, “Attention is all you need,” 2017.

[3] Stéphane Cuenat and Raphaël Couturier, “Convolutional neural network (cnn) vs vision transformer (vit) for digital holography,” 2021.

[4] Xingyu Liu, Yan Wu, Wenkai Liang, Yice Cao, and Ming Li, “High resolution sar image classification using global-local network structure based on vision transformer and cnn,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[5] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy, “Do vision transformers see like convolutional neural networks?,” *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[6] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi, “Understanding of a convolutional neural network,” in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.

[8] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, “Going deeper with convolutions,” 2014.

[9] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] Alex Krizhevsky, Geoffrey Hinton, et al., “Learning multiple layers of features from tiny images,” 2009.

[12] Han Xiao, Kashif Rasul, and Roland Vollgraf, “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms,” 2017.