Understanding Why ViT Trains Badly on Small Datasets: An Intuitive Perspective

Haoran Zhu*
New York University
hz1922@nyu.edu

Boyuan Chen*
New York University
boyuan.chen@nyu.edu

Carter Yang
New York University
py2097@nyu.edu

Abstract

Vision transformer (ViT) is an attention neural network architecture that is shown to be effective for computer vision tasks. However, compared to ResNet-18 with a similar number of parameters, ViT has a significantly lower evaluation accuracy when trained on small datasets. To facilitate studies in related fields, we provide a visual intuition to help understand why it is the case. We first compare the performance of the two models and confirm that ViT has less accuracy than ResNet-18 when trained on small datasets. We then interpret the results by showing attention map visualization for ViT and feature map visualization for ResNet-18. The difference is further analyzed through a representation similarity perspective. We conclude that the representation of ViT trained on small datasets is hugely different from ViT trained on large datasets, which may be the reason why the performance drops a lot on small datasets. Our code and documentation are publically available at: https://github.com/BoyuanJackChen/MiniProject2_VisTrans.

1 Introduction

Attention mechanism has become the most effective tool in natural language processing tasks. In recent years, it has been proven to perform well on computer vision tasks, such as image detection, image classification and video processing. With the advent of Visual Transformer (ViT)[1], the pure attention network began to rival the accuracy of convolutional neural networks (CNN) in image classification tasks. Further research on vision transformers will not only improve the capability of machine learning in vision tasks, but also improve our understanding on transformers and their relationship with CNN.

Nonetheless, one major drawback of vision transformer is its bad performance on small-scale datasets [2]. Traditional CNN can be trained to make high accuracy predictions on the test dataset, and their accuracy increases as we increase the number of parameters and layers. On the other hand, vision transformers usually have poor performance when trained on small datasets. Existing Methods such as Shifted Patch Tokenization (SPT) and Locality Self-Attention (LSA) are proven to improve the transformers' accuracy on small datasets[3], yet their accuracy is still lower than CNN's.

We explore the reason why vision transformers perform worse than CNN on smaller datasets. We provide visual evidence, such as attention visualization and forward propagation, and representation similarity analysis. We expect our results may contribute to the understanding of the attention mechanism on image data, as well as to inspire a new solution to improve vision transformer networks.

The contributions of our work could be summarized as follows:

*Equal contributors.

- By comparing the performance of ViT and ResNet on CIFAR-10, CIFAR-100, and SVHN datasets, we confirm that ViT does not perform well on small datasets compared with CNN.

- We conduct attention visualization on ViT and feature map visualization on ResNet to visualize the weights of each layer in each model.

- We empirically measure the representation similarity between ViT and ResNet on small datasets and compare the difference on large datasets in [4]. Unlike [4], which mainly compares the representation similarity on large datasets, we focus on analyzing differences and explore reasons for performance drop of ViT on small datasets.

2 Model and Data

We initialize a ViT model based on [1] and train it on three image datasets: CIFAR-10, CIFAR-100, and Street View House Numbers (SVHN) [5]. This is done with two primary objectives: to re-generate existing literature results on ViT for small datasets [3, 6], and to understand which kind of small dataset ViT learns less well, by comparing the evaluation accuracy.

We compare the performance of our ViT model, which has 9.6M parameters, to the performance of a standard ResNet18 model, which has 11.5M parameters. We choose the latter for comparison because it is widely used to assess model efficiency, and its parameter count is relatively close to that of ViT, when compared to other ResNet architectures.

Overall, ViT performs comparably to ResNet on SVHN, but significantly worse on CIFAR-10 and CIFAR-100. We will discuss their respective accuracy and provide evidence to support this finding in the following sections.

2.1 Dataset and Augmentation

We train models for image classification using the CIFAR-10, CIFAR-100, and SVHN. The CIFAR-10 data contains 50k training images and 10k testing images with 10 classes, each class having the same number of images for both training and testing sets. The CIFAR-100 data has the same image size, and the same volumes of training and testing dataset. The only difference from CIFAR-10 is that CIFAR-100 has 100 classes, evenly assigned to images in both training and testing sets. Therefore, the number of samples in each class is only 1/10 of that in CIFAR-10, making the training harder. The SVHN dataset contains 600k images of digits of house numbers, and each label is the digit that image shows. All three datasets have images of 32×32 pixels in three channels of color. The unification of this factor eliminates the possible difference in outcome based on image size.

We introduce data augmentation methods for image classification tasks including flipping and cropping. For the training set, we cropped the input image at a random location in 32×32 pixels with a padding of 4, and randomly flip the image horizontally with the probability of 0.5, which is applied to all types of datasets.

For both ResNet and ViT, we did not implement normalization on pixel values.

2.2 Model Architecture

For the ResNet, we implement the standard ResNet-18 architecture, as it is widely used for comparison in many works on image classification. Each residual block has 2 convolutional layers, with three expansions at a rate of 4 every two residual blocks.

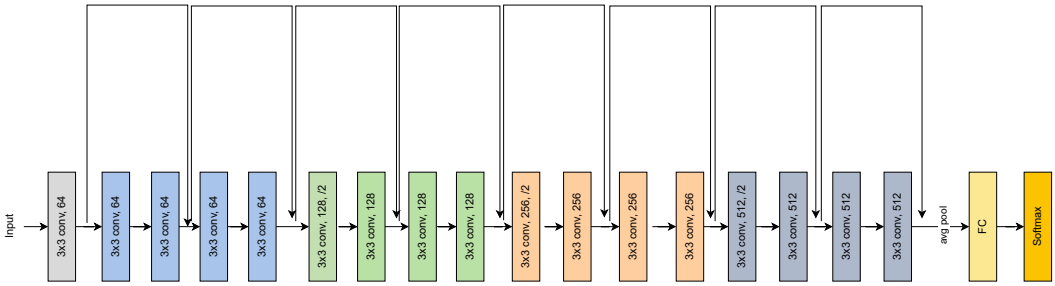


Figure 1: ResNet-18 architecture.

For Vision Transformer (ViT), we divide the image into 4 batches. Each attention layer has 8 heads, each having a dimension of 64. The transformer encoder has a depth of 6, and a drop-out rate of 0.1. Finally, the MLP layer has a dimension of 512, and a drop-out rate of 0.1.

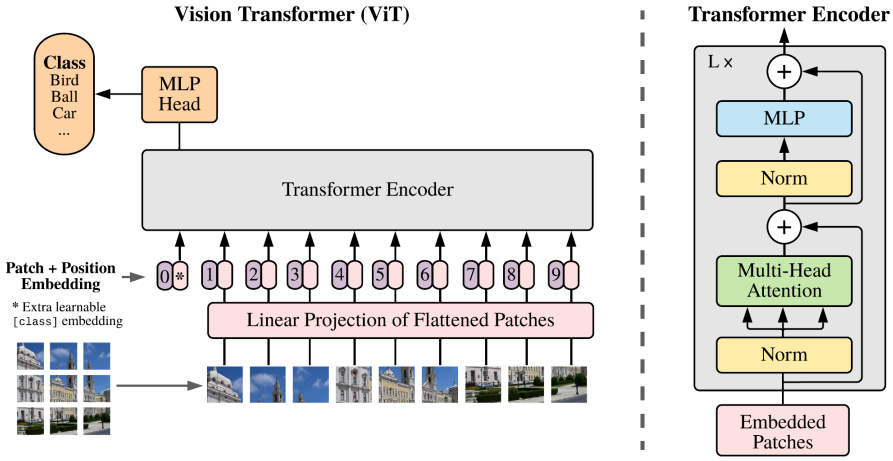


Figure 2: ViT architecture, image from [1]. In our experiments, the transformer encoder has L=6 layers.

2.3 Training Details

We train two models: ResNet-18, and Vision Transformer, on 3 different datasets: cifar10, cifar100, and svhn. To make it fair, all of the hyper-parameters are kept the same such as learning rate = 1e-4, batch size = 100, and using adam optimizer. We ran each experiment for 500 epochs. and use wandb (Weights and Biases) [7] library to track and visualize the results. The built-in visualization features in wandb provide multiple plots of metrics mainly about train/test loss and accuracy, allowing us to compare across different models with the same dataset.

3 Model Accuracy

Table 1 shows the performance of ViT compared with ResNet18 on CIFAR-10, CIFAR-100 and SVHN dataset after training for 500 epochs. Figure 3a-3c show the accuracy testing curve during training. We can see that ViT performs significantly worse on CIFAR-10 and CIFAR-100 compared to ResNet18. The error rate of the former is twice of the latter. Nonetheless, ViT performs equally well on SVHN, a colored dataset on digit recognition, though its convergence is slower than ResNet from Figure 3c.

This result confirms the assumption that ViT performs worse on small datasets. ViT achieves a similar result on SVHN because of the simplicity of the dataset. In general, ViT also performs well on MNIST, which is a one-channel version of digit recognition. If the model can fit well on one channel, then it is likely to also fit well on three channels.

	CIFAR-10	CIFAR-100	SVHN
ViT	81.36	54.31	95.17
ResNet18	**92.8**	**70.7**	**95.78**

Table 1: Top-1 accuracy(%) of ViT and ResNet18, trained from scratch on different small datasets (500 epochs).

4 Visualization of Layers

In this section, we gain intuitions about what each layer of a model does by using attention weights extraction for ViT model and feature mapping for ResNet. Unfortunately, the two visualizations are not directly comparable, as they use vastly totally different learning strategies. However, the visualization tools can still help us understand the logic behind the black box of parameters.

4.1 Attention Weights for ViT

Attention weights visualization is proposed in the original ViT paper [1] to demonstrate how the model processes image data. As the method instructs, we first extract the attention layer with shape $(n_{heads}, n_{patch}+1, n_{patch}+1)$ in each transformer block. Then we average the weight across the heads and then add with an identity matrix to account for residual connection. We then normalize and reshape the matrix to form a weight mask. To facilitate visualization, each weight mask is scaled by a common factor so that the max weight of all masks is equal to 1. Finally, each weight mask overlaps on top of the original image. The bright areas receive more attention, and the darker areas receive less attention.

While the original work only shows the weight mask for the first attention layer, here we provide the visualization of all the 6 layers' attention weights in Figure 4. The first layer exhibits concentrated attention on a small area, while later layers expand their attention to the whole image. Our experiments show that the first layer tends to put more weight on areas with higher contrast in the neighborhood, and future layers expand their attention from the previous layer.

Based on the visualizations, we speculate that the ViT model is trained to put more attention on regions with higher local contrast. While this strategy works well on most pictures in CIFAR-10 dataset, there are also images where this strategy does not perform well. Such examples can be found in the Appendix.

4.2 Feature Map Visualization for ResNet

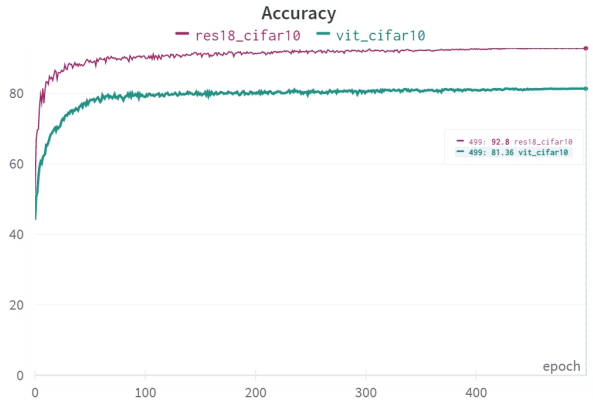
A feature map is the output of a convolution layer in a CNN network. The values across channels are averaged, so we can directly visualize the output in grayscale. For a ResNet18 network, we can generate 17 feature maps. For dimensions, layers 1-5 are of size $(32, 32)$; layers 6-9 are of size $(16, 16)$; layers 10-13 are of size $(8, 8)$; layers 14-17 are of size $(4, 4)$.

Figure 5 shows the feature maps of the model forwarding the same images as we used for ViT.

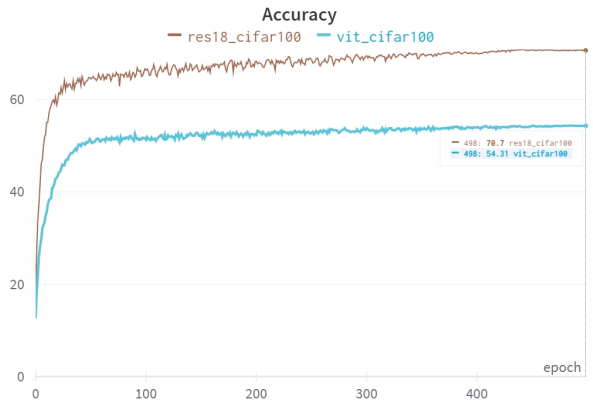
5 Representation Similarity Analysis

After confirming our hypothesis that ViT performs less well compared to ResNet on small image datasets, we next try to provide an intuitive explanation on ViT's behavior when trained on a small dataset. We use **CKA (Centered Kernel Alignment)**[4] to analyze representation similarity between ViT and ResNet:

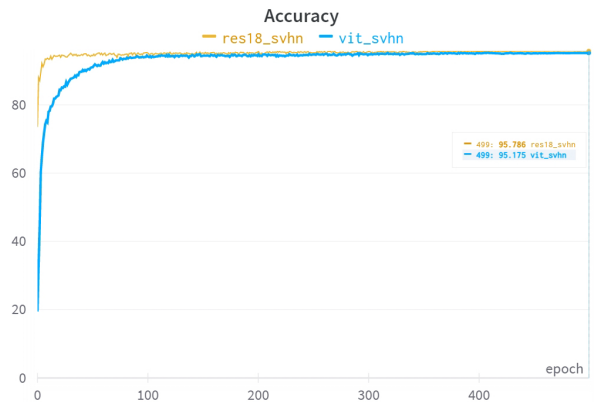
$$\mathrm{CKA}(\boldsymbol{K}, \boldsymbol{L}) = \frac{\mathrm{HSIC}(\boldsymbol{K}, \boldsymbol{L})}{\sqrt{\mathrm{HSIC}(\boldsymbol{K}, \boldsymbol{K})\,\mathrm{HSIC}(\boldsymbol{L}, \boldsymbol{L})}}$$



(a) CIFAR-10 accuracy



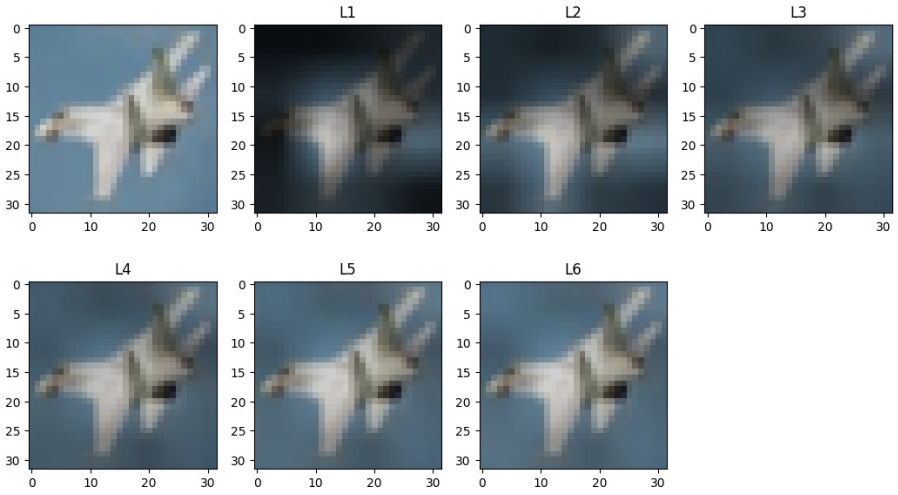
(b) CIFAR-100 accuracy



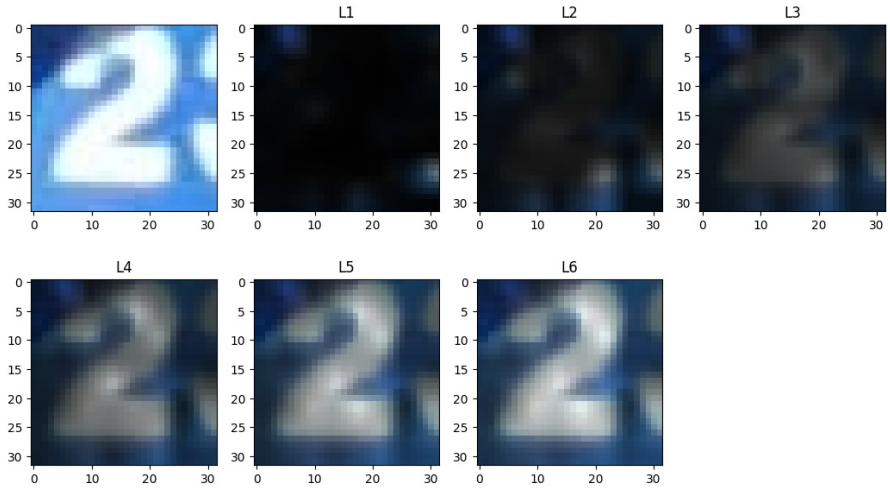
(c) SVHN accuracy

Figure 3: Accuracy in 500 epoch

where $\mathbf{X} \in \mathbb{R}^{m \times p_1}$ and $\mathbf{Y} \in \mathbb{R}^{m \times p_2}$ are representations of two layers with p_1 and p_2, $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top$ and $\boldsymbol{L} = \boldsymbol{Y}\boldsymbol{Y}^\top$ denote the Gram matrix for two layers, and HSIC is the Hilbert-Schmidt independence criterion [8]. In general, when the CKA value between two layers are high, the representations of these two layers are much similar. Using this metric, we can analyze the representation difference when ViT faces small datasets. Unlike [4] (See Figure 6), we focus on comparing the difference and giving interpretations on small datasets, which is novel. (See Figure 7a, Figure 7b and Figure 7c)



(a) Image of airplane from CIFAR-10 dataset.

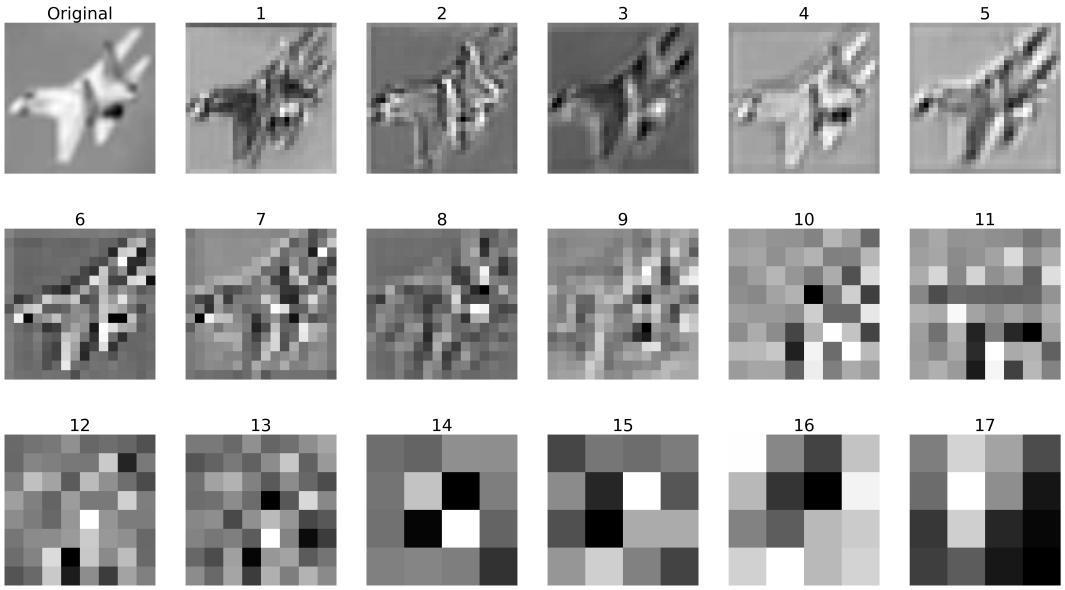


(b) Image of number 2 from SVHN dataset.

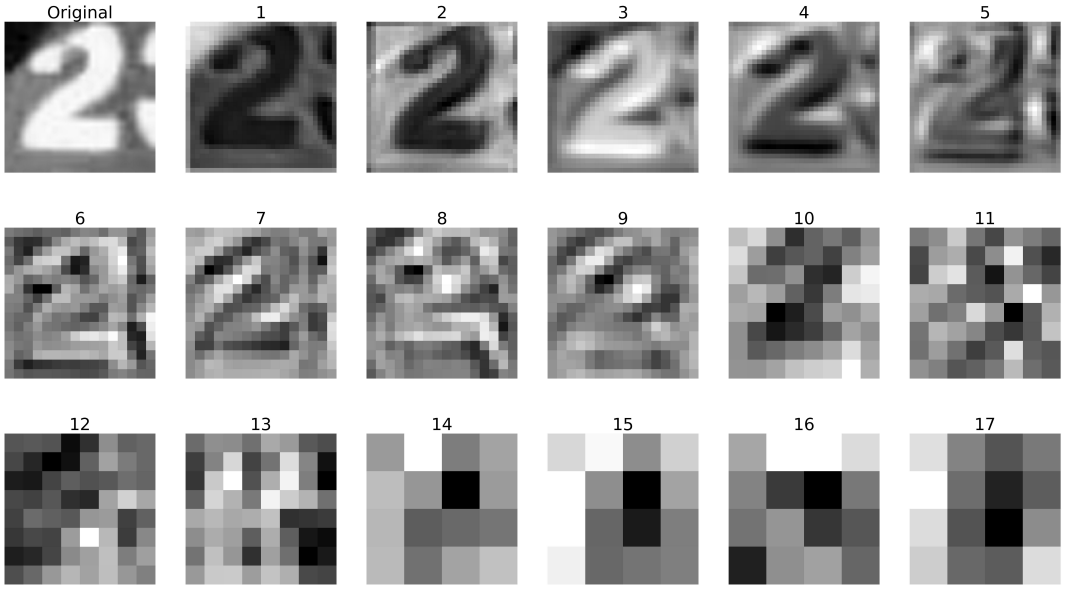
Figure 4: Attention weight visualization from ViT. The top two rows show the model trained on CIFAR-10, forwarding an image of an airplane; the lower two show the same model trained on SHVN, forwarding an image of number 2. The input is the 10th image from CIFAR-10, with the label of an airplane. The original image is shown on the top-left. The rest shows the original image overlapped with the weight mask of each attention layer. Bright regions represent higher attention weights (close to 1); darker regions represent lower attention weights (close to 0).

Note that when computing the representation similarity, we not only compare the convolution layers and attention layers, we also compare all the normalization and activation layers. By comparing Figure 6 (from [4], computed on large dataset) with Figure 7a, 7b and 7c (computed on small datasets by us), we observe a huge change of representation power of ViT when trained from large datasets to huge datasets. From Figure 6, when trained with large datasets, almost the lower half of ResNet layers have a similar representation of the lowest quarter of layers in ViT; the latter half of ResNet layers have a similar representation of the next quarter of layers in ViT; the highest quarter of layers are dissimilar with all layers of ResNet.

However, when trained on small datasets, the patterns of representation similarity change a lot. By comparing the representation similarity and observing the visualizations in the previous section, we observe:



(a) Image of number 2 from SVHN dataset.



(b) Image of number 2 from SVHN dataset.

Figure 5: Feature map visualization from lower layers to higher layers in ResNet18. The top three rows show the model train on CIFAR-10 forwarding an airplane image; the lower three rows show the same model train on SHVN forwarding an image of number 2. The top-left is the original image in grayscale; from left to right top to bottom, we exhibit feature maps of convolution layer 1 to convolution layer 17.

Figure 7a and Figure 7b change a lot, we completly lose the pattern on large datasets of Figure 6. Which means the representation on CIFAR-10 and CIFAR-100 is completely different with large datasets and causes the huge drop on final performance.

Figure 7c is most similar to Figure 6, we can still observe the lower layers of ResNet have similar representation of lower layers of ViT, higher layers of ResNet have similar representaion of middle-to-higher layers of ViT and highest layers of ViT are dissimilar with all layers of ResNet. However,

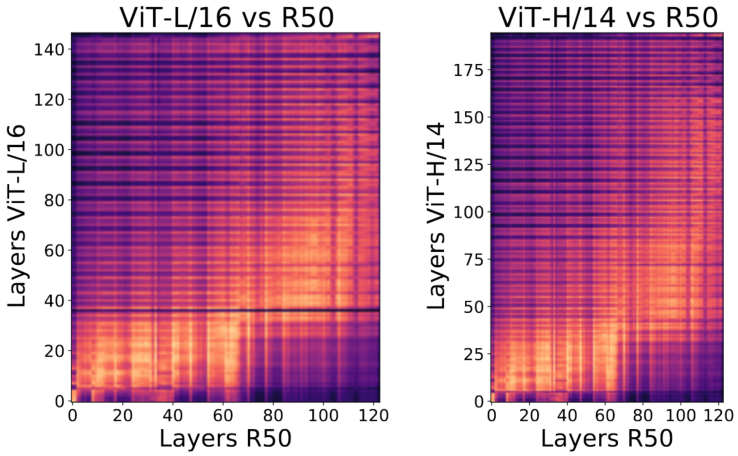
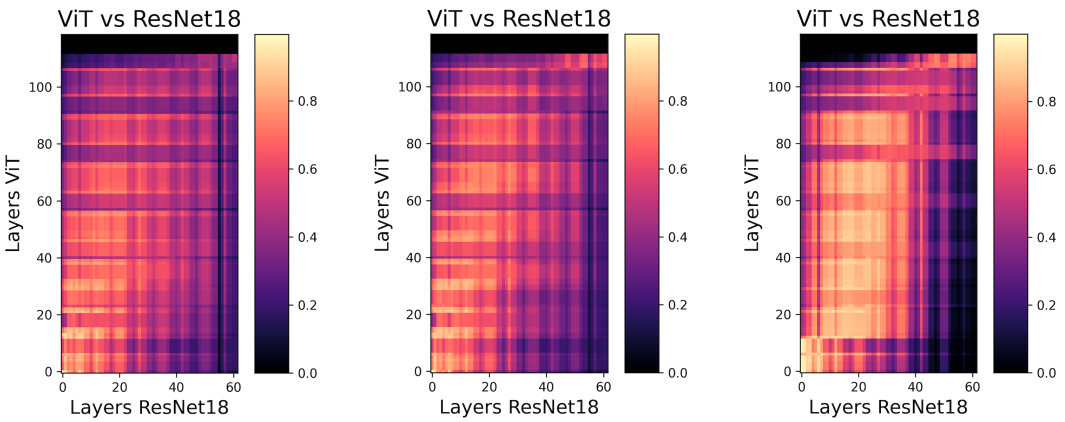


Figure 6: Figure from [4] representation similarity between ViT and ResNet on large datasets (JFT-300M)



(a) Representation similarity between ViT and ResNet on CIFAR-100

(b) Representation similarity between ViT and ResNet on CIFAR-10

(c) Representation similarity between ViT and ResNet on SVHN

Figure 7: Representation similarity between ViT and ResNet on different datasets

in this case, ViT needs more layers to get the same representation of ResNet, compared with less layers before. From Figure 4, lower layers of ViT is more focusing on local areas on SVHN dataset, which means that ViT can learn more locality on SVHN compared with CIFAR-10 and CIFAR-100. The reason might be that SVHN is a simpler dataset, thus ViT can catch the inductive bias of locality on this simple dataset and that can explain the reason why the performance of ViT is similar to ResNet on SVHN while it loses a lot on CIFAR-10 and CIFAR-100 in Table 1.

6 Conclusion

In this project, we would like to explore the reason why vision transformer does not perform well on small datasets. We firstly conduct extensive experiments to confirm the phenomenon of performance drop of ViT on small datasets. We later interpret the results by showing attention visualization and feature map visualization. Next, we conduct representation similarity analysis to further investigate the results. Finally, by comparing the difference between attention map visualization and representation similarity. We can speculate the reason for the performance drop of ViT on small datasets as follows:

- When trained with small datasets, the representation of ViT is hugely different from ViT trained with large datasets and thus affects the performance a lot.
- The huge change of representation may be due to a lack of inductive bias of locality for ViT. Lower layers of ViT can not learn the local relations well with a small amount of data on complicated small datasets, e.g., CIFAR-10 and CIFAR-100. For simpler datasets, e.g., SVHN, ViT can learn locality relatively well, it's reflected on feature map visualization and might be the reason that ViT can achieve worse but similar performance on the SVHN dataset.

7 Acknowledgements

Our code for visual transformer model construction and training is adapted from:

- https://github.com/kentaroy47/vision-transformers-cifar10

CKA Similarity Analysis is adapted from:

- https://github.com/AntixK/PyTorch-Model-Compare

ViT attention weights extraction and visualization are adapted from:

- https://github.com/jeonsworld/ViT-pytorch/blob/main/visualize_attention_map.ipynb

ResNet feature map visualization is adapted from:

- https://ravivaishnav20.medium.com/visualizing-feature-maps-using-pytorch-12a48cd1e573

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.

[2] Safwen Naimi, Rien van Leeuwen, Wided Souidene, and Slim Ben Saoud. Hybrid byol-vit: Efficient approach to deal with small datasets. *arXiv preprint arXiv:2111.04845*, 2021.

[3] Seung Hoon Lee, Seunghyun Lee, and Byung Cheol Song. Vision transformer for small-size datasets. *CoRR*, abs/2112.13492, 2021.

[4] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[5] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[6] Yahui Liu, Enver Sangineto, Wei Bi, Nicu Sebe, Bruno Lepri, and Marco Nadai. Efficient training of visual transformers with small datasets. *Advances in Neural Information Processing Systems*, 34, 2021.

[7] Lukas Biewald. Experiment tracking with weights and biases. *Software available from wandb.com*, 2, 2020.

[8] Arthur Gretton, Kenji Fukumizu, Choon Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A kernel statistical test of independence. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

Appendix

Additional Visualization Results

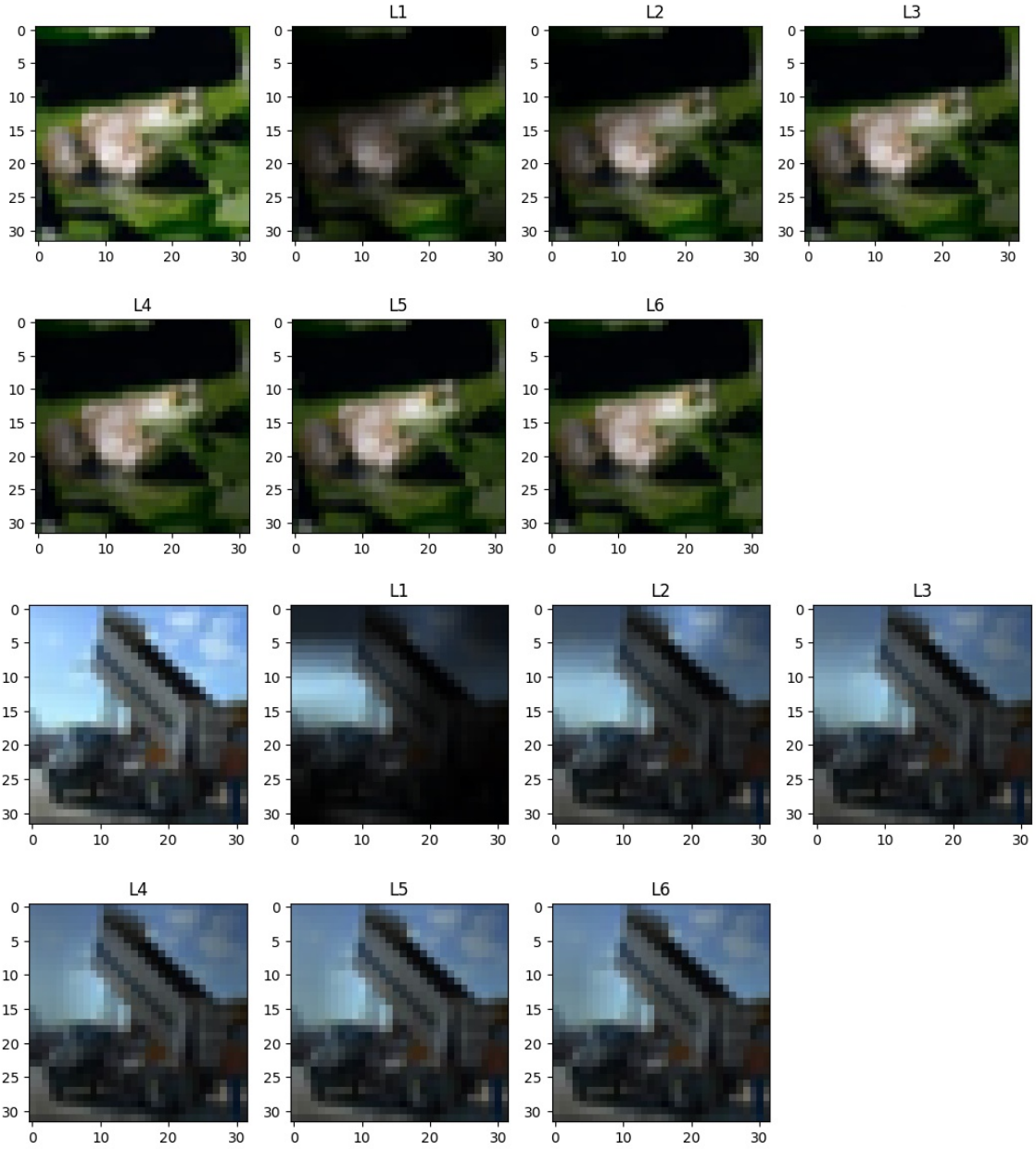


Figure 8: Visualization for ViT on CIFAR-10 dataset.