DeCAtt: Efficient Vision Transformers with Decorrelated Attention Heads

Mayukh Bhattacharyya*
Stony Brook University

Soumitri Chattopadhyay*
Jadavpur University

Sayan Nag*
University of Toronto

Abstract

The advent of Vision Transformers (ViT) has led to significant performance gains across various computer vision tasks over the last few years, surpassing the de facto standard CNN architectures. However, most of the prominent variations of Vision Transformers are resource-intensive architectures with huge parameter sizes. They are known to be data-hungry and overfit quickly on comparatively smaller datasets. Consequently, this holds back their widespread usage across low-resource settings, which brings forth the need to develop resource-efficient vision transformers. To this end, we introduce a regularization loss that prioritizes efficient utilization of model parameters by decorrelating the heads of a multi-headed attention block in a vision transformer. This forces the heads to learn distinct features rather than focus on the same ones. Using this loss provides a consistent performance improvement over a wide range of varying scenarios of models and datasets as we show in our experiments, which proves its superior effectiveness.

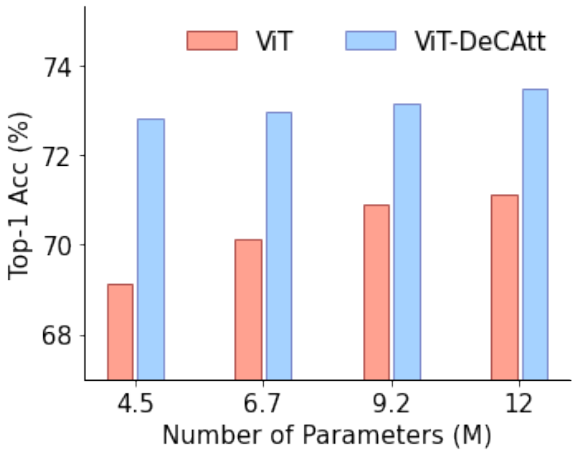

Figure 1. Performance Comparison between ViT and proposed ViT-DeCAtt with respect to Model Parameters (in Millions) on the Oxford Flowers dataset. Decorrelating attention heads leads to significant boost in model performance and similar accuracy can be achieved with approximately **2.5 to 3 times lesser** parameters.

1. Introduction

Vision transformers (ViT) [5] have developed to be the new standard in computer vision tasks ranging from recognition to even serving as encoders for semantic segmentation, object detection and multiple other downstream tasks. It has been able to beat benchmarks of complex CNN models using significantly less training times. Although a lot of research work has been done in the past few years on the applications of vision transformers in various tasks, comparatively less focus has been given to optimizing the existing architectures, which are highly resource exhaustive. Dosovitskiy et al. [5] introduced the Vision Transformer variants ViT-B, ViT-H and ViT-L which are comparatively large in terms of parameter size. The ViT-Base is itself 86M parameters in size which needs thorough pre-training on large image corpuses to achieve its best performance. On the contrary, while training on a smaller scale dataset, it runs into problems of overfitting which require strong regularizing to overcome. This leaves us with the question of whether we can improve the model efficiency by taking advantage of the transformer architecture other than relying on just traditional regularizers like Dropout, L2 etc. In order to achieve better utilization of parameters, we need to ensure that each segment of the model learns something unique or in other words, is uncorrelated to other segments. Upon inspection, we find that the multi-head attention framework is something that may lead to redundancies with different attention heads learning vastly similar features. Alleviating this is something we believe can greatly improve our training efficiency and we will be focusing on this in our work.

The main contributions of our work are as follows:

1. We introduce a loss paradigm that minimizes the cross-correlation among the heads of each layer of vision transformers. This loss acts as a regularizer that mitigates overfitting as well as improves efficiency of the model as a whole.

2. We demonstrate that introducing this auxiliary loss helps models achieve superior empirical performance on standard vision datasets. Furthermore, we show that

*all authors contributed equally.

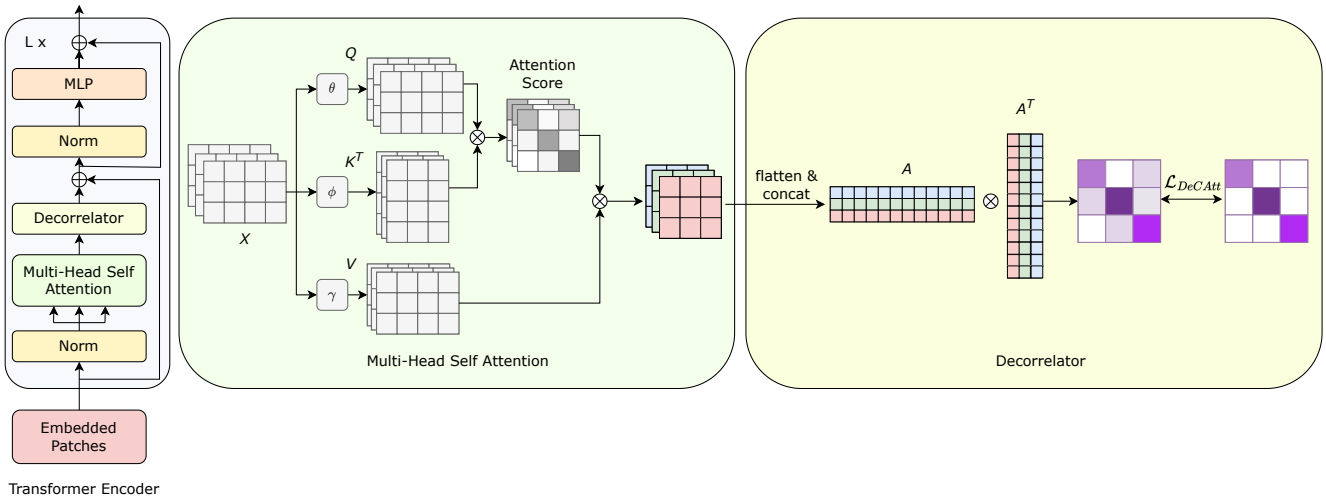

Figure 2. Vision Transformer with proposed **DeC**orrelated **Att**ention Heads and corresponding decorrelation loss $\mathcal{L}_{DeCAtt}$. Multi-head self attention maps are flattened and concatenated in the Decorrelator block (yellow) and the matrix is denoted as A. This matrix A is then multiplied with its transpose and normalized to obtain the cross-correlation matrix. $\mathcal{L}_{DeCAtt}$ objective function tries to make the off-diagonal elements of this matrix close to zero. This causes the attention heads to learn distinct features while minimizing the redundancy between them.

ViT trained with proposed loss can achieve similar or better performance with approximately 2.5 to 3 times lesser parameters (as illustrated in the Figure 1).

3. Through ablation studies, we study different aspects of this loss pertaining to the location of its usage, weightage and its impact on models across different sizes.

2. Related Works

Vision Transformers: ViTs, introduced in [5] as a modification of traditional transformers [14] for the domain of computer vision, have proven to be superior over CNN-based networks. Since then many works have explored various aspects of this particular architecture, leading to the introduction of various modified versions including several lightweight architectures [2,3,8,11,13,15]. In particular, we take some interest in DeiT [13] in which they introduced a few lightweight transformer models which serve as an inspiration for the models we have used in our case.

Decorrelation Loss: Although correlation as a loss has not been explored in the Transformer architecture, it has been used in the past in traditional neural networks in some scenarios like regularization and encoding style [1, 6]. One of the first works to use decorrelation in neural networks as a regularizer was [4] where the authors applied it to the fully connected layers of a Multi Layer Perceptron to improve the performance of models. It showed early promising signs of using this technique as a method to prevent overfitting. Other works followed suit which tried to incorporate similar ideas in convolutional neural networks. Works such as [7] explored decorrelation within the convolutional domain, albeit in a different manner.

3. Methodology

Our final goal is to reduce the amount of correlated features in the multi-headed attention layer. Although this can be achieved to some varied degree by varying the location of application, we choose the output from each head as the items to decorrelate.

Let A be the unrolled matrix of each attention map of heads with a dimension of $B \times h \times (nd)$ where B is the batch size, h is the number of heads, n is the number of patches and d is the dimension of query, key and value vectors. Then we obtain our loss through the following equations.

$$C_1 = \frac{AA^T}{|A|^2} \tag{1}$$

$$C_2[...] = \frac{1}{B}\sum_{i=0}^{B}(C_1[..., i])^2 \tag{2}$$

From C_2 we obtain our decorrelation loss (Equation 3) by summing up the non-diagonal elements. We take only the non-diagonal elements as we are only concerned with cross-correlations between heads and auto-correlations are of no value to us.

Model	Heads per Layer	Num. Params	Top-1% Acc.			Avg. Relative Train time per Epoch		
			CIFAR10	CIFAR100	Flowers	CIFAR10	CIFAR100	Flowers
ViT	3	4.5M	80.24	52.08	69.14	1.0	1.0	1.0
ViT-DeCAtt	3	4.5M	81.85	54.61	72.80	1.04	1.03	1.09
ViT	6	12M	82.48	55.36	71.12	1.0	1.0	1.0
ViT-DeCAtt	6	12M	83.16	57.76	73.49	1.06	1.09	1.05

Table 1. Evaluation accuracy in %. ViT is the lightweight model described in experiments section. ViT-DeCAtt is the same model with our decorrelation loss.

$$\mathcal{L}_{DeCAtt} = \sum_{i=0}^{N}\sum_{j=0}^{N} C_2[i,j](i \neq j) \tag{3}$$

Therefore, the following Equation 4 gives us the total loss (for classification task) that we will try to optimize:

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda\mathcal{L}_{DeCAtt} \tag{4}$$

where λ is the decorrelation weight factor and $\mathcal{L}_{CE}$ is the Cross-Entropy (CE) loss.

4. Experiments

Datasets: We have evaluated our methods on image recognition tasks on the CIFAR10, CIFAR100 and Oxford Flowers datasets. CIFAR 10 and 100 comprise of 32x32 images whereas Flowers consists of images which are 224x224.

Model configuration: or the experiments, we use two variations of a lightweight vision transformer which are influenced by the $Tiny$ variant suggested in [13]. The ViTs in use had a depth of 12, an embedding dimension of 64 per head and a final MLP dimension of 512. We explore results on two variations of the model in terms of the number of heads 3 and 6. This is important in order to view the impact of the increase of parameters and the implications of overfitting. The DeCAtt loss is applied on the first 3 layers of the ViT in all cases (unless otherwise specified). We have chosen $\lambda = 150$ by conducting a hyperparameter search. Also, please note that the ViTs are trained from scratch *without any pre-trained weights*. This is to showcase the true effect of the loss in a training scenario.

For robustness of results, all the experiments were run with multiple seeds for 100 epochs or till convergence whichever is higher, on a Nvidia A100 with Adam optimizer [9] and some light augmentations. The results from the experiments are given in Table 1.

4.1. Performance Analysis

It can be seen from the experimental results that the Decorrelation Loss leads to $2-5\%$ improvement in performance. From the results, it can be seen that we obtain similar kind of performance from a model with 3x less parameters when using the DeCAtt loss. The percentage of performance improvement is slightly more for a lighter model. It means DeCAtt loss will have a slightly greater effect when a lighter model is trying to learn complex data. This can be seen as a *bias-reducing trait*, which is somewhat in contradiction to the principles of a regularizer and also goes beyond the benefits achieved by [4] using similar techniques. It can be inferred that involving DeCAtt loss may have a beneficial effect in terms of reducing both bias and variance of a ViT model. The extent of both of these effects is something that we need to explore in greater detail.

Another aspect of involving any auxiliary loss is the small computation overhead that is added to training. We have included this in the results to provide a fair indication of the overall cost. The setup for this is the same as mentioned in the experiment section. We can see that we are taking only around 5-10% extra time which is an acceptable scenario in most cases.

4.2. Ablation Studies

To understand the different aspects of the DeCAtt loss, we perform a set of ablation studies involving different tunable aspects of model training and analyze their impacts on the overall model performance.

Number of heads: In a transformer layer, the number of heads plays an important role in the final model performance. We apply $\mathcal{L}_{DeCAtt}$ on these attention heads in order to force them to learn distinct features and reduce redundancy. We have considered the first 3 layers of the ViT model for applying the aforementioned decorrelation loss while varying the number of attention heads in the transformer. Figure 3 shows the impact of heads on model performance when trained and subsequently evaluated on the Flowers dataset. With our loss, it is possible to get significantly superior performance as compared to vanilla ViT with as less as 3 heads. This demonstrates that proposed decorrelation loss indeed helps in reducing feature redundancy whilst boosting overall performance.

Position of employing DeCAtt: We assessed the impact

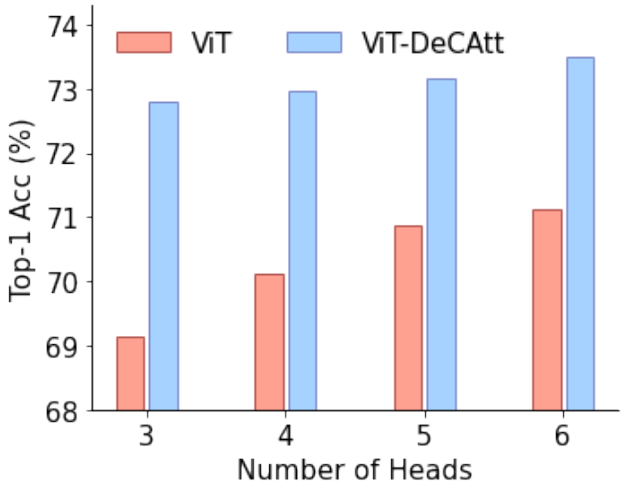

Figure 3. Top-1 % Accuracy (on Oxford Flowers) of ViT and proposed ViT-DeCAtt with respect to number of heads in a transformer layer.

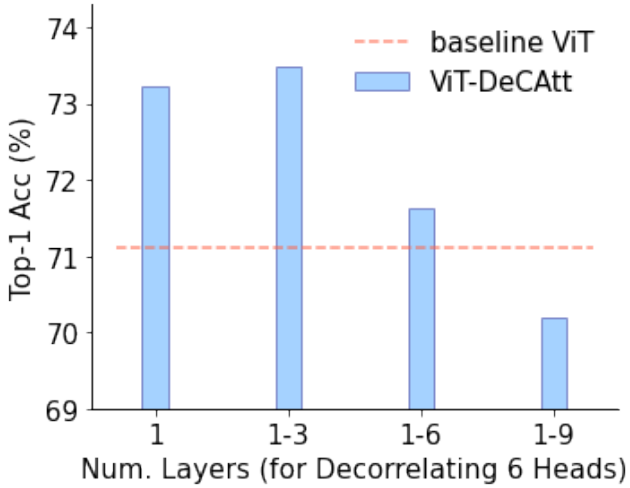

Figure 4. Top-1 % Accuracy (on Oxford Flowers) of ViT and proposed ViT-DeCAtt with respect to the number of transformer layers where decorrelation loss has been applied.

of different transformer layers (where decorrelation loss is applied) by fixing the number of heads to 6, and training and consequently evaluating the models on the Flowers dataset. It is to be noted that the transformer taken into consideration has 12 layers. Therefore, we have conducted 4 separate cases where we have applied decorrelation loss to: (i) 1^{st} layer, (ii) 1^{st} to 3^{rd} layers, (iii) 1^{st} to 6^{th} layers, and finally, (iv) 1^{st} to 9^{th} layers. We observe that when the first 3 layers are considered model performance increases, however, it significantly drops with further consideration of more layers as shown in Figure 4. Thus, we can infer that decorrelation is most impactful in the initial layers of the model where it can help different heads learn distinct features, whereas further involvement of layers negatively impacts the model performance.

Dropout	L2 Reg	DeCAtt	Top-1 Acc.
×	×	×	44.23
✓	×	×	45.32
×	✓	×	53.58
×	×	✓	49.21
✓	✓	×	55.36
✓	×	✓	50.27
×	✓	✓	56.92
✓	✓	✓	**57.76**

Table 2. Top-1 % Accuracy values obtained by using different regularization methods on CIFAR-100.

Different regularization methods: We conducted a comparative study of DeCAtt with two other regularizers Dropout [12] and L2 [10] Regularization to understand and compare the effectiveness of these in ViT. The results presented in Table 2 show that L2 regularization is the most effective among the 3 followed by DeCAtt. The combination of DeCAtt with any other form of regularization is always reflected in a further increase in performance and thus the best performance is obtained as a combination of all three. Here, we used 150.0 as λ for DeCAtt, 0.2 as dropout and $1e-4$ as L2 weight.

Computational efficiency: Application of $\mathcal{L}_{DeCAtt}$ during training leads to a small increase in training time (see Table 1) because of an extra loss overhead. However, the number of parameters in the model remains unchanged. Rather, it can be observed from Table 1 and Figure 1 that similar or better performance is achieved with 2 to 3 times fewer parameters (e.g., Top-1 % Accuracy of ViT with 6 heads per layer, i.e., 12M parameters, is 71.12, and that of ViT-DeCAtt with 3 heads per layer, i.e., 4.5M parameters, is 72.80, for the Flowers dataset).

5. Conclusion and Future Work

In this study, we have provided a baseline of using decorrelation of attention heads as a way to regularize lightweight ViT-based networks and make them more efficient. As a preliminary work, this provides a strong base on which further explorations in this direction can be undertaken, especially related to how this adjusts to bigger models. The behaviour of bigger models like ViT-Base and ViT-Large might be different which may require further variations or tweaks of the DeCAtt loss to make them beneficial to them on large datasets.

References

[1] M. Bhattacharyya and S. Nag. Hybrid style siamese network: Incorporating style loss in complimentary apparels retrieval. *CoRR*, abs/1912.05014, 2019. 2

[2] C.-F. R. Chen, Q. Fan, and R. Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *ICCV*, 2021. 2

[3] Y. Chen, X. Dai, D. Chen, M. Liu, X. Dong, L. Yuan, and Z. Liu. Mobile-former: Bridging mobilenet and transformer. In *CVPR*, 2022. 2

[4] M. Cogswell, F. Ahmed, R. B. Girshick, C. L. Zitnick, and D. Batra. Reducing overfitting in deep networks by decorrelating representations. *CoRR*, abs/1511.06068, 2016. 2, 3

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1, 2

[6] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015. 2

[7] S. Gu, Y. Hou, L. Zhang, and Y. Zhang. Regularizing deep neural networks with an ensemble-based decorrelation method. In *IJCAI*, 2018. 2

[8] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh. Rethinking spatial dimensions of vision transformers. In *ICCV*, 2021. 2

[9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[10] A. Krogh and J. Hertz. A simple weight decay can improve generalization. In *NeurIPS*, 1991. 4

[11] S. Mehta and M. Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *ICLR*, 2022. 2

[12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014. 4

[13] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 2, 3

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. 2

[15] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z.-H. Jiang, F. E. Tay, J. Feng, and S. Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *ICCV*, 2021. 2