Powerful SeqAttention for Compact Convolutional Transformer

Student Hwasik Jeong
Professor Jongbin Ryu

Abstract

As convolution and transformer architectures have advanced, performance on classification tasks has steadily improved, even surpassing human capabilities. During this development, the limitation of transformers being effective only with large datasets was addressed by the introduction of CCT, a hybrid version of convolution and transformer architectures. CCT opened up the possibility for transformers to perform well on small datasets. The sequence pool used in CCT achieved 76.93% top-1 accuracy on the CIFAR-100 dataset by learning various features, but it was found to lack feature diversity to further enhance performance. This study found out that gramian attention is much more effective at learning diverse features compared to the original sequence pooling, achieving 80.52% on the CIFAR-100 dataset with 8 heads. Taking this a step further, this study proposes a new head architecture called SeqAttention, which is much more lightweight and powerful compared to the gramian attention head. Implemented code can be found here: `https://github.com/JeongHwaSik/Powerful-Sequence-Pooling-for-CCT.git`

1. Introduction

The advancement of deep convolutional neural networks (DCNNs) in the field of computer vision has led to a series of breakthroughs in image classification [9, 18, 4]. LeNet [10] provided the first baseline on how convolutional filters could be used in deep learning, and AlexNet [9] advanced this baseline by achieving the best performance (83.6%) at ILSVRC 2012 with a model consisting of a total of 8 layers, including 5 convolutional layers and 3 fully connected layers. ZFNet [27], by adjusting the parameters of the convolutional layers in the AlexNet model, achieved even better performance (88.3%) in ImageNet classification with further enhanced visualization technique. The VGG [18] model established design rules for convolutional filters, setting a standard for making DCNNs even deeper. This resulted in models with 16 and 19 layers, which more than doubled the depth of the previous 8-layer models and broke through the 90% accuracy barrier with a 92.7% accuracy at ILSVRC 2014. As the size of the models increased, the performance in classification gradually improved, leading to more research, and it became widely accepted that increasing the model size by stacking more layers led to better classification performance in DCNNs. However, as models grew in size, they faced significant challenges, such as gradient vanishing problems and optimization difficulties, which were addressed by the introduction of residual connections in ResNet [4]. ResNet proposed a model with 152 layers, from 8 to 10 times deeper than previous models with only tens of layers, and achieved a revolutionary top-1 accuracy of 96.4% in ImageNet classification.

While research focusing on improving performance by deepening layers in DCNNs was the general trend at the time, there were also various studies that focused more on the efficiency of the network based on the size-performance tradeoff. GoogLeNet [19] proposed the Inception module and global average pooling, focusing on improving computational efficiency relative to the depth of the model. ResNeXt [25] introduced an architecture that achieved better performance than ResNet [4] within restricted computational complexity and model size. EfficientNet [20] proposed an efficient model by optimally balancing the model's depth, width, and input resolution. Additionally, lightweight models such as MobileNet [6] and MobileNetV2 [17] were developed to enable the development of deep learning models on mobile devices.

As deep convolutional neural networks were being researched from various angles in the field of computer vision, the field of natural language processing (NLP) underwent a revolutionary change with the advent of the Transformer architecture [22]. Consequently, there have been various efforts to expand Transformers into the field of computer vision by proposing new architectures to replace convolutional networks [15, 23]. Among these, the introduction of Vision Transformer [2], also called ViT, demonstrated that transformer-only architecture could be successfully applied to computer vision, showing the potential to replace convolutional operators. However, while there have been various efforts to further advance ViT [21, 24, 11, 29], its lack of inductive bias was a clear limitation for training on small datasets.

To overcome the limitation that ViT performs well only on huge datasets, an architecture called CCT [3] was proposed to transfer the inductive bias of convolutional networks to transformers. CCT replaces the patch embedding layer of general ViT with a convolutional layer and uses a method called sequence pooling instead of class token to build the classifier. However, sequence pooling does not fully capture all the representations needed for more precise classification tasks, leaving some aspects unaddressed. This study aims to find those unsolved features of sequence pooling in CCT without any further methods like pre-training with huge amounts of data or knowledge distillation [5] by analyzing the gramian attention head [16] and applying it on the CCT. Then, this study proposes a new head architecture called 'SeqAttention', which surpasses the performance of the gramian attention head in CCT with much less parameters.

2. Related Works

2.1. Transformer for Small Dataset

CCT [3] stands for Compact Convolutional Transformer, designed to allow small ViTs to be trained on small datasets while outperforming CNNs. To achieve a smaller ViT, CCT uses 7 transformer encoders, compared to the 12 used in the base ViT-B-16 [2]. Instead of the patch embedding layer found in standard ViTs, CCT incorporates one or two CNN layers to transfer inductive bias to the transformer encoder. While standard ViTs employ learnable class tokens in the classifier for classification tasks, CCT utilizes sequence pooling to leverage the output features of the transformer encoder, extracting information from different parts of the image patches without introducing additional parameters.

2.2. Last Pooling Layer

Traditional deep learning models typically use a linear layer as a classifier [10, 9, 27, 18, 4, 19, 25, 20, 6, 2], but a linear-only classifier often falls short in capturing the diverse representations of input images. To address this, gramian attention heads [16] introduce multiple heads, each comprising a gram matrix and an attention layer, to enhance the classifier's expressiveness with the help of de-correlation loss. The gram matrix gathers the correlations from feature maps produced by the final encoder and uses them as queries in the attention layer, allowing these correlations to serve as an importance scores for each patch.

3. Method

3.1. Sequence Pooling

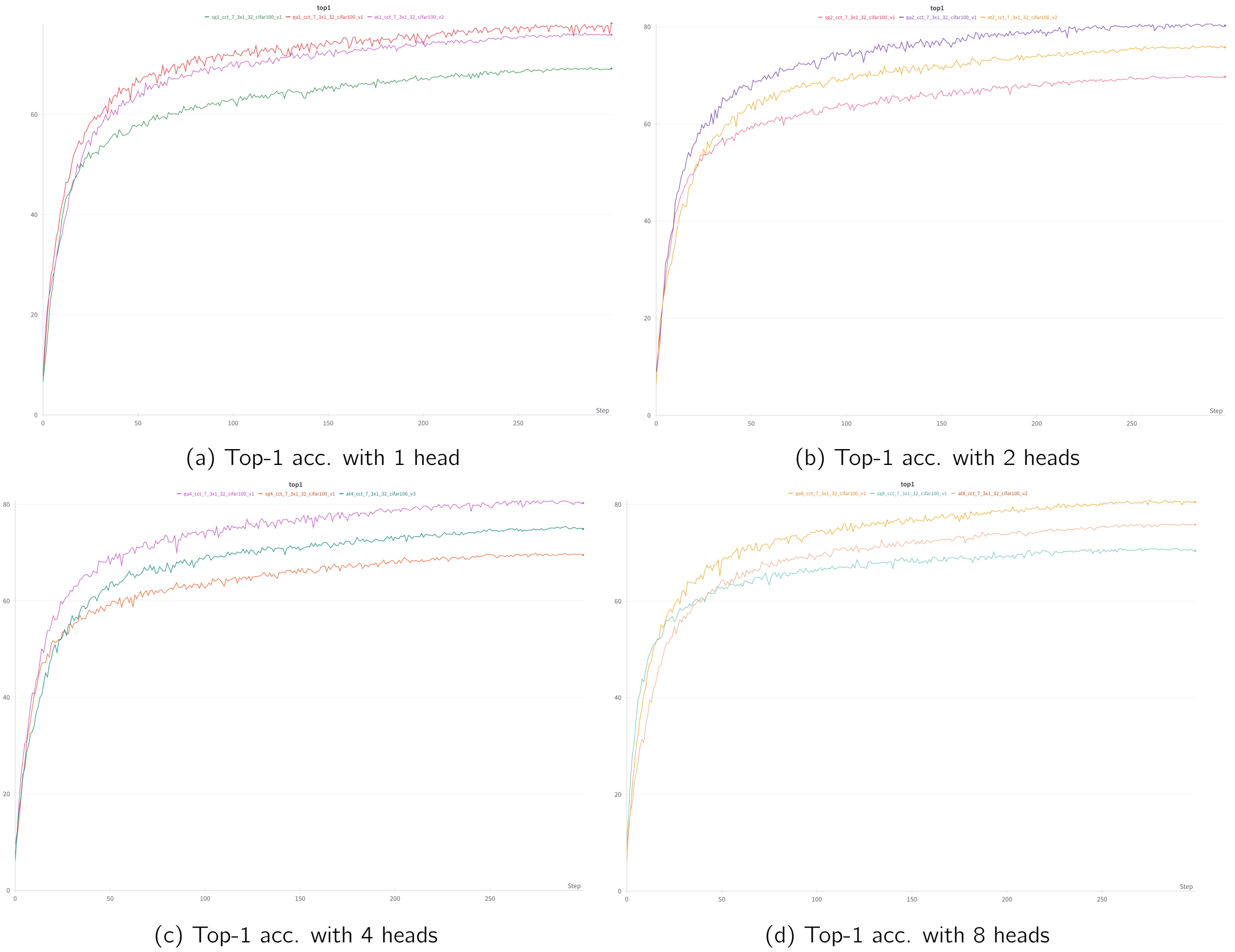
The output of the last transformer encoder in CCT [3] has the shape $Y \in R^{(B,N,D)}$. This output is passed through a linear layer to reduce the dimension to $R^{(B,N,1)}$, giving each patch a score. These scores are then converted into probability values representing the importance of each patch using the softmax function. By performing matrix multiplication between Y and these probability values, where each dimension of all patches is multiplied in a linear combination, the resulting feature is generated to be used by the final fully connected layer classifier. This process is known as sequence pooling.

3.2. Gramian Attention

Gramian attention [16] is broadly divided into two parts: the gram matrix and the attention layer. In the gram matrix, the input is the feature map $X \in R^{(B,N,D)}$, which is the output from the final transformer encoder. The N dimension of this feature map is divided into height and width corresponding to the patch size, and the dimension is compressed using a 1×1 convolution to get $X' \in R^{(B,HxW,D')}$, where D' is much smaller than D. The gram matrix is then calculated as $G = (X')^T(X') \in R^{(B,D',D')}$. As the equation shows, the gram matrix represents the correlations between the compressed dimensions. After this, the compressed dimensions are transformed back to the original dimension using a 1×1 convolution, completing the query that will be used in the attention layer. This query serves the same role as the class token in a typical ViT [2]. The key and value inputs to the attention layer are the feature map X as mentioned above, and the attention operation is then computed, forming the classifier based on the importance of each patch.

3.3. SeqAttention

This study proposes a new head, called 'SeqAttention', which outperforms both sequence pooling and gramian attention. This head architecture is a combination of sequence pooling and attention as explained above in section 3.1 and 3.2. Output of the sequence pooling layer will have $R^{(B,1,D)}$ and this will be a query in the attention layer. Key and value will be an output of the final transformer encoder with shape $R^{(B,N,D)}$. The key difference between



(a) Top-1 acc. with 1 head

(b) Top-1 acc. with 2 heads

(c) Top-1 acc. with 4 heads

(d) Top-1 acc. with 8 heads

Figure 1: **Validation for each component of the gramian head.** (a), (b), (c), and (d) represent the top-1 accuracy (%) on the CIFAR-100 dataset for different numbers of heads across three variant models ('SG': only gram matrix layer of gramian attention head, 'AT': only attention layer of gramian attention head, 'GA': original gramian attention head).

SeqAttention and gramian attention is that SeqAttention replaces the gram matrix computation in the original gramian attention with sequence pooling to calculate the importance of each token. This approach is similar to the channel-wise importance calculation process described in [7]. In contrast, gramian attention focuses more on determining the correlation between dimensions rather than between tokens using the gram matrix

4. Experiment

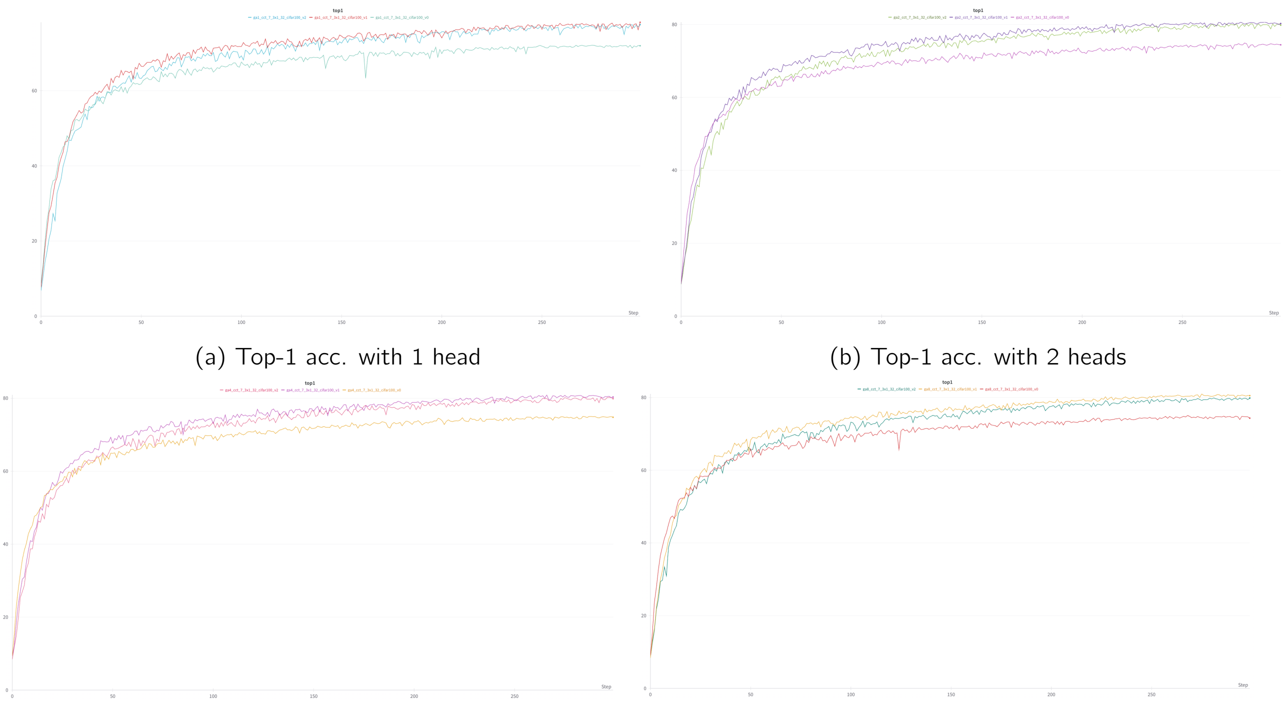
4.1. Dataset

All models are evaluated on the CIFAR-100 classification dataset [8], which includes 100 classes. The models are trained on the 50k training images and tested on the 10k validation images from scratch. Both top-1 and top-5 accuracy are assessed.

During training, various data augmentation techniques were applied to learn from diverse datasets, including cutmix [26], mixup [28], horizontal flip, random augment [1], random resized crop, random erasing, and ImageNet-based normalization. Only minimal augmentation methods were used to adjust the data size for testing

4.2. Training

All models are trained for 300 epochs using AdamW [12] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a batch size of 256, and a weight decay of 6e-2. A cosine scheduler is employed to reduce the learning rate at each iteration, and a linear learning rate warmup scheduler is applied to gradually increase the learning rate over the first 10 epochs.



(a) Top-1 acc. with 1 head

(b) Top-1 acc. with 2 heads

(c) Top-1 acc. with 4 heads

(d) Top-1 acc. with 8 heads

Figure 2: **Different dimension contraction values for GA-CCT-7-3x1-32** (a), (b), (c), and (d) are the top-1 accuracy (%) on the CIFAR-100 dataset for different number of heads of GA-CCT-7-3x1-32. Each subgraph represents GA-CCT-7-3x1-32 with different contraction dimensions: 192, 64, 32. It is observed that the best performance occurred when the dimension is 64, regardless of the number of heads.

4.3. Validation of Gram Matrix and Attention

First, a validation check experiment was conducted to determine whether each component of the gramian attention head (gram matrix and attention) [16] is effective. The classification accuracy was compared between models where the gram matrix and attention were each removed and the original model with the full gramian attention head applied. The prefix 'SG' indicates a simple gramian, meaning a model with the attention part removed; 'AT' refers to a model with attention that the gram matrix is removed; and 'GA' represents a model that reproduces the naive gramian head proposed in [16]. The experimental results in **??** showed that models with either component removed performed worse than the original gramian attention head, indicating that both the gram matrix and attention components are necessary for the head. When comparing each component individually, the performance dropped by about 6% when the attention module was removed, compared to when the gram matrix was absent. This is because, during dimension contraction in gram matrix, not all information from the feature map is encapsulated, while the attention operation retrieves all information from the original feature map as key and value.

Based on these results, all subsequent experiments were conducted using the head with both the gram matrix and attention module. For future experiments, note that the prefix 'GA8' in the model name indicates the use of 8 gramian attention heads.

4.4. Optimal Dimension Contraction in terms of CCT

Looking closely at the gram matrix part of the gramian attention head, it can be seen that it compresses the dimensions of the feature map. This is done to reduce the computational burden of calculating the gram matrix, but it is observed that the classification performance varies significantly depending on the degree of this dimension reduction (see fig. 2). Experiments are conducted with three different versions where the dimension is reduced to 192, 64, and 32, and among them, the dimension with 64 version shows overwhelming performance. If the dimension is too small, like 32, the gram matrix fails to encapsulate all the necessary information from the feature map obtained by transformer encoders, whereas if the dimension is too large, like 192, it includes lots of noises that were learned by the transformer from the input images of the small dataset [14]. [13] suggests that ViTs, which have relatively fewer parameters, possess higher robustness compared to CNNs, and this supports the findings of our experiments conducted with small datasets.

Model (CIFAR-100)	Top-1 acc. (%)	Top-5 acc. (%)	# Params.	improvements
CCT-7/3x1	76.93	93.91	3,783,269	-
GA1-CCT-7/3x1	77.40	94.23	7,191,172	-
SeqAttn1-CCT-7/3x1	**78.47**	**94.48**	**6,636,684**	-7.7%
Model (ImageNet)	**Top-1 acc. (%)**	**Top-5 acc. (%)**	**# Params.**	**improvements**
CCT-7/3x1	65.18	86.73	3,783,269	-
GA1-CCT-7/3x1	65.43	86.73	7,191,172	-
SeqAttn1-CCT-7/3x1	**65.90**	**87.37**	**6,636,684**	-7.7%

Table 1: **CIFAR-100 and ImageNet results.** CCT with SeqAttention head is compared with the other models including CCT-7/3x1 and GA1-CCT-7/3x1. All results are trained from scratch. It turns out that SeqAttention outperforms all other models with less parameters compared to the gramian attention.

4.5. Sequence Pooling with Attention

To demonstrate that the newly proposed SeqAttention is lightweight and achieves better performance, a comparative experiment is conducted on the CCT-7/3x1 model and the GA1-CCT-7/3x1 model using the CIFAR-100 dataset, focusing on top-1 accuracy, top-5 accuracy, and the number of parameters. As shown in table 1, while the model using gramian attention outperforms the original CCT, its size more than doubled. However, the model with SeqAttention demonstrates better peformance than the gramian attention model and also reduces the model size by around 1 million parameters.

5. Conclusion

This study improves the performance of the original CCT architecture on the CIFAR-100 dataset by approximately 3.6%. This enhancement is achieved by replacing sequence pooling with multiple gramian attention heads. Moreover, through various related experiments, it is found that the SeqAttention can further improve the performance of a gramian attention with 1 million less parameters. However, there may still be some unexplored feature spaces within the gramian attention head and SeqAttention. Future research focused on identifying these regions and further enhancing the performance of the gramian attention and SeqAttention could contribute to the extension of the CCT architecture for better classification of small datasets.

References

[1] Ekin D Cubuk et al. "Randaugment: Practical automated data augmentation with a reduced search space". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020, pp. 702–703.

[2] Alexey DOSOVITSKIY. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[3] Ali Hassani et al. "Escaping the big data paradigm with compact transformers". In: *arXiv preprint arXiv:2104.05704* (2021).

[4] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[5] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network". In: *arXiv preprint arXiv:1503.02531* (2015).

[6] Andrew G Howard. "MobileNets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861* (2017).

[7] Jie Hu, Li Shen, and Gang Sun. "Squeeze-and-excitation networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[8] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images". In: (2009).

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012).

[10] Yann LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[11] Youwei Liang et al. "Not all patches are what you need: Expediting vision transformers via token reorganizations". In: *arXiv preprint arXiv:2202.07800* (2022).

[12] Ilya Loshchilov and Frank Hutter. "Decoupled weight decay regularization". In: *arXiv preprint arXiv:1711.05101* (2017).

[13] Muhammad Muzammal Naseer et al. "Intriguing properties of vision transformers". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 23296–23308.

[14] Namuk Park and Songkuk Kim. "How do vision transformers work?" In: *arXiv preprint arXiv:2202.06709* (2022).

[15] Prajit Ramachandran et al. "Stand-alone self-attention in vision models". In: *Advances in neural information processing systems* 32 (2019).

[16] Jongbin Ryu, Dongyoon Han, and Jongwoo Lim. "Gramian Attention Heads are Strong yet Efficient Vision Learners". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 5841–5851.

[17] Mark Sandler et al. "Mobilenetv2: Inverted residuals and linear bottlenecks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4510–4520.

[18] Karen Simonyan. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[19] Christian Szegedy et al. "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[20] Mingxing Tan. "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *arXiv preprint arXiv:1905.11946* (2019).

[21] Hugo Touvron et al. "Training data-efficient image transformers & distillation through attention". In: *International conference on machine learning*. PMLR. 2021, pp. 10347–10357.

[22] Ashish Vaswani. "Attention is all you need". In: *arXiv preprint arXiv:1706.03762* (2017).

[23] Huiyu Wang et al. "Axial-deeplab: Stand-alone axial-attention for panoptic segmentation". In: *European conference on computer vision*. Springer. 2020, pp. 108–126.

[24] Yulin Wang et al. "Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition". In: *Advances in neural information processing systems* 34 (2021), pp. 11960–11973.

[25] Saining Xie et al. "Aggregated residual transformations for deep neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1492–1500.

[26] Sangdoo Yun et al. "Cutmix: Regularization strategy to train strong classifiers with localizable features". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6023–6032.

[27] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer. 2014, pp. 818–833.

[28] Hongyi Zhang et al. "mixup: Beyond empirical risk minimization". In: *arXiv preprint arXiv:1710.09412* (2017).

[29] Lei Zhu et al. "Biformer: Vision transformer with bi-level routing attention". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 10323–10333.