ViT Graph Head Attention for Small Sized Datasets

Hyeong Jin Kim [1] **Gyung Hyun Lee** [1] **Byoung Chul Ko** [1]

Abstract

In this paper, we propose a new type of vision transformer (ViT) based on a graph head attention (GHA). The GHA creates the graph structure using an attention map generated from the input patches. Because the attention map represents the degree of concentration between image patches, it can be regarded as a type of relationship between patches, which can be converted into a graph structure. To maintain an MHA-like performance with fewer GHAs, we apply a graph attention network to the GHA to ensure attention diversity and emphasize the correlations between graph nodes. The proposed GHA maintains both the locality and globality of the input patches and guarantees diversity of attention. The proposed GHA-ViT commonly outperforms pure ViT-based models on small-sized and a medium-sized ImageNet-1K dataset through scratch training. A top-1 accuracy of 81.7% was achieved in ImageNet-1K with GHA-B, which is a base model with approximately 29M parameters.

1. Introduction

Transformers have become one of the most powerful neural network tools and have shown a promising performance with sequential data, such as in natural language processing (NLP) (Vaswani et al., 2017) and speech recognition (Pham et al., 2019). A vision transformer (ViT) (Dosovitskiy et al., 2021), a transformer applied in the field of computer vision, is a leading algorithm used in various vision problems such as image classification (Dosovitskiy et al., 2021), image segmentation (Yan et al., 2022), object tracking (Zeng et al., 2022), depth estimation (Ranftl et al., 2021), and action recognition (Chen & Ho, 2022). However, these ViT-based methods do not properly consider the spatial geometry relationship between local regions or between global and local regions and have limitations in reducing the number of computations because they depend heavily on the combination of multi-head attentions (MHA). In addition, because a ViT requires a large number of training data, if it is trained with a small dataset without a pretraining, the performance is significantly reduced.

Contribution of This Work. In this study, to reduce the number of operations of a ViT and preserve the global and local features for image classification, we developed a graph head attention (GHA) for a ViT that replaces multi-head attentions (MHA) with fewer graph-heads using the proposed graph generation and graph attention. **i)** Unlike other graph-based transformers (Shen et al., 2021; Zheng et al., 2021; Lin et al., 2021) that operate graphs and an attention in parallel and combine the outputs, this study is the first attempt to apply a graph to the inside of the transformer's head and replace the MHA with a few GHAs. **ii)** Unlike a pure ViT, there is no need for a class token in patch embedding, and thus the number of operations can be reduced. **iii)** GHA-ViT shows a promising classification performance with only scratch training conducted on small and medium-sized datasets and no pre-training on large datasets. Figure 1 shows the overall architecture of the proposed GHA-ViT model.

2. Related Work

2.1. Graph Vision Transformer Models

Shen *et al*.(Shen et al., 2021) proposed a graph interactive transformer (GiT) for vehicle re-identification. Zheng *et al*.(Zheng et al., 2021) proposed a graph-transformer network, which is a graph representation of a whole-slide image, and a method for fusing the transformers. Mesh Graphormer (Lin et al., 2021) integrated a graph convolution with self-attention to reconstruct human poses and meshes from a single image. The vision graph neural network (ViG) (Han et al., 2022) was the first to combine graph structures with images. It regards each image patch as a single graph node and employs a k-NN to build relations between each image patch. Despite the novelty of this approach, it has the disadvantage of capturing only the similarity of the image patches without considering the latent image structure.

[1]Department of Computer Engineering, Keimyung University, Daegu, South Korea. Correspondence to: Byoung Chul Ko <niceko@kmu.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

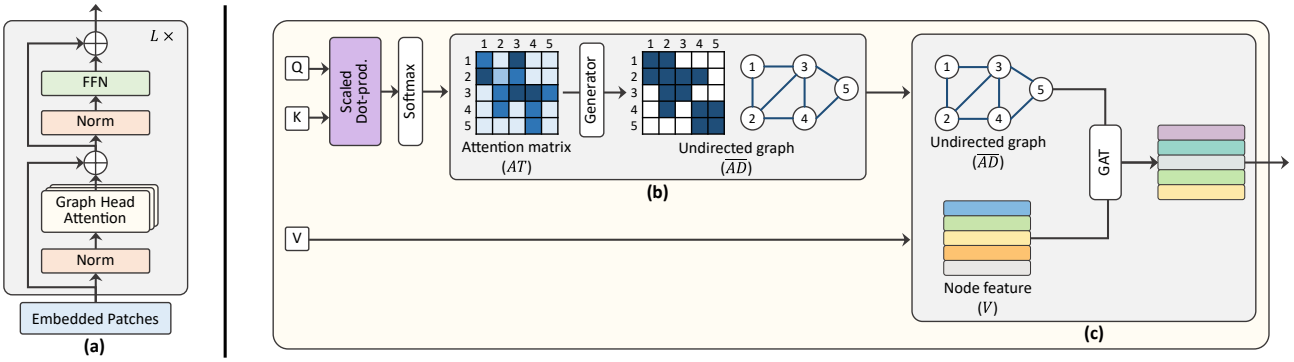

Figure 1. Overall architecture of the proposed GHA-ViT model: (a) GHA encoder layer composed of a few graph heads, (b) The attention score is calculated as the scaled dot product of Q and transposed K. Then, the graph generator is applied to the attention matrix AT for selecting sub-nodes and convert it to undirected graph $\overline{AD}$, (c) After graph generation, the value V and an undirected graph $\overline{AD}$ are applied to a graph attention network (GAT). Based on the generated graph and node features V, GAT gives a different importance to the neighboring nodes.

3. Graph Head Attention

The attention score is calculated as the scaled dot product of Q and transposed K, and can be executed in parallel. The transformer uses an MHA, and thus n heads can learn different attentions from the input and obtain a strong attention representation through their combination (Dosovitskiy et al., 2021). However, in reality, not all heads of the MHA have the same effect on the attention performance of a transformer. Rather, only a part of the head affects the attention performance of the transformer, and the remainder focuses on unnecessary parts, which negatively affects the outcome of the final attention (Michel et al., 2019). From this perspective, it is clear that an MHA is not essential in a transformer. In this section, we propose a new transformer that can receive better attention with fewer GHAs and without the use of multiple heads. Let X be the input of the encoder layer. Here, X consists of P patches, and the hidden dimension of the input patch is d_h. The Q, K, and V matrices have corresponding weight matrices W_Q, W_K and $W_V \in \mathbb{R}^{d \times d_h}$, and Q, K, and V can be obtained through the dot product of input H and the weight matrices. The attention matrix AT of the head is calculated as follows:

$$\begin{aligned} Q &= XW_Q \\ K &= XW_K \quad Q, K, V \in \mathbb{R}^{P \times d_h} \\ V &= XW_V \end{aligned} \tag{1}$$

$$AT = \text{softmax}(\frac{QK^T}{\sqrt{d_h}}) \in \mathbb{R}^{P \times P} \tag{2}$$

3.1. Graph Structure Generation

In a CNN, to provide a better generalization and performance, the pooling layer plays an important role in reducing the size of the feature map and broadening the receptive field. However, this pooling operation cannot be applied directly to the graph because there is no local information between the graph nodes. Therefore, inspired by (Gao & Ji, 2019; Lee et al., 2019), we propose graph pooling containing local information based on a mask filter. We apply the $Top-k$ function to the attention matrix AT for selecting sub-nodes with a significant connectivity. Distilled sparse nodes can be regarded as a new form of graph structure derived from an attention matrix. In other words, we consider the sparse matrix of these nodes to be the adjacency matrix AD of a graph.

$$AD = Top-k(AT, k) \tag{3}$$

We can now construct a graph consisting of node patches using the AD. Again, to consider the self-edge of the node, AD adds identity matrix I. However, in the initial graph constructed from AD, directed and undirected edges are mixed. In a transformer, because the attention is created by interactions with neighboring patches, a directed edge cannot guarantee the correct patch attention. Therefore, we must convert the mixed graph into an undirected graph. For this purpose, the following graph transformation method is proposed: First, as indicated in Eq. 4, the upper matrix of AD and its transpose are added to form a partially undirected graph:

$$AD_{triU} = triU(AD) + triU(AD)^T \tag{4}$$

$$AD_{triL} = triL(AD) + triL(AD)^T \tag{5}$$

Finally, the upper matrix AD_{triU} and the lower matrix AD_{triL} generate an undirected graph $\overline{AD}$ through an OR operation $\vee$.

$$\overline{AD} = AD_{triU} \vee AD_{triL} \tag{6}$$

3.2. Graph Head Attention Boosting

To improve the accuracy of the image classification, a detailed attention can be obtained through a combination of MHAs. However, an MHA requires weight matrices W_Q, W_K, and W_V for each head. Therefore, as the number of heads increases, more learning parameters, memories, and computational times are required. To avoid the problems caused by an MHA, we use fewer GHAs and show that the transformer can operate successfully using only the proposed GHA. To ensure a diversity of attention, similar to an MHA with fewer attention heads, and to emphasize the correlation between graph nodes, we apply a GAT (Veličković et al., 2017) to a GHA. When using a GHA instead of a general MHA mechanism, securing the diversity of attention is an extremely important part of the successful operation of the GHA model. We previously extracted $\overline{AD}$ representing the relationship between node patches from the attention matrix. We now apply a GAT to $\overline{AD}$ to conduct efficient attention computations between the nodes with $\overline{AD}$. From $\overline{AD}$ and V, the attention coefficient e between nodes i and j is obtained using the learnable weight matrix W_c.

$$e_{ij} = \mathrm{FN}(W_c \cdot v_i, W_c \cdot v_j) \tag{7}$$

Here, an FN is a simple single-layer feedforward neural network that transforms the input into $\mathbb{R}^{1\times d_h} \times \mathbb{R}^{d_h \times 1} \rightarrow \mathbb{R}$. The above expression indicates the importance of the features of nodes i–j. At this time, j does not indicate all nodes, but only the neighbors N_i of node i. Finally, if it passes the softmax function, the following normalized attention matrix $\widetilde{AD}$ can be calculated:

$$\widetilde{AD}_{ij} = \frac{exp(e_{ij})}{\sum_{k\in N_i} exp(e_{ik})} \tag{8}$$

Eq. 8 is applied to every encoder layer to ensure the diversity of the node features. The final GHA is produced by applying the node feature matrix $V$and the weight matrix W_{gat} of the GAT to $\widetilde{AD}$ as follows:

$$\mathrm{GHA}(V) = \sigma_2(\widetilde{AD} \cdot V \cdot W_{gat}) \tag{9}$$

where σ_2 is a ReLU activation function. The value of GHA is obtained through the process of Fig. 1 (c). Because the GHA encoder consists of L layers, V is input to the first layer, but from the second layer, V is changed to h_l, the output of each layer. GHA(h_{l-1}) of the previous layer is again skip-connected (element-wise sum) with input h_{l-1}.

$$\acute{h}_l = \mathrm{GHA}(h_{l-1}) + h_{l-1}, l \in \{1 \cdots L\} \tag{10}$$

After $\acute{h}_l$ is linearly normalized (LN) again and applied to the FFN, it is similarly skip-connected to the original $\acute{h}_l$ to produce the output of the final encoder block.

$$h_l = \mathrm{FFN}(\acute{h}_l) + \acute{h}_l, l \in \{1 \cdots L\} \tag{11}$$

Finally, the output of last encoder layer L, $h_L \in \mathbb{R}^{P\times d_h}$, is passed to the readout layer. For the readout layer, sequence pooling (seq) (Hassani et al., 2021), the mean and max values were used to consider the diversity (Kim & Cho, 2019). The multi-readout feature H_{out} is calculated:

$$H_{out} = h_{seq} \parallel h_{mean} \parallel h_{max} \tag{12}$$

where $\parallel$ denotes the concatenation operation between node features. The multi-readout feature H_{out}, which has passed through the readout, is then classified using MLP. The loss function was optimized using a soft distillation (Hinton et al., 2015; Wei et al., 2020).

Table 1. Details on GHA-ViT model variants. dim d means hidden dimensions in encoders, and mlp ratio means a scaling factor for hidden dimension of MLP. In GHA-*-a / b, a means number of layer and b is patch size.

Model	head	layers	dim d	mlp ratio
GHA-S-7/3	4	7	64	2
GHA-B-7/3	6	7	64	2
GHA-S-14/7	3	14	64	4
GHA-B-14/7	6	14	64	4

4. Dataset and Experimental Results

To evaluate the representation learning ability of the proposed GHA-ViT model, we compare it with ResNet (He et al., 2016) and MobileNetV2 (Sandler et al., 2018), which are representative CNN models; ViT-based methods (Dosovitskiy et al., 2021; Touvron et al., 2021b; Hassani et al., 2021; Yu et al., 2022); MLP-based approaches (Tolstikhin et al., 2021; Touvron et al., 2021a; Liu et al., 2021a); and a graph-based method (Han et al., 2022). We prove through our experiments that the performance of the proposed model is similar to that of other state-of-the-art (SoTA) methods on several benchmark datasets.

4.1. Experiment Setup

Datasets. We used the ImageNet-1K (Deng et al., 2009) to measure the capacity of the proposed GHA.

Baseline. We set the baseline of the GHA differently to prove that the proposed GHA-ViT model can reduce the number of heads and encoder layers. The basic structure of GHA-ViT is based on DeiT (Touvron et al., 2021b) because it is inherently capable of learning with a small dataset. The baseline models have two types, GHA-Base and GHA-Small according to the number of heads and layers. Table 1 summarizes the GHA model used as a baseline. In addition to GHA-ViT, we used ResNet (He et al., 2016) as the CNN baseline model for the comparative experiments. The

Table 2. Performance comparison of scratch-trained CNN and transformer-based models on the ImageNet-1K dataset. Image resolution is same as 224×224.

Model	Params (M)	MACs (G)	Top-1 (%)	Top-5(%)
ResNet-50 (He et al., 2016)	26	4.3	76.2	95.0
ResNet-101 (He et al., 2016)	45	7.9	77.4	95.4
ResNet-152 (He et al., 2016)	60	11.6	78.3	95.9
ViT-S-16 (Dosovitskiy et al., 2021)	47	10.1	78.1	-
DeiT-S (Touvron et al., 2021b)	22	4.6	79.8	95.0
CCT-14/7×2 (Hassani et al., 2021)	22	18.6	80.6	-
T2T-ViT-14 (Yuan et al., 2021b)	22	4.8	81.5	-
PoolFormer-S12 (Yu et al., 2022)	12	1.8	77.2	-
PoolFormer-S24 (Yu et al., 2022)	30	3.0	80.3	-
Mixer-B/16 (Tolstikhin et al., 2021)	59	12.7	76.4	-
ResMLP-12 (Touvron et al., 2021a)	15	3.0	76.6	-
gMLP-Ti (Liu et al., 2021a)	6	1.4	72.3	-
gMLP-S (Liu et al., 2021a)	20	4.5	79.6	-
ViG-Ti (Han et al., 2022)	7	1.3	73.9	92.0
ViG-S (Han et al., 2022)	23	4.5	80.4	95.2
GHA-S-14/7	10	1.8	77.4	93.5
GHA-B-14/7	29	5.9	81.7	95.8

ResNet model modified the last MLP layer to suit the number of classes for each experimental dataset. Pure ViT and DeiT are used as transformer models for comparison with the GHA-ViT. These methods also changed the last MLP layer to obtain suitable outputs for each set of experimental data.

4.2. Compare Performance with State-of-the-arts model

We conducted experiments using the ImageNet-1K dataset to demonstrate that the proposed model works effectively not only on small but also on medium-sized dataset. Table 2 shows the results of a performance comparison between the proposed GHA-ViT and SoTA methods. Compared to ResNet-152 among CNN-based methods (He et al., 2016), the number of parameters in the GHA-S model is reduced by up to 6-times, and the number of operations is up to 6.4-times faster. In terms of the accuracy, the GHA-S model is slightly inferior to ResNet-152, whereas the GHA-B model improved the accuracy by 3.4% in Top-1 accuracy. In a comparison with ViT-based methods (Dosovitskiy et al., 2021; Touvron et al., 2021b; Hassani et al., 2021; Yuan et al., 2021b; Yu et al., 2022), in terms of accuracy, the GHA-S model increased the Top-1 accuracy by 0.5% compared to PoolFormer-S12 under the same conditions (Param and MACs). In the case of GHA-B, the Top-1 accuracy was the highest at 81.7%; however, the number of operations was 1.1 higher than that of T2T-ViT-14 with a similar accuracy. In comparison with MLP-based models (Tolstikhin et al., 2021; Touvron et al., 2021a; Liu et al., 2021a),both the GHA-S and GHA-B models increased their Top-1 accuracy by approximately 2% on gMLP-Ti and gMLP-S, which have similar numbers of parameters and operations. In addition, in comparison with ViG methods (Han et al., 2022) using graph structures, the numbers of parameters and operations are slightly higher, whereas the GHA-S and GHA-B models showed a high accuracy of 3.8% and 1.3%, respectively. This is because the proposed GHA-ViT model can generate a graph structure with a higher efficiency than the graph generation method used in ViG.

5. Conclusion

In this paper, we proposed a new GHA method that can overcome the limitations of MHA, the core module of ViT. By converting the attention map operation from a matrix perspective to a graph perspective, it was possible to significantly reduce the number of unnecessary operations and parameters while maintaining the accuracy of image classification. We also proved that the attention feature space embedded in multi-heads was not significantly different from that when only fewer graph heads were used. In the future, we will apply the method of combining the approaches of graph pooling such as graph U-NET (Gao & Ji, 2019) to improve the mask filter for constructing the graph and for more meaningful attention output. Through these additional studies, the ViT performance of the GHA structure is expected to be significantly improved than that of the MHA-based ViT approaches.

References

Chen, J. and Ho, C. M. Mm-vit: Multi-modal video transformer for compressed video action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applica-*

tions of Computer Vision (WACV), pp. 1910–1921, 2022.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255. Ieee, 2009.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICML)*, pp. 1–21, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Gao, H. and Ji, S. Graph u-nets. In *International Conference on Machine Learning (ICML)*, pp. 2083–2092. PMLR, 2019.

Han, K., Wang, Y., Guo, J., Tang, Y., and Wu, E. Vision gnn: An image is worth graph of nodes. *arXiv preprint arXiv:2206.00272*, 2022.

Hassani, A., Walton, S., Shah, N., Abuduweili, A., Li, J., and Shi, H. Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Hinton, G., Vinyals, O., Dean, J., et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Kim, J.-Y. and Cho, S.-B. Electric energy consumption prediction by deep learning with state explainable autoencoder. *Energies*, 12(4):739, 2019.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lee, J., Lee, I., and Kang, J. Self-attention graph pooling. In *International conference on machine learning (ICML)*, pp. 3734–3743. PMLR, 2019.

Lee, J., Jeong, M., and Ko, B. C. Graph convolution neural network-based data association for online multi-object tracking. *IEEE Access*, 9:114535–114546, 2021.

Lee-Thorp, J., Ainslie, J., Eckstein, I., and Ontanon, S. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*, 2021.

Li, Y., Zhang, K., Cao, J., Timofte, R., and Van Gool, L. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

Lin, K., Wang, L., and Liu, Z. Mesh graphormer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12939–12948, 2021.

Liu, H., Dai, Z., So, D., and Le, Q. V. Pay attention to mlps. In *Advances in neural information processing systems (NeurIPS)*, volume 34, pp. 9204–9215, 2021a.

Liu, Y., Sangineto, E., Bi, W., Sebe, N., Lepri, B., and Nadai, M. Efficient training of visual transformers with small datasets. In *Advances in neural information processing systems (NeurIPS)*, volume 34, pp. 23818–23830, 2021b.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10012–10022, 2021c.

Michel, P., Levy, O., and Neubig, G. Are sixteen heads really better than one? In *Advances in neural information processing systems (NeurIPS)*, volume 32, pp. 1–11, 2019.

Pham, N.-Q., Nguyen, T.-S., Niehues, J., Müller, M., Stüker, S., and Waibel, A. Very deep self-attention networks for end-to-end speech recognition. *arXiv preprint arXiv:1904.13377*, 2019.

Ranftl, R., Bochkovskiy, A., and Koltun, V. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12179–12188, 2021.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.

Shen, F., Xie, Y., Zhu, J., Zhu, X., and Zeng, H. Git: Graph interactive transformer for vehicle re-identification. *arXiv preprint arXiv:2107.05475*, 2021.

Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al. Mlp-mixer: An all-mlp architecture for vision. In *Advances in neural information processing systems (NeurIPS)*, volume 34, pp. 24261–24272, 2021.

Touvron, H., Bojanowski, P., Caron, M., Cord, M., El-Nouby, A., Grave, E., Izacard, G., Joulin, A., Synnaeve, G., Verbeek, J., et al. Resmlp: Feedforward networks for image classification with data-efficient training. *arXiv preprint arXiv:2105.03404*, 2021a.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning (ICML)*, pp. 10347–10357. PMLR, 2021b.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems (NeurIPS)*, volume 30, pp. 1–11, 2017.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., and Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 568–578, 2021.

Wei, L., Xiao, A., Xie, L., Zhang, X., Chen, X., and Tian, Q. Circumventing outliers of autoaugment with knowledge distillation. In *European Conference on Computer Vision (ECCV)*, pp. 608–625. Springer, 2020.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Yan, X., Tang, H., Sun, S., Ma, H., Kong, D., and Xie, X. After-unet: Axial fusion transformer unet for medical image segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3971–3981, 2022.

Ying, C., Cai, T., Luo, S., Zheng, S., Ke, G., He, D., Shen, Y., and Liu, T.-Y. Do transformers really perform badly for graph representation? In *Advances in neural information processing systems (NeurIPS)*, volume 34, pp. 28877–28888, 2021.

Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., and Yan, S. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10819–10829, 2022.

Yuan, K., Guo, S., Liu, Z., Zhou, A., Yu, F., and Wu, W. Incorporating convolution designs into visual transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 579–588, 2021a.

Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.-H., Tay, F. E., Feng, J., and Yan, S. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 558–567, 2021b.

Yun, S., Jeong, M., Kim, R., Kang, J., and Kim, H. J. Graph transformer networks. In *Advances in neural information processing systems (NeurIPS)*, volume 32, pp. 1–11, 2019.

Yun, S., Jeong, M., Yoo, S., Lee, S., Sean, S. Y., Kim, R., Kang, J., and Kim, H. J. Graph transformer networks: Learning meta-path graphs to improve gnns. *Neural Networks*, 153:104–119, 2022.

Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X., and Wei, Y. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision (ECCV)*, pp. 659–675. Springer, 2022.

Zhao, W., Wang, W., and Tian, Y. Graformer: Graph-oriented transformer for 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20438–20447, 2022.

Zheng, Y., Gindra, R., Betke, M., Beane, J. E., and Kolachalama, V. B. A deep learning based graph-transformer for whole slide image classification. *medRxiv*, 41:1–14, 2021.