GreedyViG: Dynamic Axial Graph Construction for Efficient Vision GNNs

Supplementary Material

A. Further Ablation Studies

The ablation studies are conducted on ImageNet-1K [3]. Table 4 reports the ablation study of GreedyViG-B (GViG-B) on the effects of graph convolutions at higher resolution stages and Table 5 reports the effects of static versus dynamic graph construction.

Graph convolutions at higher resolution stages. In Table 4 we can see that adding graph convolutions at higher resolution stages improves top-1 accuracy with a relatively small increase in parameters. By 1-stage, 2-stage, 3-stage, and 4-stage we mean that the DAGC blocks (graph convolution block) will be used in stage 4, stages 3 and 4, stages 2, 3, and 4, or in all stages as shown in Figure 4. GreedyViG-B increases in top-1 accuracy as we move from 1-stage to 4-stage increasing from 83.1% at the 1-stage configuration to 83.5% at the 2-stage configuration. Moving from the 2-stage configuration to the 3-stage configuration we see a 0.2% increase in accuracy reaching 83.7%. Finally, moving from 3-stage to 4-stage we see a 0.2% increase in accuracy reaching 83.9% top-1 accuracy at the 4-stage configuration. Comparing the 1-stage and 4-stage configurations we see a 0.8% gain in top-1 accuracy with only an increase of 4.4 M parameters, showing the benefits of graph convolutions at higher resolution stages.

Table 4. **Ablation study for graph convolutions at higher resolution stages on ImageNet-1K benchmark.** 1-S, 2-S, 3-S, and 4-S indicate that graph convolutions were used in 1-stage, 2-stages, 3-stages, or all 4-stages. A check mark indicates this component was used in the experiment. A (-) indicates this component was not used.

Model	Params (M)	1-S	2-S	3-S	4-S	Top-1 (%)
GViG-B	26.5	✓	-	-	-	83.1
GViG-B	29.7	-	✓	-	-	83.5
GViG-B	30.7	-	-	✓	-	83.7
GViG-B	**30.9**	-	-	-	✓	**83.9**

Static versus dynamic graph construction. Compared to the static graph construction method (SVGA) proposed in [30], DAGC connects only the similar connections based on Euclidean distance resulting in improved performance. In Table 5 we can see the direct benefit of using DAGC compared to SVGA as it adds no parameters and increases the top-1 accuracy of GreedyViG-B with 4-stages by 0.4% from 83.5% to 83.9%. We can also see the benefit of DAGC and our overall GreedyViG architecture compared to the MobileViG architecture, which uses SVGA, through comparing MobileViG-B (MViG-B) and a 1-stage configuration of GreedyViG-B. The 1-stage configuration of GreedyViG-B shows a 0.5% improvement in top-1 accuracy from 82.6% to 83.1% while reducing parameters by 0.2 M, showing the benefits of dynamic graph construction.

Table 5. **Ablation study for static versus dynamic graph construction on ImageNet-1K benchmark.** 1-S indicates that graph convolutions were only used in Stage 4, while 4-S indicates that graph convolutions were used in stages 1, 2, 3, and 4. A check mark indicates this component was used in the experiment. A (-) indicates this component was not used.

Model	Params	SVGA	DAGC	1-S	4-S	Top-1 (%)
MViG-B [30]	26.7 M	✓	-	✓	-	82.6
GViG-B	26.5 M	-	✓	✓	-	83.1
GViG-B	30.9 M	✓	-	-	✓	83.5
GViG-B	**30.9 M**	-	✓	-	✓	**83.9**

B. Network Configurations

The detailed network architectures for GreedyViG-S, M, and B are provided in Table 6. We report the configuration of the stem, stages, and classification head. In each stage the number of MBConv and DAGC blocks repeated as well as their channel dimensions is reported. For GreedyViG-B, stage 4 has 3 repeated MBConv and DAGC blocks instead of 4 in order to have comparable parameters to other competing architectures.

Table 6. **Architecture details of GreedyViG** showing configuration of the stem, stages, and classification head. C represents the channel dimensions.

Stage	GreedyViG-S	GreedyViG-M	GreedyViG-B
Stem	Conv ×2	Conv ×2	Conv ×2
Stage 1	$MBConv \times 2$ $DAGC \times 2$ $C = 48$	$MBConv \times 3$ $DAGC \times 3$ $C = 56$	$MBConv \times 4$ $DAGC \times 4$ $C = 64$
Stage 2	$MBConv \times 2$ $DAGC \times 2$ $C = 96$	$MBConv \times 3$ $DAGC \times 3$ $C = 112$	$MBConv \times 4$ $DAGC \times 4$ $C = 128$
Stage 3	$MBConv \times 6$ $DAGC \times 2$ $C = 192$	$MBConv \times 9$ $DAGC \times 3$ $C = 224$	$MBConv \times 12$ $DAGC \times 4$ $C = 256$
Stage 4	$MBConv \times 2$ $DAGC \times 2$ $C = 384$	$MBConv \times 3$ $DAGC \times 3$ $C = 448$	$MBConv \times 3$ $DAGC \times 3$ $C = 512$
Head	Pooling & MLP	Pooling & MLP	Pooling & MLP

C. Computational Complexity of Graph Construction

The computational complexity for KNN, DAGC, and SVGA for a single node in the image (in terms of comparisons from that node) is given below. W and H are the width and height of the image, K is the number of nearest neighbors, and N is the number of hops selected in SVGA and DAGC.

1. **KNN**: O($W \times H \times K$). For each node, KNN finds the K nearest by comparing every node to the current node.
2. **DAGC**: O($\frac{W+H}{N}$). For each node, DAGC only needs to compare nodes that are every N hops away, thus decreasing the number of comparisons. Also, since DAGC computes the μ and σ beforehand, it makes connections in the first search through of the image rather than needing to compare again for K connections.
3. **SVGA**: O(1). Connects each node along the axes.

DAGC is more computationally expensive than SVGA, but more representative. KNN may be more representative than DAGC, but can cause oversmoothing and is more computationally expensive. The measured time taken for graph construction is 0.06 ms in DAGC, 0.38 ms in KNN, and 0.04 ms in SVGA when measured on an Nvidia RTX A6000; this shows DAGC is slower than SVGA and faster than KNN in graph construction time. This can also be seen through our latency results in Table 7. GreedyViG-S is faster and more accurate than PViG-Ti, but is slower and more accurate than a smaller MobileViG-S model. GreedyViG-S is slower compared to MobileViG-S because DAGC is slower than SVGA, GreedyViG has more parameters, and because GreedyViG contains more global processing stages that perform graph convolution (DAGC blocks) as compared to MobileViG which only does graph convolution at its lowest resolution stage after multiple downsample layers.

Table 7. **Graph construction impact on accuracy and latency.** We show GreedyViG-S with KNN and DAGC to compare with PViG-Ti with KNN and DAGC. We also show MobileViG-S with SVGA to show it is less accurate, but faster than GreedyViG-S.

Model	Params	Latency	Acc (%)
MobileViG-S [30] w/ SVGA	7.2 M	27.1 ms	78.2
PViG-Ti [7] w/ KNN	10.7 M	79.4 ms	78.2
PViG-Ti [7] w/ DAGC	10.7 M	63.3 ms	79.1
GreedyViG-S (Ours) w/ KNN	12.0 M	73.6 ms	80.2
GreedyViG-S (Ours) w/ DAGC	**12.0 M**	**53.4 ms**	**81.1**

The graph construction and architecture of GreedyViG both contribute to the performance of GreedyViG models. When using DAGC with the original ViG architecture and KNN with our GreedyViG architecture in Table 7, we can see that DAGC is faster and provides higher accuracy compared to KNN in these cases. GreedyViG-B with SVGA can also be seen Table 5, showing with the same configuration DAGC has 83.9% accuracy compared to SVGA's 83.5%.