Global Context Vision Transformers

Ali Hatamizadeh [1] **Hongxu Yin** [1] **Greg Heinrich** [1] **Jan Kautz** [1] **Pavlo Molchanov** [1]

Abstract

We propose global context vision transformer (GC ViT), a novel architecture that enhances parameter and compute utilization for computer vision. Our method leverages global context self-attention modules, joint with standard local self-attention, to effectively and efficiently model both long and short-range spatial interactions, without the need for expensive operations such as computing attention masks or shifting local windows. In addition, we address the lack of the inductive bias in ViTs, and propose to leverage a modified fused inverted residual blocks in our architecture. Our proposed GC ViT achieves state-of-the-art results across image classification, object detection and semantic segmentation tasks. On ImageNet-1K dataset for classification, the variants of GC ViT with 51M, 90M and 201M parameters achieve **84.3%**, **85.0%** and **85.7%** Top-1 accuracy, respectively, at 224×224 image resolution and without any pre-training, hence surpassing comparably-sized prior art such as CNN-based ConvNeXt and ViT-based MaxViT and Swin Transformer by a large margin. Pre-trained GC ViT backbones in downstream tasks of object detection, instance segmentation, and semantic segmentation using MS COCO and ADE20K datasets outperform prior work consistently. Specifically, GC ViT with a 4-scale DINO detection head achieves a box AP of **58.3%** on MS COCO dataset. Code is available at https://github.com/NVlabs/GCViT.

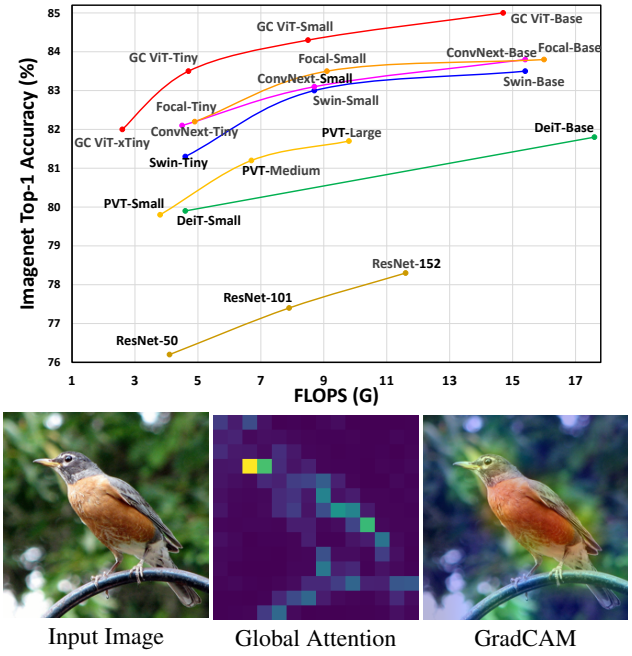


Figure 1 – GC ViT achieves a new Pareto-front with respect to ImageNet Top-1 vs number of parameters trade-off. For fair comparison, models that are trained and evaluated with input image size of 224×224 on ImageNet-1K dataset and without pre-training are considered. GC ViT is capable of capturing both short and long-range information using its global attention mechanism. We visualize corresponding attention and GradCAM maps from GC ViT to demonstrate the effectiveness of the proposed global attention mechanism.

1. Introduction

During the recent years, Transformers (Vaswani et al., 2017) have achieved State-Of-The-Art (SOTA) performance in Natural Language Processing (NLP) benchmarks and became the de facto model for various tasks. A key element in the success of Transformers is the self-attention mechanism which allows for capturing contextual representations via attending to both distant and nearby tokens (Yin et al., 2021). Following this trend, Vision Transformer (ViT) (Dosovitskiy et al., 2020) proposed to utilize image patches as tokens in a monolithic architecture with minor differences comparing to encoder of the original Transformer. Despite the historic dominance of Convolutional Neural Network (CNN) in computer vision, ViT-based models have achieved SOTA or competitive performance in various computer vision tasks.

In essence, the self-attention mechanism in ViT allows for learning more uniform short and long-range information (Raghu et al., 2021) in comparison to CNN. However, the monolithic architecture of ViT and quadratic computational complexity of self-attention baffle their swift applica-

[1]NVIDIA. Correspondence to: Ali Hatamizadeh <ahatamizadeh@nvidia.com>.

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

tion to high resolution images (Yang et al., 2021a) in which capturing multi-scale long-range information is crucial for accurate representation modeling.

Several efforts (Liu et al., 2021; Dong et al., 2022; Chu et al., 2021a; Tu et al., 2022), most notably Swin Transformer (Liu et al., 2021), have attempted to address the balance between short- and long-range spatial dependencies by proposing multi-resolution architectures in which the self-attention is computed in local windows. In this paradigm, cross-window connections such as window shifting are used for modeling the interactions across different regions. Despite the progress, the limited receptive field of local windows challenges the capability of self-attention to capture long-range information, and window-connection schemes such as shifting only cover a small neighborhood in the vicinity of each window. Subsequent efforts such as Focal Transformer (Yang et al., 2021b) attempted to address this issue by designing highly sophisticated self-attention modules with increased model complexity.

In this work, we introduce the Global Context (GC) ViT network to address these limitations. Specifically, we propose a hierarchical ViT architecture consisting of local and global self-attention modules. At each stage, we compute global query tokens, using a novel fused inverted residual blocks, which we refer to as modified Fused-MBConv blocks, that encompass global contextual information from different image regions. While the local self-attention modules are responsible for modeling short-range information, the global query tokens are shared across all global self-attention modules to interact with local key and value representations.

The design of our proposed framework for global query generator and self-attention is intuitive and simple and can be efficiently implemented using major deep learning framework. Hence, it eliminates sophisticated and computationally expensive operations and ensures the effectiveness of self-attention when applied to high-resolution images. In addition, we propose a novel downsampling block with a parameter-efficient fused-MBConv layer to address the lack of inductive bias in ViTs and enhancing the modeling of inter-channel dependencies.

We have extensively validated the effectiveness of the proposed GC ViT using three publicly available datasets for various computer vision tasks. For image classification using ImageNet-1K dataset, GC ViT with 51M, 90M, 201M parameters achieve new SOTA benchmarks of **84.3%**, **85.0%**, **85.7%** Top-1 accuracy and without using extra data or pre-training.

Hence, GC ViT consistently outperforms both ConvNeXt (Liu et al., 2022b), MaxViT (Tu et al., 2022) and Swin Transformer (Liu et al., 2021) models, sometimes by a significant margin (see Fig. 1).

Using an ImageNet-1K pre-trained GC ViT base backbone with a Cascade Mask RCNN (He et al., 2017) head, our model achieves a box mAP of **52.9** for object detection and a mask mAP of **45.8** for instance segmentation on the MS COCO dataset and by using single-scale inference. We also used an ImageNet-21K GC ViT model as backbone with a 4-scale DINO detection head and achieved a box AP of **58.3%**.

In addition, using an UPerNet (Xiao et al., 2018) head, our model achieves a mIoU of **49.2** on ADE20K for semantic segmentation by only using a single-scale inference scheme. Other variants of GC ViT with different learning capacities also demonstrate SOTA results when compared to similarly-sized models on both MS COCO and ADE20K datasets. Hence, GC ViT demonstrates great scalability for high-resolution images on various downstream tasks, validating the effectiveness of the proposed framework in capturing both short and long-range information.

The main contributions of our work are summarized as follows:

- We introduce a compute and parameter-optimized hierarchical ViT with reparametrization of the design space (*e.g.*, embedding dimension, number of heads, MLP ratio).
- We design an efficient CNN-like token generator that encodes spatial features at different resolutions for global query representations.
- We propose global query tokens that can effectively capture contextual information in an efficient manner and model both local and global interactions.
- We introduce a parameter-efficient downsampling module with modified Fused MB-Conv blocks that not only integrates inductive bias but also enables the modeling of inter-channel dependencies.
- We demonstrate new SOTA benchmarks for : (1) ImageNet classification with Pareto fronts on ImageNet-1K for number of parameters and FLOPs (2) downstream tasks such as detection, instance segmentation and semantic segmentation on MS COCO and ADE20K, respectively.

2. GC ViT architecture

Architecture. Fig. 2 depicts the architecture of GC ViT. We propose a hierarchical framework to obtain feature representations at several resolutions (called stages) by decreasing the spatial dimensions while expanding the embedding dimension, both by factors of 2.

At first, given an input image with resolution of $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, we obtain overlapping patches by applying a 3×3 convolutional layer with a stride of 2 and appropriate padding. Then patches are projected into a C-dimensional

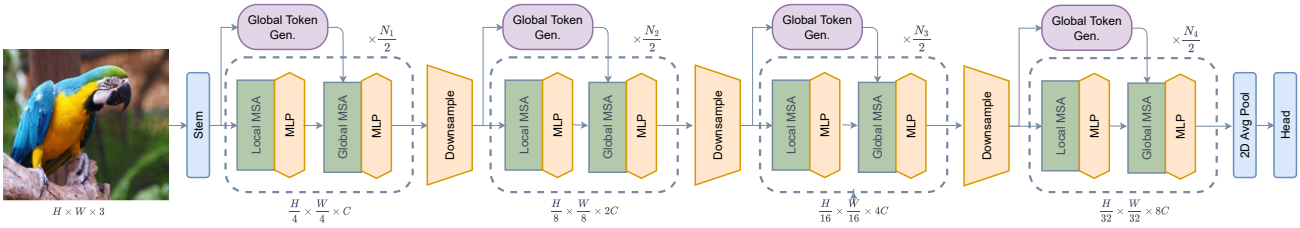


Figure 2 – Architecture of the proposed GC ViT. At each stage, a query generator extracts global query tokens which captures long-range information by interacting with local key and value representations. We use alternating blocks of local and global context self attention layers. Best viewed in color.

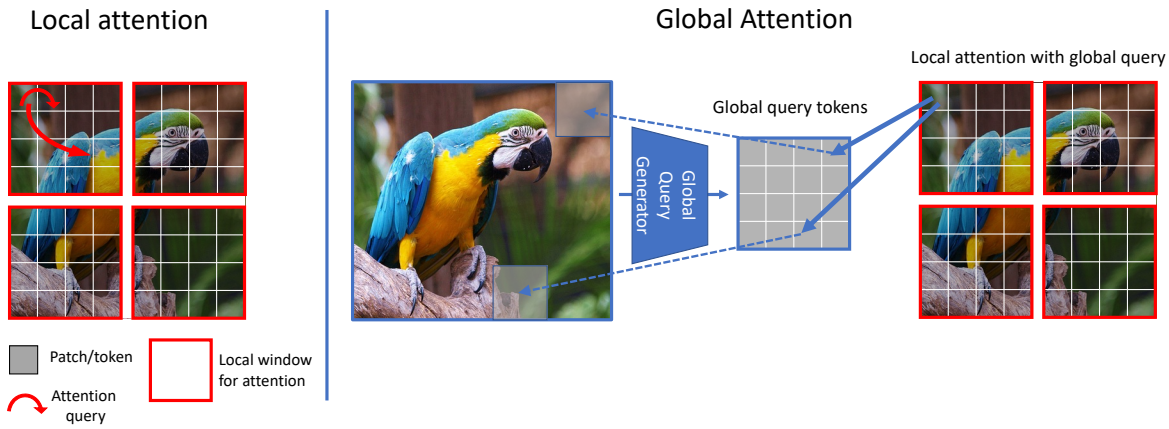


Figure 3 – Attention formulation. Local attention is computed on feature patches within local window only (left). On the other hand, the global features are extracted from the entire input features and then repeated to form global query tokens. The global query is interacted with local key and value tokens, hence allowing to capture long-range information via cross-region interaction. Best viewed in color.

embedding space with another 3×3 convolutional layer with stride 2.

Every GC ViT stage is composed of alternating local and global self-attention modules to extract spatial features. Both operate in local windows like Swin Transformer (Liu et al., 2021), however, the global self-attention has access to global features extracted by the global query generator. The query generator is a CNN-like module that extracts features from the entire image only once at every stage. After each stage, the spatial resolution is decreased by 2 while the number of channels is increased by 2 via a downsampling block. Resulting features are passed through average pooling and linear layers to create an embedding for a downstream task.

The GC ViT architecture benefits from novel blocks such as *a downsampling operator*, *a global query generator* and *a global self-attention module* described in the next sections.

Downsampler. We leverage an idea of spatial feature contraction from CNN models that imposes locality bias and cross channel interaction while reducing dimensions. We utilize a modified Fused-MBConv block, followed by a max pooling layer with a kernel size of 3 and stride of 2 as a downsampling operator. The Fused-MBConv block in our work is similar to the one in EfficientNetV2 (Tan & Le, 2021) with modifications as in

$$\begin{aligned} \hat{\mathbf{x}} &= \text{DW-Conv}_{3\times3}(\mathbf{x}), \\ \hat{\mathbf{x}} &= \text{GELU}(\hat{\mathbf{x}}), \\ \hat{\mathbf{x}} &= \text{SE}(\hat{\mathbf{x}}), \\ \mathbf{x} &= \text{Conv}_{1\times1}(\hat{\mathbf{x}}) + \mathbf{x}, \end{aligned} \tag{1}$$

where SE, GELU and DW-Conv$_{3\times3}$ denote Squeeze and Excitation block (Hu et al., 2018), Gaussian Error Linear Unit (Hendrycks & Gimpel, 2016) and 3×3 depth-wise convolution, respectively. In our proposed architecture, the Fused-MBConv blocks provide desirable properties such as inductive bias and modeling of inter-channel dependencies. It is ablated in Table 8.

2.1. Global Self-Attention

Fig. 3 demonstrates the main idea behind our contribution. Local self-attention can only query patches within a local window, whereas the global attention can query different image regions while still operating within the window. At each stage, the global query component is pre-computed. The global self-attention utilizes the extracted global query

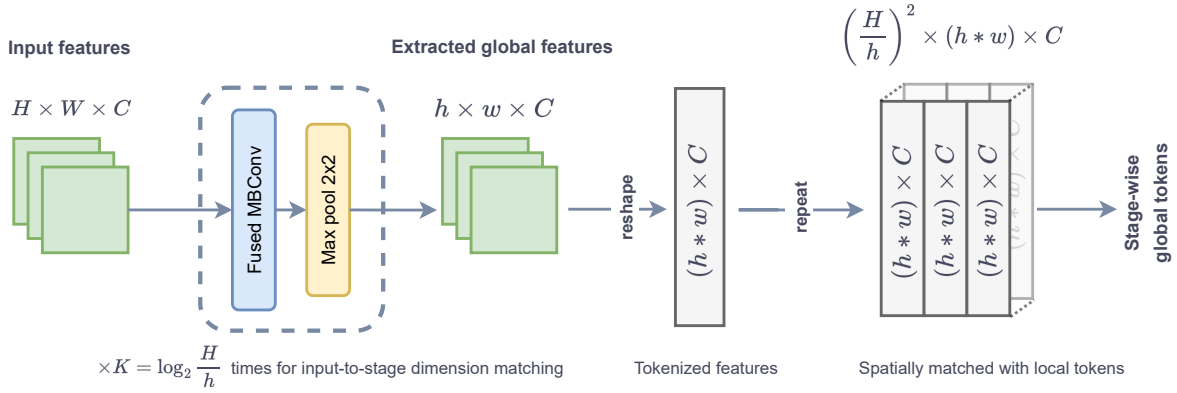


Figure 4 – Global query generator schematic diagram. It is designed to (i) transform an input feature map to the current stage of dimension H, W, C denoting height, width, and channel respectively, (ii) extract features via repeating the modified Fused MBConv block, joint with down-sampling, $\log_2 \frac{H}{h}$ times for dimension matching to local window size h (iii) output is reshaped and repeated to $(\frac{H}{h})^2$ number of local tokens that can attend to global contextual information. $\star$ denotes merged dimensions during reshaping.

tokens and shared across all blocks, to interact with the local key and value representations. In addition, GC ViT employs alternating local and global self-attention blocks to effectively capture both local and global spatial information. Fig. S.1 illustrates the difference between local and global self-attention. The global attention query $\mathbf{q_g}$ has a size of $B \times C \times h \times w$, wherein B, C, h and w denote batch size, embedding dimension, local window height and width, respectively. Moreover, $\mathbf{q_g}$ is repeated along the batch dimension to compensate for the overall number of windows and aggregated batch size $B^* = B \times N^*$ where N^* is the number of local windows. $\mathbf{q_g}$ is further reshaped into multiple heads. The value and key are computed within each local window using a linear layer.

Algorithm. 1 Global Attention Pseudocode

```
# Input/output shape: (B*, N, C);
# B*: Aggregated Batch Size; H: Height;
# W: Width; C: dim; q_g: Global Token;
# F: Num Attention Head; N: H x W.
def init():
    f = nn.Linear(C, 2*C)
    softmax = nn.Softmax(dim=-1)

def forward(x, q_g):
    B*, N, C = x.shape
    B, C, h, w = q_g.shape
    kv = f(x).reshape(B*, N, 2, F, C // F)
    kv = kv.permute(2, 0, 3, 1, 4)
    k, v = split(kv, (1, 1), 0)
    q_g = q_g.repeat(1, B* // B, 1, 1)
    q_g = q_g.reshape(B*, F, N, C // F)
    qk = matmul(q_g,k.transpose(-2, -1))
    attn = softmax(qk)
    return matmul(attn, v).reshape(B*, N, C)
```

Since the partitioned windows only contain local information, interaction with rich contextual information embedded in the global query tokens provides an effective way of enlarging the receptive field and attending to various regions in the input feature maps. The self-attention module is computed as in

$$\text{Attention}(\mathbf{q_g}, \mathbf{k}, \mathbf{v}) = \text{Softmax}(\frac{\mathbf{q_g k}}{\sqrt{\mathbf{d}}} + \mathbf{b})\mathbf{v}, \quad (2)$$

where $\mathbf{d}$ is scaling factor and $\mathbf{b}$ is a learnable relative position bias term. Assuming position change between $[-p+1, p-1]$ along horizontal and vertical axes, $\mathbf{b}$ is sampled from the grid $\hat{\mathbf{b}} \in \mathbb{R}^{(2p-1)\times(2p-1)}$. As shown in Sec. 4, relative position bias improves the performance, especially for dense prediction downstream tasks. In Algorithm 1, we present a PyTorch-like pseudocode for computing global self-attention in GC ViT.

2.2. Complexity Analysis

Given an input feature map of $x \in \mathcal{R}^{H\times W\times C}$ at each stage with a window size of $h \times w$, the computational complexity of GC ViT is as follows

$$\mathcal{O}(\text{GC ViT}) = 2HW(2C^2 + hwC), \quad (3)$$

The efficient design of global query token generator and other components allows to maintain a similar computational complexity in comparison to Swin Transformer (Liu et al., 2021) while being able to capture long-range information and achieve better higher accuracy for classification and downstream tasks such as detection and segmentation.

3. Experiments

For image classification, we trained and tested our model on ImageNet-1K dataset (Deng et al., 2009). To allow for a fair comparison, all GC ViT variants are trained by following training configurations of previous efforts (Liu et al., 2021; Yang et al., 2021b; Chu et al., 2021a). Specifically, all models are trained with the AdamW (Kingma & Ba, 2014)

Table 1 – Image classification benchmarks on **ImageNet-1K** dataset (Deng et al., 2009). Models that are trained on ImageNet-1K dataset and without any pre-training or usage of extra data are considered.

Model	Param (M)	FLOPs (G)	Image Size	Top-1 (%)
ConvNet				
ResNet50 (He et al., 2016)	25	4.1	224^2	76.1
ResNet-101 (He et al., 2016)	44	7.9	224^2	77.4
ResNet-152 (He et al., 2016)	60	11.6	224^2	78.3
EfficientNetV2-B2 (Tan & Le, 2021)	10	1.6	260^2	80.2
EfficientNetV2-B3 (Tan & Le, 2021)	14	2.9	300^2	82.0
EfficientNetV2-S (Tan & Le, 2021)	21	8.0	384^2	83.9
RegNetY-040 (Radosavovic et al., 2020)	20	6.6	288^2	83.0
RegNetY-064 (Radosavovic et al., 2020)	30	10.5	288^2	83.7
ConvNeXt-T (Liu et al., 2022b)	29	4.5	224^2	82.1
ConvNeXt-S (Liu et al., 2022b)	50	8.7	224^2	83.1
ConvNeXt-B (Liu et al., 2022b)	89	15.4	224^2	83.8
ConvNeXt-L (Liu et al., 2022b)	198	34.4	224^2	84.3
Transformer				
ViT-B (Dosovitskiy et al., 2020)	86	17.6	224^2	77.9
DeiT-S/16 (Touvron et al., 2021)	22	4.6	224^2	79.9
DeiT-B (Touvron et al., 2021)	86	17.6	224^2	81.8
Swin-T (Liu et al., 2021)	29	4.5	224^2	81.3
Swin-S (Liu et al., 2021)	50	8.7	224^2	83.0
Swin-B (Liu et al., 2021)	88	15.4	224^2	83.3
Twins-S (Chu et al., 2021a)	24	2.8	224^2	81.7
Twins-B (Chu et al., 2021a)	56	8.3	224^2	83.1
Twins-L (Chu et al., 2021a)	99	14.8	224^2	83.7
Focal-T (Yang et al., 2021b)	29	4.9	224^2	82.2
Focal-S (Yang et al., 2021b)	51	9.1	224^2	83.5
Focal-B (Yang et al., 2021b)	90	16.0	224^2	83.8
PoolFormer-S36 (Yu et al., 2022)	31	5.0	224^2	81.4
PoolFormer-M36 (Yu et al., 2022)	56	8.8	224^2	82.1
PoolFormer-M58 (Yu et al., 2022)	73	11.6	224^2	82.4
SwinV2-T (Liu et al., 2022a)	28	4.4	256^2	81.8
SwinV2-S (Liu et al., 2022a)	49	8.5	256^2	83.8
SwinV2-B (Liu et al., 2022a)	88	15.1	256^2	84.6
Hybrid				
CrossViT-S (Chen et al., 2021)	27	5.1	224^2	81.0
CrossViT-B (Chen et al., 2021)	105	20.1	224^2	82.2
CoAtNet-0 (Dai et al., 2021)	25	4.2	224^2	81.6
CoAtNet-1 (Dai et al., 2021)	42	8.4	224^2	83.3
CoAtNet-2 (Dai et al., 2021)	42	8.4	224^2	83.3
CoAtNet-3 (Dai et al., 2021)	168	34.7	224^2	84.5
PVT-v2-B2 (Wang et al., 2022)	25	4.0	224^2	82.0
PVT-v2-B3 (Wang et al., 2022)	45	6.9	224^2	83.2
PVT-v2-B5 (Wang et al., 2022)	82	11.8	224^2	83.8
CSwin-T (Dong et al., 2022)	23	4.3	224^2	82.7
CSwin-S (Dong et al., 2022)	35	6.9	224^2	83.6
CSwin-B (Dong et al., 2022)	78	15.0	224^2	84.2
MaxViT-T (Tu et al., 2022)	31	5.6	224^2	83.6
MaxViT-S (Tu et al., 2022)	69	11.7	224^2	84.4
MaxViT-B (Tu et al., 2022)	120	74.2	224^2	84.9
MaxViT-L (Tu et al., 2022)	212	43.9	224^2	85.1
GC ViT				
GC ViT-XXT	12	2.1	224^2	**79.9**
GC ViT-XT	20	2.6	224^2	**82.0**
GC ViT-T	28	4.7	224^2	**83.5**
GC ViT-T2	34	5.5	224^2	**83.7**
GC ViT-S	51	8.5	224^2	**84.3**
GC ViT-S2	68	10.7	224^2	**84.8**
GC ViT-B	90	14.8	224^2	**85.0**
GC ViT-L	201	32.6	224^2	**85.7**

optimizer for 300 epochs with an initial learning rate of 0.001, weight decay of 0.05, cosine decay scheduler and 20 warm-up and cool-down epochs, respectively.

For object detection and instance segmentation, we trained our model on MS COCO (Lin et al., 2014) with DINO (He et al., 2017) and a Mask-RCNN (He et al., 2017) heads, using $\times 3$ LR schedule with an initial learning rate of 0.0001, a batch size of 16 and weight decay of 0.05. Following (Liu et al., 2022b), we compared against Tiny, Small and Base model variants using Cascade Mask-RCNN but only compared against Tiny variant using Mask-RCNN. For semantic segmentation, we used the ADE20K dataset (Zhou et al., 2017) with a UPerNet (Xiao et al., 2018) segmentation head. Following previous efforts, we used a random crop size of 512×512 for the input images.

3.1. Classification

We present the ImageNet-1K classification benchmarks in Table 1 and compare against CNN and ViT-based models across different model sizes. Our model achieves better performance when compared to other established benchmarks such as ConvNeXt (Liu et al., 2022b). Furthermore, as shown in Fig. 1, GC ViT models have better or comparable computational efficiency in terms of number FLOPsover the competing counterpart models.

3.2. Detection and Instance Segmentation

In Table 2, we present object detection and instance segmentation benchmarks on MS COCO dataset. Using a Mask-RCNN head, the model with pre-trained GC ViT-T (47.9/43.2) backbone outperforms counterparts with pre-trained ConvNeXt-T (Liu et al., 2022b) (46.2/41.7) by +1.7 and +1.5 and Swin-T (Liu et al., 2021) (46.0/41.6) by +1.9 and +1.6 in terms of box AP and mask AP, respectively. Using a Cascade Mask-RCNN head, the models with pre-trained GC ViT-T (51.6/44.6) and GC ViT-S (52.4/45.4) backbones outperform ConvNeXt-T (Liu et al., 2022b) (50.4/43.7) by +1.2 and +0.9 and ConvNeXt-S (Liu et al., 2022b) (51.9/45.0) by +0.5 and +0.4 in terms of box AP and mask AP, respectively. Furthermore, the model with GC ViT-B (52.9/45.8) backbone outperforms the counterpart with ConvNeXt-B (Liu et al., 2022b) (52.7/45.6) by +0.2 and +0.2 in terms of box AP and mask AP, respectively.

As shown in Table 2, we have also tested the performance of GC ViT-L model, pre-trained on ImageNet-21K dataset, with a 4-scale DINO (Zhang et al., 2022) detection head and achieved a box AP of **58.3%** on MS COCO dataset. Hence our model outperforms the counterpart with Swin-L backbone.

3.3. Semantic Segmentation

We present semantic segmentation benchmarks on ADE20K dataset in Table 4. The models using pre-trained GC ViT-T (47.0), GC ViT-S (48.3) and GC ViT-B (49.2) backbones outperform counterpart models with pre-trained Twins-SVT-S (Chu et al., 2021a) (46.2), Twins-SVT-B (Chu et al., 2021a) (47.7) and Twins-SVT-L (Chu et al., 2021a) (48.8) by +0.8, +0.6 and +0.4 in terms of mIoU, respectively. In addition, models with GC ViT backbones significantly outperform counterparts with Swin Transformer backbones, hence demonstrating the effectiveness of the global self-

Table 2 – Object detection and instance segmentation benchmarks using Mask R-CNN and Cascade Mask R-CNN on **MS COCO** dataset (Lin et al., 2014). All models employ 3× schedule.

Backbone	Param (M)	FLOPs (G)	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	AP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}
Mask-RCNN 3× schedule								
Swin-T (Liu et al., 2021)	48	267	46.0	68.1	50.3	41.6	65.1	44.9
ConvNeXt-T (Liu et al., 2022b)	48	262	46.2	67.9	50.8	41.7	65.0	44.9
GC ViT-T	48	291	**47.9**	**70.1**	**52.8**	**43.2**	**67.0**	**46.7**
Cascade Mask-RCNN 3× schedule								
DeiT-Small/16 (Touvron et al., 2021)	80	889	48.0	67.2	51.7	41.4	64.2	44.3
ResNet-50 (He et al., 2016)	82	739	46.3	64.3	50.5	40.1	61.7	43.4
Swin-T (Liu et al., 2021)	86	745	50.4	69.2	54.7	43.7	66.6	47.3
ConvNeXt-T (Liu et al., 2022b)	86	741	50.4	69.1	54.8	43.7	66.5	47.3
GC ViT-T	85	770	**51.6**	**70.4**	**56.1**	**44.6**	**67.8**	**48.3**
X101-32 (Xie et al., 2017)	101	819	48.1	66.5	52.4	41.6	63.9	45.2
Swin-S (Liu et al., 2021)	107	838	51.9	70.7	56.3	45.0	68.2	48.8
ConvNeXt-S (Liu et al., 2022b)	108	827	51.9	70.8	56.5	45.0	68.4	49.1
GC ViT-S	108	866	**52.4**	**71.0**	**57.1**	**45.4**	**68.5**	**49.3**
X101-64 (Xie et al., 2017)	140	972	48.3	66.4	52.3	41.7	64.0	45.1
Swin-B (Liu et al., 2021)	145	982	51.9	70.5	56.4	45.0	68.1	48.9
ConvNeXt-B (Liu et al., 2022b)	146	964	52.7	71.3	57.2	45.6	68.9	49.5
GC ViT-B	146	1018	**52.9**	**71.7**	**57.8**	**45.8**	**69.2**	**49.8**

Backbone	Head	Scale	AP^{box}
ResNet-50 (He et al., 2016)	DINO (Zhang et al., 2022)	4	50.9
ResNet-50 (He et al., 2016)	DINO (Zhang et al., 2022)	5	51.2
Swin-L‡ (Liu et al., 2021)	DINO (Zhang et al., 2022)	4	58.0
GC ViT-L‡	DINO (Zhang et al., 2022)	4	**58.3**

Table 3 – Object detection benchmarks using DINO (Zhang et al., 2022) network on **MS COCO** dataset (Lin et al., 2014). ‡ denotes models that are pre-trained on ImageNet-21K dataset.

attention.

4. Ablation

Component-wise Analysis. As shown in Table 5, we study the role of each component in GC ViT model for classification, detection, instance and semantic segmentation. For simplicity, we start with Swin Transformer as the base model and progressively re-design the components to demonstrate their importance in improving the performance. Firstly, we remove the window shifting and predictably observe significant performance degradation across all tasks. Changing distribution of parameters to our design improves the performance by +1.7, +2.8, +2.2 and +1.7 in terms of accuracy, box AP, mask AP and mIoU. Such reparametrization includes changing the window size, MLP ratio, number of layers to name but a few. Adding the CNN-based stem of GC ViT to the model provides additional improvements of +0.3, +0.2, +0.2 and +0.2 in terms of accuracy, box AP, mask AP and mIoU. In addition, using our proposed downsampler further improves the accuracy, box AP, mask AP and

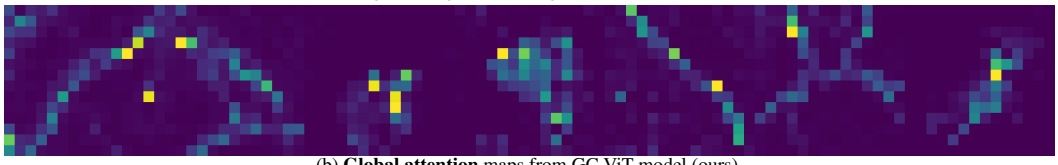
Backbone	Param (M)	FLOPs (G)	mIoU
DeiT-Small/16 (Touvron et al., 2021)	52	1099	44.0
Swin-T (Liu et al., 2021)	60	945	44.5
ResNet-101 (He et al., 2016)	86	1029	44.9
Focal-T (Yang et al., 2021b)	62	998	45.8
Twins-SVT-S (Chu et al., 2021a)	55	-	46.2
GC ViT-T	58	947	**47.0**
Swin-S (Liu et al., 2021)	81	1038	47.6
Twins-SVT-B (Chu et al., 2021a)	89	-	47.7
Focal-S (Yang et al., 2021b)	85	1130	48.0
GC ViT-S	84	1163	**48.3**
Swin-B (Liu et al., 2021)	121	1188	48.1
Twins-SVT-L (Chu et al., 2021a)	133	-	48.8
Focal-B (Yang et al., 2021b)	126	1354	49.0
GC ViT-B	125	1348	**49.2**

Table 4 – Semantic segmentation benchmarks **ADE20K** validation set with UPerNet (Xiao et al., 2018) and pre-trained ImageNet-1K backbone. All models use a crop size of 512×512 and use single-scale inference.

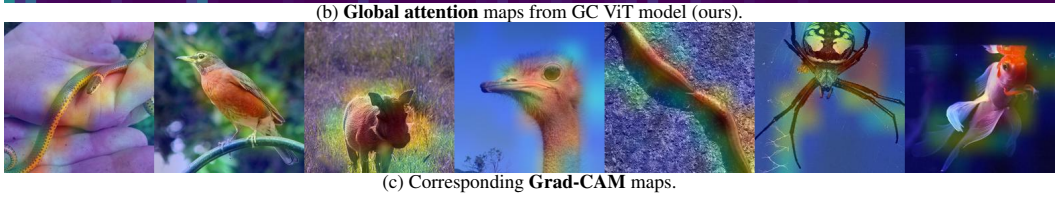
	ImageNet	COCO		ADE20k
	top-1	AP^{box}	AP^{mask}	mIoU
Swin-T	81.3	50.4	43.7	44.5
Swin-T w/o Window Shifting	80.2	47.7	41.5	43.3
+ Reparam. (window, #blocks, ratio)	81.9	50.5	43.7	45.0
+ GC ViT-T Stem	82.2	50.7	43.9	45.2
+ GC ViT-T Down-sampler	82.6	50.8	44.0	45.8
+ GC ViT-T Global Self-attention	**83.5**	**51.6**	**44.6**	**47.0**

Table 5 – Ablation study on the effectiveness of various components in GC ViT on classification, detection and segmentation performance.

(a) Original images from ImageNet-1K validation set.



(b) **Global attention** maps from GC ViT model (ours).



(c) Corresponding **Grad-CAM** maps.

Figure 5 – Visualization of : (a) input images (b) global self-attention maps from GC ViT-T model (c) corresponding Grad-CAM attention maps. Both short and long-range spatial dependencies are captured effectively.

Model	Param (M)	FLOPs (G)	Top-1 (%)
Swin-L (Liu et al., 2021)	197	34.5	86.3
CSwin-L (Dong et al., 2022)	173	31.5	86.5
ConvNeXt-L (Liu et al., 2022b)	198	34.4	**86.6**
GC ViT-L	201	32.6	**86.6**

Table 6 – Classification benchmarks of **ImageNet-21K** trained models on **ImageNet-1K** dataset (Deng et al., 2009).

Model	Accuracy-Matched Frequency	Accuracy-Threshold-0.7
GC ViT-XT	71.3	78.8
GC ViT-T	73.1	80.5
GC ViT-S	73.8	80.7
GC ViT-B	74.4	81.1
GC ViT-L	74.9	81.8

Table 7 – Classiication benchmarks of GC ViT models on ImageNetV2 dataset.

Down-sampler	Architecture	Top-1
Conv	Conv (s=1), Maxpool	82.7
Swin	Linear	82.9
GC ViT	Modified Fused-MBConv (s=2)	**83.5**

Table 8 – Ablation study on the effectiveness of down-sampler in GC ViT architecture on ImageNet Top-1 accuracy.

mIoU by +0.4, +0.1, +0.1 and +0.3, respectively. The last two changes demonstrate the importance of convolutional inductive bias and capturing the inter-channel dependencies in our model. Finally, leveraging the proposed global self-attention improves the performance by by +0.9, +0.8, +0.6 and +1.2 in terms of accuracy, box AP, mask AP and mIoU. Hence, this validates the effectiveness of the proposed global self-attention, in particular for downstream tasks with high resolution images such as semantic segmentation in which modeling long-range spatial dependencies is critical.

4.1. ImageNet-21K

In Table 6, we compare the performance of GC ViT-L model which pretrained on ImageNet-21K dataset and finetuned on ImageNet-1K dataset with counterpart approaches. GC ViT-L outperforms Swin-L and CSwin-L by +0.3% and +0.1% in terms of Top-1 accuracy respectively, while performing on-par with ConvNeXt-L model. As a result, it validates the effectiveness of the model in large-scale data regimes.

4.2. Generalizability

In Table 7, we have evaluated the performance of GC ViT on ImageNetV2 dataset (Recht et al., 2019) to further measure its robustness. Specifically, we have used different sampling strategies of Matched Frequency and Threshold-0.7. These benchmarks demonstrate the competetive performance of GC ViT on ImageNetV2 dataset and validates its effectiveness in robustness and generalizability.

4.3. Downsampler Design

We studied the effectiveness of various downsampler blocks in Table 8. The simplest alternative to our design is a pair of convolutional and maxpooling layers. However, it results in a reduction of ImageNet Top-1 accuracy by -0.8. Patch merging is another variant which was introduced in Swin Transformers (Liu et al., 2021).

However, it reduces the accuracy by -0.6. Finally, our downsampler which consists of a modified Fused-MBConv block and strided convolution and shows the best result. Impor-

tance of the former component is explained by the SE operation which boosts cross channel interaction while keeping number of parameters and FLOPs low. We conclude that our proposed down-sampler is essential to achieve high accuracy as it introduces convolutional inductive bias.

5. Interpretability

To provide further insights on interpretability of the proposed global self-attention and query tokens, we demonstrate visualization of the learned attention and Grad-CAM (Selvaraju et al., 2017) maps in Fig. 5. The associated attention distributions uncovered by the global self-attention modules align with image semantics, and hence act as an informative source for local attention modules. In addition, corresponding Grad-CAM maps demonstrate accurate object localization with most intricate details.

6. Related work

ConvNet. Since the advent of deep learning, CNNs (Krizhevsky et al., 2012; Simonyan & Zisserman, 2014; Howard et al., 2017; He et al., 2016; Szegedy et al., 2016; Huang et al., 2017; Hu et al., 2018) have dominated computer vision benchmarks with SOTA performance. Recently, ConvNeXt (Liu et al., 2022b) proposed modifications to the architecture of ResNet (He et al., 2016), and achieved competitive benchmarks for classification, detection and segmentation tasks.

Transformer. The ViT (Dosovitskiy et al., 2020) was first proposed as an alternative to CNNs with the advantage of enlarged receptive field, due to its self-attention layers. However, it lacked desirable properties of CNNs such as inductive biases and translation invariance and required large-scale training datasets to achieve competitive performance. Data-efficient Image Transformers (DeiT) (Touvron et al., 2021) introduced a distillation-based training strategy which significantly improved the classification accuracy.

Hybrid. LeViT (Graham et al., 2021) proposed a hybrid model with re-designed multi-layer perceptron (MLP) and self-attention modules that are highly-optimized for fast inference. Cross-covariance Image Transformer (XCiT) (Ali et al., 2021) introduced a transposed self-attention module for modeling the interactions of feature channels. Convolutional vision Transformer (CvT) (Wu et al., 2021) introduced convolutional token embedding layer and Transformer block in a hierarchical architecture to improve the efficiency and accuracy of ViTs. Conditional Position encoding Vision Transformer (CPVT) (Chu et al., 2021b) demonstrated improved performance on various tasks such as image classification and object detection by conditioning the position encoding on localized patch token. Tokens-To-Token Vision Transformer (T2T-ViT) (Yuan et al., 2021) proposed a transformation layer for aggregating adjacent tokens and establishing image prior by exploiting spatial correlations. Pyramid Vision Transformer (PVT) (Wang et al., 2021) proposed a hierarchical architecture with patch embedding at the beginning of each stage and spatial dimension reduction to improve the computational efficiency. Independently, Swin Transformers (Liu et al., 2021) also proposed a hierarchical architecture in which self-attention is computed within local windows which are shifted for region interaction. Twins Transformer (Chu et al., 2021a) proposed a spatially separable self-attention with locally-grouped and global sub-sampling modules to improve the efficiency.

Global Attention. Other efforts such as EdgeViT (Pan et al., 2022) in computer vision and BigBird (Zaheer et al., 2020) in NLP have proposed global self-attention in their respective applications. The global attention in GC ViT is fundamentally different than these approaches. For instance, EdgeViT samples representative tokens and only computes sparse self-attention between these representative tokens with reduced feature size. On the contrary, GC ViT computes self-attention between the global queries (not just the token) and local keys and values without any subsampling in their respective local regions. Furthermore, in EdgeViT, only subsampled representative tokens per region interact In the self-attention module; however, in GC ViT, the global queries interact with the entire local regions. Furtermore, BigBird uses a combination of random, window and global attention mechanisms, which is different from the proposed local and global self-attention scheme in GC ViT. BigBird does not have any specific mechanisms for extracting global tokens as the existing tokens or additional special tokens can be specified as global tokens. However, the global tokens in GC ViT are obtained by the query generator via extracting contextual information from the entire input features. Lastly, BigBird employs a set of global tokens which attend to the entire input sequence. However, in GC ViT, the global query tokens attend to local key and value tokens in partitioned windows, since attending to the entire input sequence is not feasible considering the larger size of input features.

7. Conclusion

In this work, we introduced a novel hierarchical ViT, referred to as GC ViT, which can efficiently capture global context by utilizing global query tokens and interact with local regions. We have extensively validated the effectiveness of our model on various tasks. The proposed GC ViT model achieves new SOTA benchmarks for image classification across various model sizes on ImageNet-1K dataset, and surpasses both CNN and ViT-based counterparts by a significant margin. We have also achieved SOTA or competitive performance for downstream tasks of detection and semantic segmentation on high-resolution images.

References

Ali, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al. Xcit: Cross-covariance image transformers. *Advances in neural information processing systems*, 34, 2021.

Chen, C.-F., Fan, Q., and Panda, R. Crossvit: Cross-attention multi-scale vision transformer for image classification, 2021.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., and Shen, C. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021a.

Chu, X., Tian, Z., Zhang, B., Wang, X., Wei, X., Xia, H., and Shen, C. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021b.

Contributors, M. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. `https://github.com/open-mmlab/mmsegmentation`, 2020.

Dai, Z., Liu, H., Le, Q. V., and Tan, M. Coatnet: Marrying convolution and attention for all data sizes. *Advances in Neural Information Processing Systems*, 34:3965–3977, 2021.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., and Guo, B. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12124–12134, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

Graham, B., El-Nouby, A., Touvron, H., Stock, P., Joulin, A., Jégou, H., and Douze, M. Levit: a vision transformer in convnet's clothing for faster inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12259–12269, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Howard, A. G., Zhu, M., and Chen, B. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2017.

Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

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

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

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*, pp. 10012–10022, 2021.

Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12009–12019, 2022a.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. A convnet for the 2020s. *arXiv preprint arXiv:2201.03545*, 2022b.

Pan, J., Bulat, A., Tan, F., Zhu, X., Dudziak, L., Li, H., Tzimiropoulos, G., and Martinez, B. Edgevits: Competing light-weight cnns on mobile devices with vision transformers. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XI*, pp. 294–311. Springer, 2022.

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., and Dollár, P. Designing network design spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10428–10436, 2020.

Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., and Dosovitskiy, A. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34, 2021.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. URL `http://arxiv.org/abs/1409.1556`.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Tan, M. and Le, Q. Efficientnetv2: Smaller models and faster training. In *International Conference on Machine Learning*, pp. 10096–10106. PMLR, 2021.

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*, pp. 10347–10357. PMLR, 2021.

Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., and Li, Y. Maxvit: Multi-axis vision transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV*, pp. 459–479. Springer, 2022.

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*, pp. 5998–6008, 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*, pp. 568–578, 2021.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., and Shao, L. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022.

Wightman, R. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., and Zhang, L. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22–31, October 2021.

Xiao, T., Liu, Y., Zhou, B., Jiang, Y., and Sun, J. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 418–434, 2018.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

Yang, H., Yin, H., Molchanov, P., Li, H., and Kautz, J. NViT: Vision transformer compression and parameter redistribution. *arXiv preprint arXiv:2110.04869*, 2021a.

Yang, J., Li, C., Zhang, P., Dai, X., Xiao, B., Yuan, L., and Gao, J. Focal attention for long-range interactions in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021b.

Yin, H., Vahdat, A., Alvarez, J., Mallya, A., Kautz, J., and Molchanov, P. A-ViT: Adaptive tokens for efficient vision transformer. *arXiv preprint arXiv:2112.07658*, 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*, pp. 10819–10829, 2022.

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

Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*, 2020.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L. M., and Shum, H.-Y. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

A. Appendix

A.1. GC ViT Model Configurations

GC ViT model configurations are presented in Table S.1 describing the choice of internal hyper parameters to obtain models with various compute load and parameter number.

	Output Size (Downs. Rate)	GC ViT-XT	GC ViT-T	GC ViT-S	GC ViT-B
Stem	112×112 (2×)	Conv, C:64, S:2, LN	Conv, C:64, S:2, LN	Conv, C:96, S:2, LN	Conv, C:128, S:2, LN
		[F-MBConv C:64] × 1	[F-MBConv C:64] × 1	[F-MBConv C:96] × 1	[F-MBConv C:128] × 1
Stage 1	56×56 (4×)	Conv, C:128, S:2, LN	Conv, C:128, S:2, LN	Conv, C:192, S:2, LN	Conv, C:256, S:2, LN
		[LG-SA, C:64, head:2] × 3, F-MBConv, C:128	[LG-SA, C:64, head:2] × 3, F-MBConv, C:128	[LG-SA, C:96, head:3] × 3, F-MBConv, C:192	[LG-SA, C:128, head:4] × 3, F-MBConv, C:256
Stage 2	28×28 (8×)	Conv, C:256, S:2, LN	Conv, C:256, S:2, LN	Conv, C:384, S:2, LN	Conv, C:512, S:2, LN
		[LG-SA, C:64, head:4] × 4, F-MBConv, C:256	[LG-SA, C:64, head:4] × 4, F-MBConv, C:256	[LG-SA, C:96, head:6] × 4, F-MBConv, C:384	[LG-SA, C:128, head:8] × 4, F-MBConv, C:512
Stage 3	14×14 (16×)	Conv, C:512, S:2, LN	Conv, C:512, S:2, LN	Conv, C:768, S:2, LN	Conv, C:1024, S:2, LN
		[LG-SA, C:64, head:8] × 6 , F-MBConv, C:512	[LG-SA, C:64, head:8] × 19, F-MBConv, C:512	[LG-SA, C:96, head:12] × 19, F-MBConv, C:768	[LG-SA, C:128, head:16] × 19, F-MBConv, C:1024
Stage 4	7×7 (32×)	Conv, C:1024, S:2, LN	Conv, C:1024, S:2, LN	Conv, C:1536, S:2, LN	Conv, C:2048, S:2, LN
		[LG-SA, C:64, head:16] × 5, F-MBConv, C:1024	[LG-SA, C:64, head:16] × 5, F-MBConv, C:1024	[LG-SA, C:96, head:24] × 5, F-MBConv, C:1536	[LG-SA, C:128, head:32] × 5, F-MBConv, C:2048

Table S.1 – Architecture configurations for GC ViT. LG-SA and Conv denotes local, global self-attention and 3×3 convolutional layer, respectively. GC ViT-XT, GC ViT-T, GC ViT-S and GC ViT-B denote XTiny, Tiny, Small and Base variants, respectively.

A.2. Ablation

A.2.1. Global Query

We performed ablation studies to validate the effectiveness of the proposed global query. Using the same architecture, instead of global query, we compute: (1) global key and value features and interact them with local query (2) global value features and interact it with local query and key. As shown in Table S.2, replacing global query may significantly impact the performance for image segmentation and downstream tasks such as object detection, instance segmentation and semantic segmentation.

	ImageNet	COCO		ADE20k
	top-1	AP^{box}	AP^{mask}	mIoU
w. Global KV	82.5	49.9	41.3	44.6
w. Global V	82.7	50.8	42.4	45.1
GC ViT-T	**83.5**	**51.6**	**44.6**	**47.0**

Table S.2 – Ablation study on the effectiveness of the proposed global query for classification, detection and segmentation.

A.2.2. Effect of Global Context Module

In Fig. S.1, we illustrate the difference between GC ViT local and global attention blocks. In order to demonstrate the effectiveness of Global Context (GC) self-attention module, we use Swin Transformers as the base model and add our propoped GC module. In this analysis, we remove the window shifting operation from Swin Transformers, since GC module is capable of modeling cross-region interactions. As shown in Table S.3, addition of GC module improves the ImageNet Top-1 accuracy by $+0.9\%$ and $+0.7\%$ for Swin Transformers Tiny and Small variants respectively.

Model	Added Component	Top-1
Swin-T	None	81.3
Swin-T	GC Module	82.2
Swin-S	None	83.0
Swin-S	GC Module	83.7

Table S.3 – Ablation study on the effectiveness of Global Context (GC) module in Swin Transformers architecture on ImageNet Top-1 accuracy.

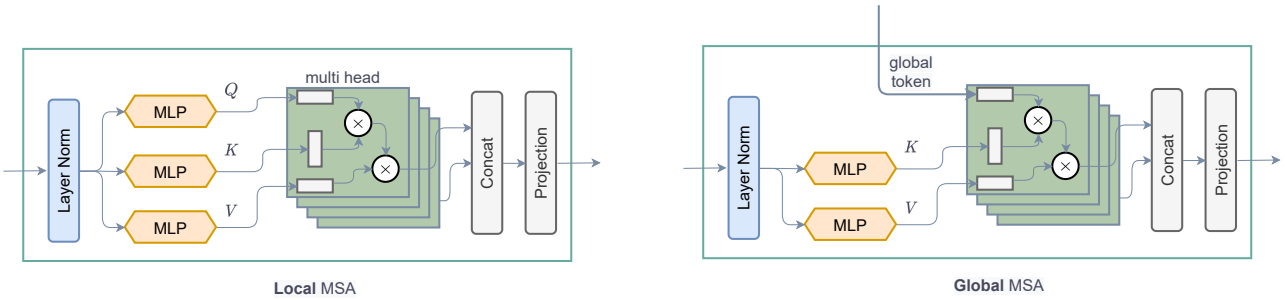
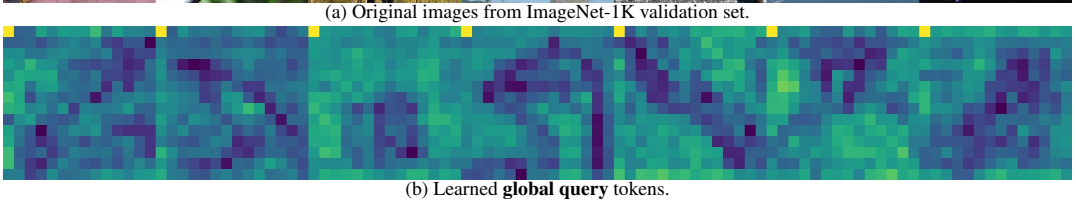


Figure S.1 – Local and global attention blocks. Global attention block does not compute query vector and reuses global query computed via Global Token Generation.

(a) Original images from ImageNet-1K validation set.



(b) Learned **global query** tokens.

Figure S.2 – Visualization of : (a) input images (b) learned global query token feature maps.

A.2.3. EMA AND BATCH SIZE

We also used used Exponential Moving Averages (EMA) and observed slight improvement in terms of ImageNet TOp-1 accuracy. Furthermore, the performance of the model across different batch sizes were stable as we did not observe significant changes. Table S.4 demonstrates the effect of EMA and batch size on the accuracy of a GCViT-T model.

A.3. Training Details

For image classification, GC ViT models were trained using four computational nodes with 32 NVIDIA A100 GPUs. The total training batch size is 1024 (32 per GPU) for GC ViT-S, GC ViT-B, GC ViT-L and 4096 (128 per GPU) for GC ViT-XXT, GC ViT-XT and GC ViT-T. On average, each model required 32 hours of training with the specified hyper-parameters as indicated in the paper. All classification models were trained using the `timm` package (Wightman, 2019). Object detection and instance segmentation models as well as semantic segmentation models were trained using one computational node with 8 NVIDIA A40 GPUs using a total batch size of 16, hence a batch size of 2 per GPU. Detection and instance segmentation models were trained using `mmdetection` (Chen et al., 2019) package and on average required 56 hours of training. Semantic segmentation models were trained using `mmsegmentation` (Contributors, 2020) package, and on average required 34 hours of training.

Model	Local Batch Size	Global Batch Size	EMA	Top-1
GC ViT-T	32	1024	No	83.45
GC ViT-T	128	4096	No	83.46
GC ViT-T	32	1024	Yes	83.47
GC ViT-T	128	4096	Yes	83.48

Table S.4 – Ablation study on the effect of EMA and batch size on GC ViT-T ImageNet Top-1 accuracy.

A.4. Interpretability

In Fig. S.2, we illustrate the learned global query token maps and demonstrate their effectiveness in capturing long-range contextual representations from different image regions.