GLOBAL CONTEXT VISION TRANSFORMERS

Anonymous authors
Paper under double-blind review

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 yet 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 tiny, small and base variants of GC ViT with 28M, 51M and 90M parameters achieve **83.4%**, **83.9%** and **84.5%** Top-1 accuracy, respectively, surpassing comparably-sized prior art such as CNN-based ConvNeXt and ViT-based 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, sometimes by large margins. Code and pre-trained models available at https://anonymous.4open.science/r/GCViTs-8852/.

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 application 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., 2021; 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 and cross-window connections such as window shifting are used for modeling the interactions across different regions. Despite this progress, the limited receptive field of local windows baffles 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. The burden of computationally expensive operations such as token unfolding and rolling and additional computations of key and values can significantly affect the run-time and throughput.

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 modified fused inverted residual

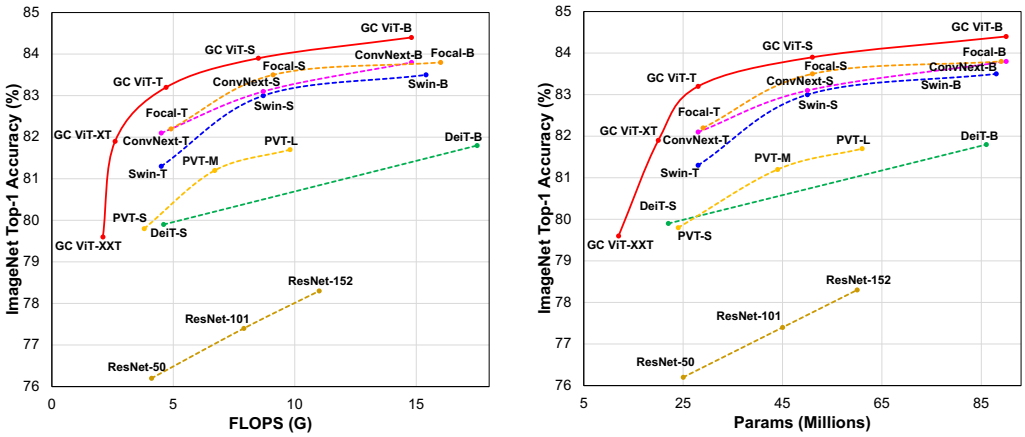


Figure 1 – Top-1 accuracy *vs.* model FLOPs/parameter size on ImageNet-1K dataset. GC ViT achieves new SOTA benchmarks for different model sizes as well as FLOPs, outperforming competing approaches by a significant margin.

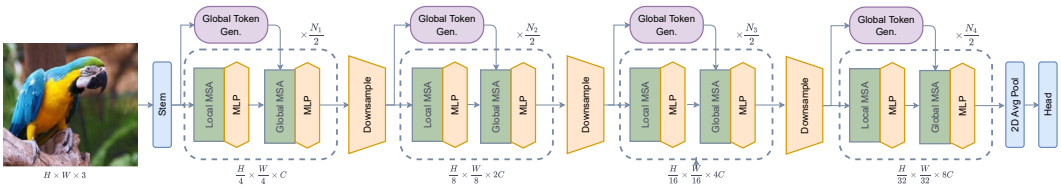


Figure 2 – Architecture of the proposed Global Context ViT. We use alternating blocks of local and global context self attention layers in each stage of the architecture.

blocks, which we refer to as 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 values. As we demonstrate in Sec. 5, sharing global queries allow for reducing number of parameters and FLOPs and increasing generalizability. The design of our proposed framework for global query generation 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.

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 28M, 51M and 90M parameters, referred to as tiny, small and base variants, achieve new SOTA benchmarks of **83.4%**, **83.9%** and **84.5%** Top-1 accuracy. Hence, GC ViT consistently outperforms both ConvNeXt (Liu et al., 2022) and Swin Transformer (Liu et al., 2021) models by a significant margin (see Fig. 1). Using a pre-trained GC ViT base backbone with Cascade Mask RCNN (He et al., 2017) head, our model achieves a box mAP of **52.2** for object detection and a mask mAP of **45.2** for instance segmentation on the MS COCO test-dev. In addition, using an UPerNet (Xiao et al., 2018) head, our model achieves a mIoU of **49.0** on ADE20K for semantic segmentation. 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:

- A novel hierarchical Transformer model, called GC ViT, that can be employed as a general backbone in various computer vision tasks such as classification, detection, instance segmentation down-stream tasks such as detection, instance and semantic segmentation.

- A novel and yet simple design comprising of global self-attention and token generation modules that allow for modeling long-range dependencies by capturing global contextual information and hence eliminating the need for highly sophisticated or complex operations.
- As demonstrated in Fig. 1, the proposed GC ViT achieves new SOTA benchmarks on ImageNet-1K dataset for a variety of model sizes and FLOPs, outperforming both CNN and ViT-based models by a significant margin. Using GC ViT as the backbone yields SOTA or competitive performance for object detection and semantic segmentation on MS COCO and ADE20K datasets, respectively.

2 GC ViT ARCHITECTURE

Architecture. Fig. 2 depicts the architecture of GC ViT. Similar to previous efforts (Liu et al., 2021; Yang et al., 2021b), we use a hierarchical framework to obtain feature representations at several resolutions (called stages) by decreasing the spatial dimensions while expanding the embedding dimension by factors of 2 and 2, respectively. At first, given an input image with resolution of $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, following Wang et al. (2022), we obtain overlapping patches by applying a 3×3 convolutional layer with a stride of 2 and appropriate padding. Then patches are projected it into a C-dimensional embedding space. We show the benefit of using overlapping as opposed to non-overlapping patches in Sec. 5. 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 accesses global features extracted by the Global Token Generator (GTG). The GTG is a CNN-like module that extracts features from the entire image only once at every stage. The spatial resolution is decreased by 2 while increasing the number of channels by a downsampling block after every stage. Resulting features are passed through average pooling and linear layers to create an embedding for a downstream task.

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

Downsampling. We borrow an idea of spatial feature contraction from CNN models that imposes locality bias and cross channel communication while reducing dimensions. We use 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, see Fig 3. The Fused-MBConv block in our work is similar to the one in EfficientNetV2 (Tan & Le, 2021) with modifications as in

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

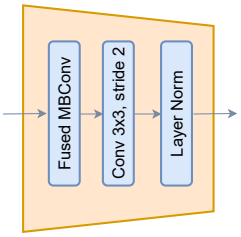


Figure 3 – Downsampling block for dimension reduction.

where SE, GELU and DW-Conv$_{3\times3}$ denote Squeeze and Excitation block (Hu et al., 2018b), 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.

Attention. Multi-head self-attention is the the core computational operator in the GC ViT architecture to extract semantic information from the image. GC ViT is composed of local and global self-attention modules illustrated in Fig 4. Similar to Swin Transformer (Liu et al., 2021), we benefit from splitting the image into windows and performing local self-attention within them, this leads to linear complexity scaling with image size. The local self-attention extracts local, short-range, information and in order to facilitate long range dependencies we propose to use a novel global self attention to allow cross-patch communication with those far beyond the local window. Global self-attention attends other regions in the image via global query token that represents image embedding extracted with CNN-like module.

2.1 GLOBAL QUERY GENERATOR

We propose to generate global query tokens that encompass information across the entire input feature maps for interaction with local key and value features. Specifically, as shown in Fig. 5, a layer $\mathbf{f}$ in

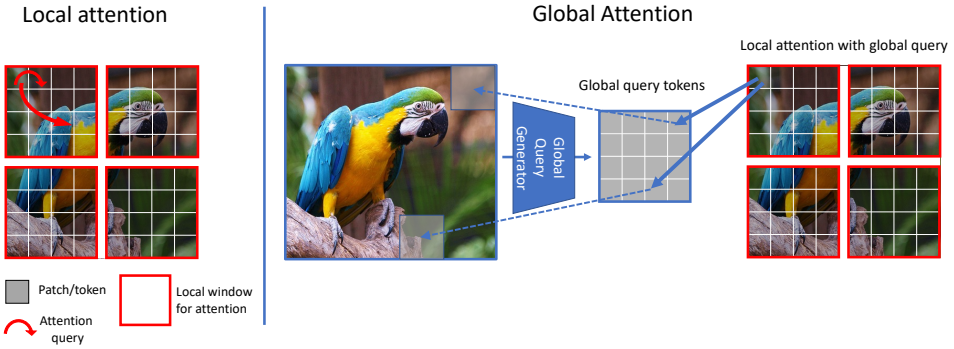


Figure 4 – Attention formulation. Local attention is computed on feature patches within local window only (left). On the other hand, the global features are extracted from the entire input features and then repeated to form global query tokens. The global query is interacted with local key and value tokens, hence allowing to capture long-range information via cross-region interaction.

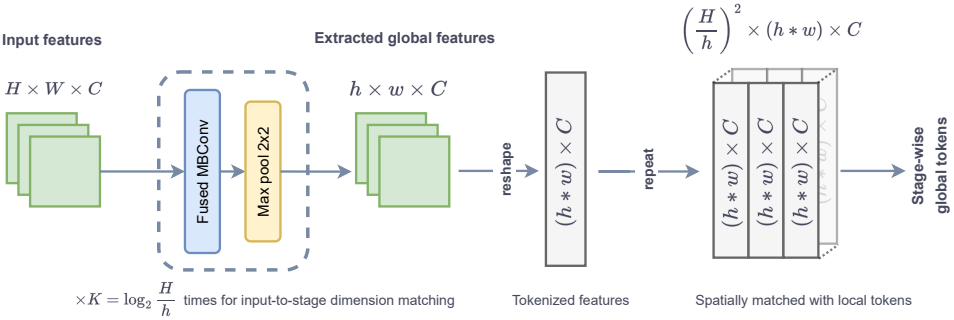


Figure 5 – Global query generator schematic diagram. It is designed to (i) transform an input feature map to current stage of dimension H, W, C being height, width, and channel respectively, (ii) extract features via repeating the Fused MBConv block, joint with down-sampling, $\log_2 \frac{H}{h}$ times for dimension matching to local window size h, output of which is (iii) reshaped and repeated to $(\frac{H}{h})^2$ number of local tokens that can now each quickly attend to global information. $\star$ denotes merged dimensions during reshaping.

the the generator consists of a Fused-MBConv block followed by a max pooling layer, similar to the one described in Sec. 2, and the final global query $\mathbf{q_{g,i}}$ at stage i ($i \in \{1, 2, 3, 4\}$) of GC ViT is computed according to

$$\begin{aligned} \mathbf{x}^i &= \text{F-MBConv}(\mathbf{x}^{i-1}), \\ \mathbf{x}^i &= \text{MaxPool}(\mathbf{x}^i). \end{aligned} \tag{2}$$

These query tokens are computed once at every stage of GC ViT model and shared across all global attention blocks, hence decreasing number of parameters and FLOPs and improving the generalizability. In addition, the Global attention layers only learn local key and value features which will be used for interaction with global query tokens.

2.2 Global Self-Attention

Fig. 4 demonstrates the main idea of our contribution. Local self-attention can only query patches within a local window, whereas with global attention attention can query image globally while still operating in the window. The only difference in implementation is that query component is pre-computed in case of global attention and the rest is the same as in local attention. In each stage, GC ViT employs alternating local and global self-attention blocks to effectively capture both local and global spatial information. The global self-attention utilizes global query tokens, obtained according to Eq. 2 and shared across all blocks, to interact with extracted local key and value features. Fig. 6 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, $\mathbf{q_g}$

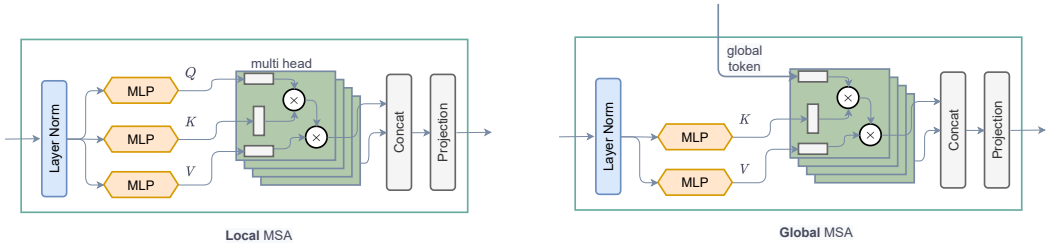


Figure 6 – Local and global attention blocks. Global attention block does not compute query vector and reuses global query computed via Global Token Generation.

is repeated along the batch dimension to compensate for the overall number of windows and batch size $B^* = B \times N$ where N is the number of local windows. $\mathbf{q_g}$ is further reshaped into multiple head. The value and key are computed within each local window using a linear layer. The global self-attention query, key and value features are computed as follows

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

$$\text{Attention}(\mathbf{q_g}, \mathbf{k}, \mathbf{v}) = \text{Softmax}(\frac{\mathbf{q_g}\mathbf{k}}{\sqrt{\mathbf{d}}} + \mathbf{b})\mathbf{v}, \quad (3)$$

where $\mathbf{d}$ is scaling factor and $\mathbf{b}$ is a learnable relative position bias term. Following previous efforts (Hu et al., 2018a; Liu et al., 2021), 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. 5, 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. A complexity analysis of the global self-attention is presented in the supplementary materials.

Algorithm. 1 Global Attention Pseudocode

```
# Input/output shape: (B*, N, C)
# B*: Batchsize*Num Windows; H: Height;
# W: Width; C: dim; q_g: Global Token;
# F: Num Attention Head; N: Num Windows;

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_global.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)
```

2.3 GC ViT Model Configurations

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

$$\mathbf{Q_g} \in \mathbb{R}^{B^* \times C \times h \times w} := [\mathbf{q_g}, ..., \mathbf{q_g}], \mathbf{q_g} \in \mathbb{R}^{B \times C \times h \times w}, \quad (4)$$

$$\mathbf{q_g} \in \mathbb{R}^{B^* \times N \times C} \xleftarrow{\text{reshape}} \mathbf{Q_g} \in \mathbb{R}^{B^* \times C \times h \times w}, \quad (5)$$

$$\mathbf{k}, \mathbf{v} = \mathbf{g}(\mathbf{x}) \in \mathbb{R}^{B^* \times N \times C}. \quad (6)$$

3 Related Work

ViT. 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. (2021b)

	Output Size (Downs. Rate)	GC ViT-XT	GC ViT-T	GC ViT-S	GC ViT-B
Stem	128×128 (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 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. GC ViT-XXT configuration is presented in the supplementary materials.

introduced a distillation-based training strategy which significantly improved the classification accuracy. 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. EdgeViT (Pan et al., 2022) introduced a lightweight ViT model, with global sparse attention and local aggregations and propagation modules for capturing short and long-range information. Please see the supplementary materials for further discussion.

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., 2018b) have dominated computer vision benchmarks with SOTA performance. Recently, ConvNeXt (Liu et al., 2022) proposed modifications to the architecture of ResNet (He et al., 2016), and achieved competitive benchmarks for classification, detection and segmentation tasks.

4 Experiments

Image classification. 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) 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.

Table 2 – Image classification benchmarks on **ImageNet-1K** dataset (Deng et al., 2009). All models are trained and evaluated on 224×224.

Method	Param (M)	FLOPs (G)	Image Size	Top-1 (%)
ResMLP-S12 (Touvron et al., 2021a)	15	3.0	224^2	76.6
PVT-v2-B1 (Wang et al., 2022)	13	2.1	224^2	78.7
GC ViT-XXT	12	2.1	224^2	**79.8**
DeiT-Small/16 (Touvron et al., 2021b)	22	4.6	3446	79.9
T2T-ViT-14 (Yuan et al., 2021)	22	5.2	2501	81.5
CPVT-Small-GAP (Chu et al., 2021b)	23	4.6	-	81.5
GC ViT-XT	20	2.6	1662	**82.0**
ResNet50 (He et al., 2016)	25	4.1	5841	76.1
CrossViT-S (Chen et al., 2021)	26	5.6	-	81.0
PVT-Small (Wang et al., 2021)	24	3.8	1951	79.8
Twins-PCPVT-S (Chu et al., 2021a)	24	3.8	1889	81.2
Swin-T (Liu et al., 2021)	29	4.5	2376	81.3
Twins-SVT-S (Chu et al., 2021a)	24	2.9	2095	81.7
PVT-v2-B2 (Wang et al., 2022)	25	4.0	-	82.0
ConvNeXt-T (Liu et al., 2022)	29	4.5	2071	82.1
Focal-T (Yang et al., 2021b)	29	4.9	-	82.2
GC ViT-T	28	4.7	1130	**83.4**
ResNet-101 (He et al., 2016)	44	7.9	3299	77.4
ResMLP-S24 (Touvron et al., 2021a)	30	6.0	2700	79.38
PVT-Medium (Wang et al., 2021)	44	6.7	1258	81.2
T2T-ViT-19 (Yuan et al., 2021)	39	8.9	1553	81.9
Twins-PCPVT-B (Chu et al., 2021a)	44	6.7	1222	82.7
Swin-S (Liu et al., 2021)	50	8.7	1431	83.0
Twins-SVT-B (Chu et al., 2021a)	56	8.6	949	83.2
ConvNeXt-S (Liu et al., 2022)	50	8.7	1276	83.1
PVT-v2-B3 (Wang et al., 2022)	45	6.9	-	83.2
Focal-S (Yang et al., 2021b)	51	9.1	-	83.5
GC ViT-S	51	8.5	756	**83.9**
ResNet-152 (He et al., 2016)	60	11.6	2307	78.3
ViT-Base/16 (Dosovitskiy et al., 2020)	86	17.6	1296	77.9
ResMLP-B24 (Touvron et al., 2021a)	116	23.0	304	81.0
PVT-Large (Wang et al., 2021)	61	9.8	-	81.7
DeiT-Base/16 (Touvron et al., 2021b)	86	17.6	1296	81.8
CrossViT-B (Chen et al., 2021)	104	21.2	-	82.2
T2T-ViT-24 (Yuan et al., 2021)	64	14.1	1104	82.3
CPVT-B (Chu et al., 2021b)	88	17.6	-	82.3
Twins-PCPVT-L (Chu et al., 2021a)	61	9.8	852	83.1
Swin-B (Liu et al., 2021)	88	15.4	985	83.3
PVT-v2-B4 (Wang et al., 2022)	62	10.1	-	83.6
Twins-SVT-L (Chu et al., 2021a)	99	15.1	655	83.7
ConvNeXt-B (Liu et al., 2022)	89	15.4	846	83.8
Focal-B (Yang et al., 2021b)	90	16.0	-	83.8
PVT-v2-B5 (Wang et al., 2022)	82	11.8	-	83.8
GC ViT-B	90	14.8	517	**84.4**
ConvNeXt-L (Liu et al., 2022)	89	15.4	846	83.8

Object detection and semantic segmentation For object detection and instance segmentation, we trained our model on MS COCO (Lin et al., 2014) with a Mask-RCNN (He et al., 2017) head, 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. (2022), 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.

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., 2022)	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., 2021b)	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., 2022)	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., 2022)	108	827	51.9	70.8	56.5	45.0	68.4	49.1
GC ViT-S	146	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., 2022)	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**

Table 3 – 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.

4.1 CLASSIFICATION

We present the ImageNet-1K classification benchmarks in Table 2 and compare against CNN, ViT and MLP-based models across different model sizes. Our model achieves new SOTA benchmarks in all categories by a large margin. Specifically, the proposed GC ViT surpasses similar-sized counterpart models by +0.5% for GC ViT-XT (82.0%) compared to CPVT-Small-GAP (Chu et al., 2021b) (81.5%), +1.2% for GC ViT-T (83.4%) over Focal-T (Yang et al., 2021b) (82.2%), +0.4% for GC ViT-S (83.9%) over Focal-S (Yang et al., 2021b) (83.5%) and +0.7% for GC ViT-B (84.5%) compared to Focal-B (Yang et al., 2021b) (83.8%), respectively. In addition, our models demonstrate similar or better performance gap compared to other established benchmarks such as ConvNeXt-T (Liu et al., 2022). Furthermore, as shown in Fig. 1, GC ViT models have better or comparable computational efficiency in terms of number FLOPs over the competing counterpart models.

Backbone	Param (M)	FLOPs (G)	mIoU
DeiT-Small/16 (Touvron et al., 2021b)	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	**49.0**

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.

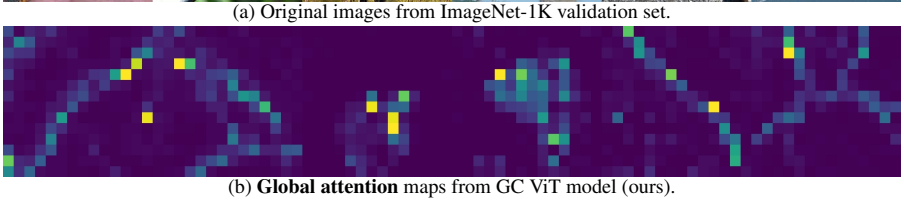
4.2 DETECTION

In Table 3, we present object detection and instance segmentation benchmarks using on MS COCO dataset. Using a Mask-RCNN head, model with pre-trained GC ViT-T (46.5/41.8) backbone outperforms counterparts with pre-trained ConvNeXt-T (Liu et al., 2022) (46.2/41.7) by +0.3 and +0.1 and Swin-T (Liu et al., 2021) (46.0/41.6) by +0.5 and +0.2 in terms of box AP and mask AP, respectively. Using a Cascade Mask-RCNN head, models with pre-trained GC ViT-T (51.1/44.4) and GC ViT-S (52.4/45.4) backbones surpass all competing counterparts. Specifically, it outperforms ConvNeXt-T (Liu et al., 2022) (50.4/43.7) by +0.7 and +0.7 and ConvNeXt-S (Liu et al., 2022) (51.9/45.0) by +0.5 and +0.4 in terms of box AP and mask AP, respectively. However, ConvNeXt-B (Liu et al., 2022) (52.7/45.6) demonstrates better performance compared to GC ViT-B (52.2/45.2) by +0.5 and +0.4 in terms of box AP and mask AP, respectively.

4.3 ADE20K SEMANTIC SEGMENTATION RESULTS

We present the semantic segmentation benchmarks on ADE20K in Table 4. Models using pre-trained GC ViT-T (46.5), GC ViT-T (48.3) backbones surpass competing models with pre-trained Twins-

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



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

Figure 7 – Visualization of : (a) input images (b) global self-attention maps (c) learned global query tokens from GC ViT-T model. Both short and long-range spatial dependencies are captured effectively.

SVT-S (Chu et al., 2021a) (46.2) and Focal-S (Yang et al., 2021b) (48.0) by +0.3 and +0.3 in terms of mIoU, respectively. Models pre-trained with GC ViT-B (49.0) and Focal-B (Yang et al., 2021b) (49.0) both achieve the same segmentation performance. GC ViT significantly outperforms Swin Transformer counterparts, hence demonstrating the effectiveness of the global self-attention.

5 Ablation

Component-wise Analysis. As shown in Table 5, replacing global self-attention blocks with their local counterparts significantly impacts the performance. Specifically, segmentation shows the most significant degradation. Sharing global context query features consistently improves the performance for all tasks, according to Table 5, and decreases the number of parameters and operations. Such improvements may imply that model generalizability improves by re-using the same tokens throughout the local and global attention blocks. Our experiments also show the benefits of employing relative position bias in both local and global self-attention blocks.

	ImageNet	COCO		ADE20k
	top-1	AP^{box}	AP^{mask}	mIoU
w/o Global-SA	82.8	46.1	41.5	44.9
w/o Global-SA query share	83.0	46.3	41.5	45.9
w/o relative position bias	83.1	46.3	41.6	46.3
w/o overlapping patches	83.1	46.4	41.6	46.4
Tiny GC ViT	**83.4**	**46.5**	**41.8**	**46.5**

Table 5 – Ablation study on the effectiveness of various components in GC ViT architecture. All experiments follow the Tiny GC ViT architecture as the baseline.

6 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. 7. 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.

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 using MS COCO and ADE20K datasets.

REFERENCES

Alaaeldin Ali, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, et al. Xcit: Cross-covariance image transformers. *Advances in neural information processing systems*, 34, 2021.

Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification, 2021.

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021a.

Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021b.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. *arXiv preprint arXiv:2107.00652*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. 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.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

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

Andrew G. Howard, Menglong Zhu, and Bo Chen. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2017.

Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3588–3597, 2018a.

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

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

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

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

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *arXiv preprint arXiv:2201.03545*, 2022.

Junting Pan, Adrian Bulat, Fuwen Tan, Xiatian Zhu, Lukasz Dudziak, Hongsheng Li, Georgios Tzimiropoulos, and Brais Martinez. Edgevits: Competing light-weight cnns on mobile devices with vision transformers. In *European Conference on Computer Vision*, pp. 294–311. Springer, 2022.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 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.

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

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

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

Hugo Touvron, Piotr Bojanowski, Mathilde Caron, Matthieu Cord, Alaaeldin El-Nouby, Edouard Grave, Gautier Izacard, Armand Joulin, Gabriel Synnaeve, Jakob Verbeek, et al. Resmlp: Feedforward networks for image classification with data-efficient training. *arXiv preprint arXiv:2105.03404*, 2021a.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357. PMLR, 2021b.

Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. *arXiv preprint arXiv:2204.01697*, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. 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.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022.

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

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22–31, October 2021.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 418–434, 2018.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

Huanrui Yang, Hongxu Yin, Pavlo Molchanov, Hai Li, and Jan Kautz. NViT: Vision transformer compression and parameter redistribution. *arXiv preprint arXiv:2110.04869*, 2021a.

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal attention for long-range interactions in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021b.

Hongxu Yin, Arash Vahdat, Jose Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. *arXiv preprint arXiv:2112.07658*, 2021.

Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token ViT: Training vision transformers from scratch on imagenet. In *ICCV*, 2021.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297, 2020.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

A APPENDIX

B GC VIT MODEL CONFIGURATIONS

We present GC ViT-XXT architecture configuration in Table S.1. For comparison, we also include architecture configurations of GC ViT-T, GC ViT-S and GC ViT-B models. GC ViT-XXT is a lightweight model which achieves a competitive Top-1 classification accuracy of **79.8%** on ImageNet-1K dataset with only 12M parameters.

	Output Size (Downs. Rate)	GC ViT-XXT	GC ViT-T	GC ViT-S	GC ViT-B
Stem	128×128 (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:64, S:2, LN	Conv, C:64, S:2, LN	Conv, C:96, S:2, LN	Conv, C:128, S:2, LN
		[LG-SA, C:64, head:2] × 2, F-MBConv, C:64	[LG-SA, C:64, head:2] × 3, F-MBConv, C:64	[LG-SA, C:96, head:3] × 3, F-MBConv, C:96	[LG-SA, C:128, head:4] × 3, F-MBConv, C:128
Stage 2	28×28 (8×)	Conv, C:64, S:2, LN	Conv, C:64, S:2, LN	Conv, C:96, S:2, LN	Conv, C:128, S:2, LN
		[LG-SA, C:64, head:4] × 2, F-MBConv, C:64	[LG-SA, C:64, head:4] × 4, F-MBConv, C:64	[LG-SA, C:96, head:6] × 4, F-MBConv, C:96	[LG-SA, C:128, head:8] × 4, F-MBConv, C:128
Stage 3	14×14 (16×)	Conv, C:64, S:2, LN	Conv, C:64, S:2, LN	Conv, C:96, S:2, LN	Conv, C:128, S:2, LN
		[LG-SA, C:64, head:8] × 6 , F-MBConv, C:64	[LG-SA, C:64, head:8] × 19, F-MBConv, C:64	[LG-SA, C:96, head:12] × 19, F-MBConv, C:96	[LG-SA, C:128, head:16] × 19, F-MBConv, C:128
Stage 4	7×7 (32×)	Conv, C:64, S:2, LN	Conv, C:64, S:2, LN	Conv, C:96, S:2, LN	Conv, C:128, S:2, LN
		[LG-SA, C:64, head:16] × 2, F-MBConv, C:64	[LG-SA, C:64, head:16] × 5, F-MBConv, C:64	[LG-SA, C:96, head:24] × 5, F-MBConv, C:96	[LG-SA, C:128, head:32] × 5, F-MBConv, C:128

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-XXT, GC ViT-T, GC ViT-S and GC ViT-B denote XTiny, Tiny, Small and Base variants, respectively.

C COMPARISON TO OTHER GLOBAL SELF-ATTENTION MODULES

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. In this section, we discuss the differences between the proposed global self-attention in GC ViT and these efforts.

EdgeViT : EdgeViT and GC ViT use completely different self-attention blocks. The EdgeViT uses a series of local aggregation (convolution), sparse attention and local propagation(depthwise convolution), whereas GC ViT only uses an interleaved pattern of local and global self-attention layers without convolution in order to compute self-attention. The proposed global sparse attention in EdgeViT and GCViT are competently different. 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, instead of interacting with each other, and hence provide an effective mechanism for capturing both short and long-range spatial dependencies.

In addition, GC ViT generates global query tokens by using a series of modified Fused MB-Conv from the entire image and without subsampling. Note that the resolution of global query tokens are the same as local query and values. However, in EdgeViT: (A) the representative tokens are obtained per local window, not the entire image, and by subsampling and reducing the feature resolution. Hence, since generated tokens have a lower resolution compared to their respective local windows, this could result in loss of spatial information and impact the effectiveness of self-attention. Unlike EdgeViT, the downsampler in GCViT also benefits from modified Fused-MBConv blocks which allows for modeling cross channel interactions and impose more locality and convolutional inductive bias.

Down-sampler	Architecture	Top-1
Conv	Conv (s=1), Maxpool	82.7
Swin	Linear	82.9
ConvNeXt	Conv (s=2)	83.1
GC ViT	Modified Fused-MBConv, Conv (s=2)	**83.4**

Table S.2 – Ablation study on the effectiveness of down-sampler in GC ViT architecture on ImageNet Top-1 accuracy.

Model	Top-1
Swin-T	81.3
Swin-T w/o Window Shifting	80.2
Reparametrization (window size, number of blocks, mlp ratio)	81.9
+ GC ViT-T Stem	82.2
+ GC ViT-T Down-sampler	82.6
+ GC ViT-T Global Self-attention	**83.4**

Table S.3 – Ablation study on the effectiveness of various components in GC ViT architecture on ImageNet Top-1 accuracy. Swin-T is used as a starting point and modified progressively to GC ViT-T which shares the same number of parameters.

BigBird : Bigbird, which is primarily introduced for NLP applications with 1D inputs, has significant differences compared to GC ViT, which is proposed for computer vision with mainly 2D inputs. Firstly, 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. In addition, 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. On the contrary, the global tokens in GC ViT are extracted by the proposed global query generator module which consists of a series of modified Fused MB-Conv blocks to extract contextual information from the entire input features. Lastly, BigBird employs a set of global tokens which attend to the entire input sequence; in this case, select global query, key and values attend to local query, key and value tensors. However, as opposed to this formulation, in GC ViT, the global query tokens attend to local key and value tokens in partitioned windows. This is due to the fact that attending to the entire input sequence, as done in BigBird, is not feasible considering the larger size of input features in computer vision.

D ABLATION

D.1 IMPORTANCE OF DOWN-SAMPLER

We studied the effectiveness of various down-sampler blocks in Table S.2. 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 0.7%. Patch merging is another variant which was introduced in Swin Transformers (Liu et al., 2021). However, it reduces the accuracy by 0.5%. Another variant is the down-sampler introduced in ConvNeXt (Liu et al., 2022) which is a 3×3 strided convolution (s=2). However, it also reduces the accuracy by 0.3% when used in our setting. Finally, our down-sampler which consists of a modified Fused-MBConv block and strided convolution and shows the best result. Importance 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.

D.2 COMPARISON TO SWIN TRANSFORMER

The most comparable architecture to GC ViT is the Swin Transformer (Liu et al., 2021) with local window splitting. As a result, we start with a Swin Transformer base design and progressively re-design the components to demonstrate their importance in improving the accuracy. Firstly, we remove window shifting and predictably observe 1.1% drop in accuracy. Changing distribution of parameters to our design improves the accuracy to 81.9%. Such reparametrization includes changing the window size, MLP ratio, number of layers to name but a few, as shown in Table 1. Adding the

Model	Accuracy-Matched Frequency	Accuracy-Threshold-0.7
GC ViT-XXT	69.3	77.2
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	74.5	81.5

Table S.4 – Classiication benchmarks of GC ViT models on ImageNetV2 dataset.

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

CNN-based stem of GC ViT to the model provides an additional 0.3% improvement in terms of ImageNet Top-1 accuracy. In addition, changing the down-sampling block to the one proposed for GC ViT improves the performance to 82.6%. The last two changes demonstrate the importance of convolutional inductive bias in our model. Finally, adding the proposed global self-attention improves the accuracy to 83.4%. Note that number of layers is not changed to the previous experiment where all layers benefit from only local self-attention. These experiments validate the effectiveness of GC ViT components and demonstrate their improvement over Swin Transformers (Liu et al., 2021) as the total number of parameters are kept the same.

E 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 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.

F 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

$$\mathcal{O}(\text{GC ViT}) = 2HW(2C^2 + hwC), \tag{7}$$

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.

G IMAGENETV2 BENCHMARKS

In Table S.4, 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.

H EFFECT OF GLOBAL CONTEXT MODULE

In order to demonstrate the effectiveness of Global Context (GC) 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.5, addition of GC module improves the ImageNet Top-1 accuracy by $+0.9\%$ and $+0.7\%$ for Swin Transformers Tiny and Small variants respectively.