ViT-LSLA: Vision Transformer with Light Self-Limited-Attention

Zhenzhe Hechen, Wei Huang, Yixin Zhao[*]

College of Computer and Information Science, Southwest University, Chongqing, China
zyx_cq@swu.edu.cn

Abstract

Transformers have demonstrated a competitive performance across a wide range of vision tasks, while it is very expensive to compute the global self-attention. Many methods limit the range of attention within a local window to reduce computation complexity. However, their approaches cannot save the number of parameters; meanwhile, the self-attention and inner position bias (inside the softmax function) cause each query to focus on similar and close patches. Consequently, this paper presents a light self-limited-attention (LSLA) consisting of a light self-attention mechanism (LSA) to save the computation cost and the number of parameters, and a self-limited-attention mechanism (SLA) to improve the performance. Firstly, the LSA replaces the K (Key) and V (Value) of self-attention with the X(origin input). Applying it in vision Transformers which have encoder architecture and self-attention mechanism, can simplify the computation. Secondly, the SLA has a positional information module and a limited-attention module. The former contains a dynamic scale and an inner position bias to adjust the distribution of the self-attention scores and enhance the positional information. The latter uses an outer position bias after the softmax function to limit some large values of attention weights. Finally, a hierarchical **Vi**sion **T**ransformer with **L**ight **s**elf-**L**imited-**a**ttention (ViT-LSLA) is presented. The experiments show that ViT-LSLA achieves 71.6% top-1 accuracy on IP102 (2.4% absolute improvement of Swin-T); 87.2% top-1 accuracy on Mini-ImageNet (3.7% absolute improvement of Swin-T). Furthermore, it greatly reduces FLOPs (3.5GFLOPs vs. 4.5GFLOPs of Swin-T) and parameters (18.9M vs. 27.6M of Swin-T).

1 Introduction

The advent of Transformer (Vaswani et al. 2017) has made a profound effect on natural language processing (NLP) (Devlin et al. 2018; Radford et al. 2018; Howard and Ruder 2018). In addition, the Vision Transformer (ViT) (Dosovitskiy et al. 2021) has shown a promising performance compared with its CNN counterparts. Inspired by the ViT, several visual Transformers were proposed (Touvron et al. 2021; Wu et al. 2021a; Yuan et al. 2021). However, it is unsuitable for various vision tasks to adopt the primal full self-attention, which results in expensive computation cost (the computational complexity of self-attention is quadratic to image size).

To address this issue, on the one hand, a typical way is confining the range of global self-attention to a local region. Swin Transformer (Liu et al. 2021) limited the computation of self-attention to local windows and constructed cross-window connections between two successive blocks. CSwin (Dong et al. 2021) and Pale Transformer (Wu et al. 2021b) designed cross-shaped windows and Pale-shaped windows respectively. Shuffle Transformer (Huang et al. 2021) proposed shuffled windows. Axial-DeepLab (Wang et al. 2020) applied two axial-attention layers consecutively for the height-axis and width-axis, improving both global connection and efficient computation. On the other hand, some recent works were devoted to the linearization of self-attention (Choromanski et al. 2020; Bello 2021; Shen et al. 2021). The CoaT(Xu et al. 2021) especially proposed a factorized attention mechanism whose computation complexity is quadratic w.r.t. the channel while linear w.r.t. the image size. These methods reduce the computation cost to some degree; nevertheless, they cannot save the number of parameters.

Furthermore, many previous works adopted a fixed scale value to deal with the large values of the dot product (Vaswani et al. 2017; Liu et al. 2021; Dosovitskiy et al. 2021; Dong et al. 2021; Xu et al. 2021; Lee et al. 2022). The fixed scale can prevent the minimal gradients in the softmax function and adjust the variance of attention scores to 1. Because it cannot help self-attention grasp the positional information, in some works (Liu et al. 2021; Bao et al. 2020; Hu et al. 2018), an inner relative position bias is used to enhance the ability to capture positional information. However, the values of attention scores are the similarities between each vector to the others, which means that the similar vectors have big attention scores. After adding the inner positional bias to the attention scores, the computation of self-attention can be regarded as a local information enhancement. This makes the queries prone to focusing on the similar and close patches including itself rather than really relevant ones.

This paper proposes a Light Self-Limited-Attention (LSLA) consisting of a light self-attention mechanism (LSA) and a self-limited-attention mechanism (SLA).

Unlike the encoder-decoder architecture and cross-attention mechanism in the natural language processing

[*] Yixin Zhao is the corresponding author.

(NLP) tasks (e.g. machine translation), the vision Transformers have encoder-only architecture and self-attention mechanism in classification tasks. Moreover, there are two kinds of language input in machine translation, but image classification only needs to deal with the image input. Therefore, the LSA changes self-attention calculation from Q, K, V to Q, X, X, which can significantly reduce the parameters and computation cost of the self-attention layer. It can be beneficial to stack more self-attention blocks, which helps Transformers go deeper and get better performance.

The self-limited-attention mechanism (SLA) proceeds as follows. Firstly, the dynamic scale cooperating with the inner position bias can explicitly indicate the positional information (see blue patches in Figure 3(a)). Based on it, the outer position bias can powerfully limit some large values of attention weights (see red patches in Figure 3(b)), which is helpful to pay attention to the meaningful patches instead of the similar but unimportant ones. These processes can be regarded as a local information integration, which is conducive to retaining the diversity of information for each query patch.

Through the LSLA, this paper designs a hierarchical **Vi**sion **T**ransformer (as illustrated in Figure 1(a)) with **L**ight self-**L**imited-attention (ViT-LSLA), which achieves a better performance than previous approaches and reduces parameters and computation cost significantly. The ViT-LSLA (18.9M, 3.5GFLOPs) achieves 71.6% Top-1 classification accuracy (2.4% and 1.2% absolute improvement of Swin-T (27.6M, 4.5GFLOPs) and MPViT-S (22.6M, 4.8GFLOPs)) on IP102, and 87.2% Top-1 classification accuracy on Mini-ImageNet (3.7% and 1.1% absolute improvement of Swin-T and MPViT-S). The models in this paper were trained on two Tesla P100 GPUs.

The contributions of this work are summarized as follows:

- A light self-attention mechanism (LSA) is provided as a plug-and-play module. The current Transformers can save the number of parameters and FLOPs conveniently without losing the accuracy by applying the LSA in self-attention blocks.
- A self-limited-attention mechanism (SLA) is introduced. Based on positional information, an outer position bias is adopted to powerfully limit the large attention weights. Thus, Transformers can capture truly meaningful information rather than which just has high similarity.
- Establishing a simple Transformer model with the above components can not only remarkably reduce the computation cost and the number of parameters, but also significantly improve the performance.

2 Related Work

In NLP tasks, there are various efficient Transformer models (Beltagy, Peters, and Cohan 2020; Choromanski et al. 2020; Katharopoulos et al. 2020; Kitaev, Kaiser, and Levskaya 2020; Roy et al. 2021). In the CV field, lots of visual Transformers (Dosovitskiy et al. 2021; Touvron et al. 2021; Wu et al. 2021a; Yuan et al. 2021) adopted the original full self-attention. However, because the resolution of images is very high, it is essential to design efficient self-attention mechanisms for vision tasks. To avoid the quadratic computation complexity caused by self-attention, many approaches (Wang et al. 2021; Zhu et al. 2020; Liu et al. 2021; Huang et al. 2021) are devoted to improving efficiency while maintaining performance.

2.1 Local Self-Attention Mechanism

Long-range dependency is important for NLP tasks, because one word may be associated with another far word. However, two patches, which are far apart in an image, may not be relevant (e.g. sky and ground). Namuk Park et al. (Park and Kim 2022) introduce that an appropriate inductive bias is more profitable to improve the performance of MSA than long-range dependency. As an extreme example, a local 3×3 receptive field outperforms the global one, because it reduces unnecessary degrees of freedom. Many works (Liu et al. 2021; Huang et al. 2021; Fang et al. 2021; Wang et al. 2020; Dong et al. 2021; Yang et al. 2021) focus on strengthening the local feature extraction of Vision Transformers and reducing the quadratic complexity of self-attention.

2.2 Efficient Self-attentions Mechanism

Several works improve efficiency by constraining the sequence length of key and value. Deformable attention (Zhu et al. 2020) sampled a part of keys from the full set by a linear layer, hoping to get similar global attention. However, such a downsampling operation may result in information confusion. CoaT (Xu et al. 2021) proposed an efficient factorized self-attention, which has the linear computation complexity w.r.t the number of image size and quadratic complexity w.r.t the channels. This modification can save quite a few computation cost compared with the original self-attention (whose computation complexity is quadratic w.r.t the number of image size). An efficient self-attention (Bello 2021) introduced a global context modeling, which constructed the relative position embeddings in local context modeling by convolutions. Different from the above works, this paper presents a simple light self-attention mechanism, which discards the Key and Value in MSA.

2.3 Positional Encoding

Because self-attention cannot grasp the positional information, it is essential to apply positional encoding for Transformer models. A direct way is to inject absolute positional encoding (APE) into the input embedding (Vaswani et al. 2017). Relative positional encoding (RPE) (Liu et al. 2021; Bao et al. 2020; Hu et al. 2018) and convolutional positional encoding (CPE) (Xu et al. 2021; Chu et al. 2021) are also widely used in Transformers. All these methods are applied inside the softmax function to reevaluate the attention scores. However, attention scores indicate the values of the dot product, and two similar vectors have a big dot product. It makes attention focus on the near and similar areas. In this paper, a different way of positional encoding is presented, which can be deployed outside the softmax function to reevaluate the attention weights. It can limit the neighbors' big attention weights and maintain meaningful ones.

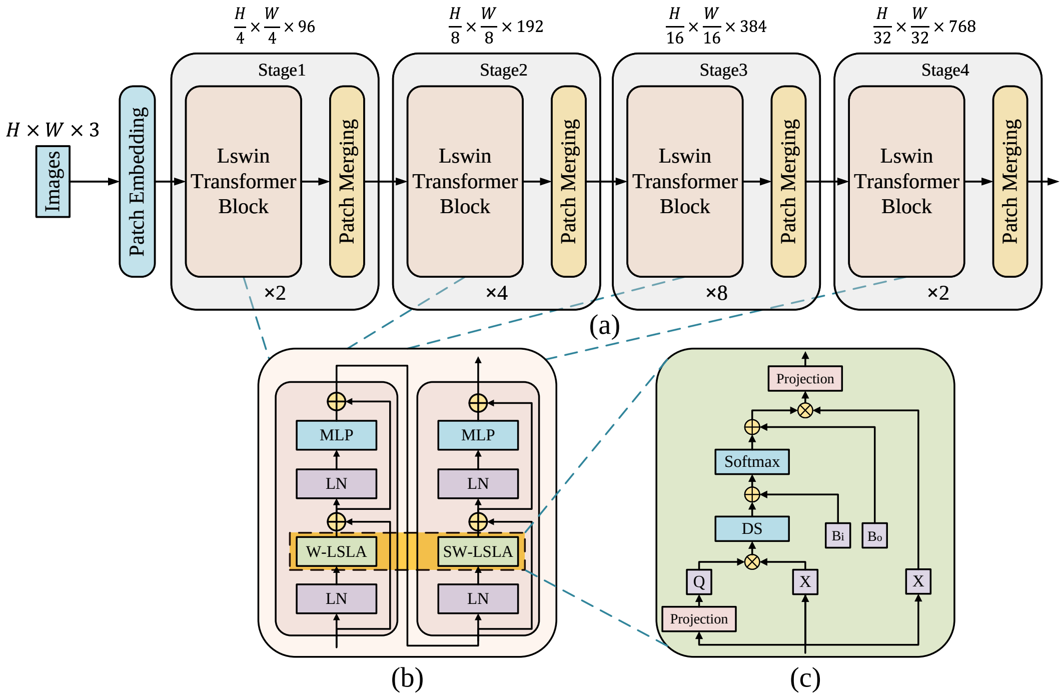


Figure 1: (a) is the model architecture of the ViT-LSLA Transformer; (b) shows two successive ViT-LSLA Transformer blocks; (c) is the description of the LSLA.

3 Method

3.1 Architecture

The overview architecture of the ViT-LSLA is illustrated in Figure 1(a). Following (Lee et al. 2022), for an input image with the size of $H \times W \times 3$, a stem block consisting of two 3×3 convolutional layers is adopted to obtain a feature with the size of $H/4 \times W/4 \times 96$. The whole model is composed of four stages referring to (Liu et al. 2021). For producing a hierarchical representation, the patch merging layer following (Dong et al. 2021) is used to reduce the number of tokens and expand the channel dimension between two adjacent stages.

The rough architecture of ViT-LSLA blocks (see Figure 1(b)) follows Swin Transformer blocks. The difference between the blocks of Swin and ViT-LSLA is that the latter replaces the original self-attention mechanism with the light self-limited-attention mechanism (LSLA).

As shown in Figure 1(c), there are the components of our LSLA module; the primary contributions of this paper are the light self-attention mechanism (QXX), and the self-limited-attention mechanism consisting of a dynamic scale (DS) and an outer position bias (B_o). Each of these components is severally elaborated in the following subsections.

3.2 Light Self-Attention Mechanism

Before introducing the light self-attention (LSA), the FC (fully-connected layers) of the MLP (multilayer perceptron) is firstly considered. It is essential for MLP to apply an activation function after each layer. Otherwise, the MLP will collapse into a linear model (Zhang et al. 2021):

$$H = XW_1 + b_1, \tag{1}$$

$$O = HW_2 + b_2, \tag{2}$$

where $X \in \mathbb{R}^{n \times n}$ is the input; H and O are the hidden variables and the output, and there is no nonlinear activation function between both of them; $W_1 \in \mathbb{R}^{n \times m}$, $W_2 \in \mathbb{R}^{m \times d}$ are the weight matrices, and $b_1 \in \mathbb{R}^{1 \times m}$, $b_2 \in \mathbb{R}^{1 \times d}$ are the biases. Substituting Eq. (1) into Eq. (2), the following equation can be derived:

$$\begin{aligned} O &= (XW_1 + b_1)W_2 + b_2 \\ &= XW_1W_2 + b_1W_2 + b_2. \end{aligned} \tag{3}$$

Adding the hidden layer requires the model to track and update additional sets of parameters, but it is not beneficial to improving performance. Assume that $W = W_1W_2$ and $b = b_1W_2 + b_2$. The solution is deduced as:

$$O = XW + b, \tag{4}$$

where $W \in \mathbb{R}^{n \times d}$ is the weight matrix and $b \in \mathbb{R}^{1 \times d}$ is the bias. It shows that the hidden layer and the output layer can be collapsed into a single layer.

Inspired by this, the original self-attention mechanism can also be simplified in the same way.

QX or QK: This part firstly expounds on how **K** (Keys) can be easily replaced by **X**. For the sake of simplicity, the formula of original MSA is defined as below:

$$\text{Attention}(Q, K, V) = \text{SoftMax}(QK^T)V. \tag{5}$$

In Eq. (5), where Q, K, and $V \in \mathbb{R}^{M^2 \times d}$ respectively are the matrices of Query, Key, and Value; M^2 is the number of patches in a window (assume that the M is 7 in this paper) and d is the channel dimension. Moreover, the Q, K, and V are derived from X via an FC, whose bias is omitted for simplicity:

$$Q = XW_q,\ K = XW_k,\ V = XW_v, \tag{6}$$

where $X \in \mathbb{R}^{M^2 \times d}$ is the original input and $W_q, W_k, W_v \in \mathbb{R}^{d \times d}$ are the weight matrices of FC. On the basis of Eq. (5) and (6), the QK^T can be mathematically formulated as below:

$$QK^T = (QW_k^T)X^T = X(W_qW_k^T)X^T. \tag{7}$$

Let $W_qW_k^T$ be $W \in \mathbb{R}^{d \times d}$, the following expression can be derived:

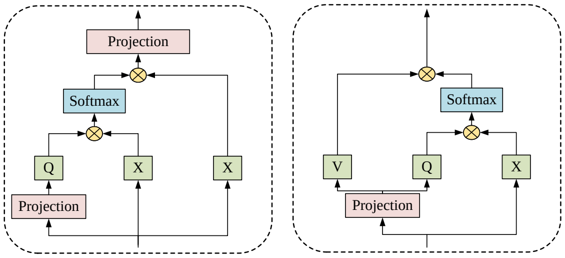
$$QK^T = XWX^T. \tag{8}$$

Let XW be $\overline{Q} \in \mathbb{R}^{M^2 \times d}$, Eq. (7) reduces to:

$$QK^T = \overline{Q}X^T. \tag{9}$$

Note that the $\overline{Q}$ can also be derived from X via an FC; both Q and $\overline{Q}$ can be optimized in the training process. To sum up, the $\overline{Q}X^T$ is equivalent to QK^T; let $\overline{Q}$ be Q, the Eq. (5) can be rewritten as:

$$\text{Attention}(Q, X, V) = \text{SoftMax}(QX^T)V. \tag{10}$$



(a) QXX with projection (Ours)

(b) QXV without projection

Figure 2: The rough architecture of QXX and QXV.

QXV or QXX: Similarly, the QXV can also reduce to QXX in the same way. Eq. (10) is predigested as:

$$\text{Attention}(Q, X, V) = AV. \tag{11}$$

From Eq. (6), the following equation can be derived:

$$AV = AXW_v. \tag{12}$$

It is reasonable for Transformers, which replace V with X, to get a weaker performance. However, there is a linear projection applied at the end of the MSA module (see Figure 2(a)), which is the point why can V be replaced with X.

Let the weight matrix of this linear projection be $W_o \in \mathbb{R}^{d \times d}$. The following equation can be derived:

$$AVW_o = AXW_vW_o. \tag{13}$$

Let W_vW_o be $W \in \mathbb{R}^{d \times d}$, it can be expressed as:

$$AXW_vW_o = AXW. \tag{14}$$

Therefore, it is rational to conclude that the linear projection of V (W_v) is equivalent to the last one (W_o); namely, either of them can be discarded as shown in Figures 2(a) and 2(b). In this paper, the first method is selected; therefore, the light self-attention (LSA) mechanism is defined as:

$$\text{Attention}(Q, X, X) = \text{SoftMax}(QX^T)X. \tag{15}$$

3.3 Self-Limited-Attention Mechanism

The Self-Limited-Attention (SLA) has a positional information module and a limited-attention module. The former contains a dynamic scale and an inner position bias to enhance the positional information, which can mark the patches that need to be limited. Based on it, the latter adopts an outer position bias after the softmax function to limit the values of attention weights marked before.

Positional Information Module: Since the Transformer models cannot capture the positional information, it is necessary to apply some relative or absolute position for the image patches. In (Liu et al. 2021; Bao et al. 2020; Hu et al. 2018) a relative position bias is introduced to each head in computing similarity. This paper follows (Liu et al. 2021) by maintaining an inner position bias (inside the softmax function). The LSA with fixed scale (FS) is formulated as:

$$\text{Attention}(Q, X, X) = \text{SoftMax}(QX^T/\sqrt{d} + B_i)X, \tag{16}$$

where $\sqrt{d}$ is the query dimension; $B_i \in \mathbb{R}^{M^2 \times M^2}$ is the inner position bias.

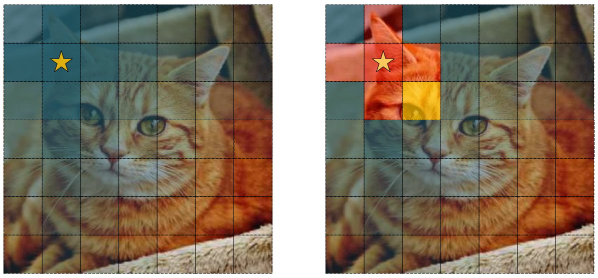
To enhance the ability to capture positional information, a dynamic scale (DS) is designed to replace the fixed scale. It concerts with the inner position bias, containing a group of learnable parameters. The LSA with DS can be formulated as follows:

$$\text{Attention}(Q, X, X) = \text{SoftMax}(QX^T \times DS + B_i)X, \tag{17}$$

where $DS \in \mathbb{R}^{M^2 \times M^2}$, it denotes that the patches near the query patch (golden star) have larger values of dynamic scale and inner position bias, as shown in Figures 3(a) and 4(a). It is helpful for the query patches to capture the positional information of each image patch. This can be regarded as a local information enhancement.

Limited-Attention Module: Depending on the positional information module, LSA can enhance the local information. Nevertheless, the values of dot products are the similarities between each vector to the others, which means that the similar vectors have big attention weights. However, such a local information enhancement forces self-attention to focus on the close and similar patches, including itself.

To retain the diversity of information for each query patch, an additional outer position bias $B_o \in \mathbb{R}^{M^2 \times M^2}$ (outside the softmax function) is used, whose construction is as same as the inner one. The outer position bias cooperating with the dynamic scale and inner position bias can overwhelmingly restrict the big attention weights close to the



(a) The dynamic scale and inner position bias at the eighth query patch

(b) The neighbors' attention weights of the query are limited by the outer position bias

Figure 3: In(a), it represents the DS and inner position bias enhancing the local positional information; the patches close to the query have stronger DS and inner position bias. (b) indicates the outer position bias can powerfully limit large neighbors' attention weights of queries and retain the truly meaningful ones.

query patch. As shown in Figures 3(b) and 4(b), the eighth patch is the query. The outer position bias can severely limit its neighbors' attention weights, especially its own. But the attention weights of the other patches are retained. Finally, the light self-limited-attention (LSLA) is formulated as below:

$$\text{Attention}(Q, X, X) = (\text{SoftMax}(QX^T \times DS + B_i) + B_o)X. \quad (18)$$

It is easy to understand the rationality of outer position bias. In Figure 3(b), the query patch (golden star) locates at the cat's ear and the notable patch (golden patch) locates at the cat's eye. This process can be regarded as a local information integration, which is conducive to self-attention to capture the notable information instead of a close and similar one. The code of the LSLA is shown in Algorithm 1.

4 Experiments

To show the effectiveness of ViT-LSLA, several experiments are conducted on different image classification datasets, and the training settings are introduced. Furthermore, detailed ablation studies are performed to analyze every module of ViT-LSLA.

4.1 Image Classification

Datasets: This paper evaluates ViT-LSLA on three benchmark datasets, IP102 (Wu et al. 2019), Food101 (Bossard, Guillaumin, and Van Gool 2014), and Mini-ImageNet (Ravi and Larochelle 2017).

Experiments Setup: Both training and evaluation are conducted with the input size of 224 × 224 on all datasets above. All models train 300 epochs on the three datasets; the training setting follows (Liu et al. 2021). The AdamW optimizer with a weight decay of 0.05 is used. The default batch size and initial learning rate are 256 and 2.5e-4; except that 192 and 1.875e-4 of MPViT-S (due to the limit of graphic memory). All experiments are performed with two Tesla P100 GPUs.

Algorithm 1: Light Self-Limited-Attention

```
% Input: The feature map of image, shape [N, D]: x
% Output: The result of light self-attention on input, shape [N, D]: y
% Signal meanings:
 % N: the number of input tokens
 % H: the number of attention heads
 % HD: the number of head-dimension
 % D: the dimension of token vector
 % @: matrix multiplication
def init():
 fc_q = Linear(D, D)
 inner_bias = Parameter(size=(H, N, N))
 outer_bias = Parameter(size=(H, N, N))
 dynamic_scale = Parameter(size=(N, N))
 proj = Linear(D, D)

def forward(x):
% First step: compute attention scores
 % linearly map the tokens to query
 q = fc_q(x) % shape: [N, D]
 % split into multi-head
 q = q.reshape(H, N, HD)
 x = x.reshape(H, N, HD)
 % compute the attention scores
 % shape: [H, N, N]
 attn = (q @ x.transpose())
% Second step: capture positional information
 attn = attn * dynamic_scale
 attn = attn + inner_bias
% Third step: compute attention weights
 attn = softmax(attn, dim=-1)
 % Limit neighbours' attention weights by outer position bias
 attn = attn + outer_bias
 % Final step: concatenate all heads
 y = (attn @ x).concat() % shape: [N, D]
 y = proj(y)
 return y
```

IP102: IP102 is a large-scale dataset of insect pets, which consists of more than 75,000 images on 102 insect pets. For classification, it has 45,095 images for training and 22619 testing images. As illustrated in Table 1, this paper compares some recent SOTA Vision Transformers with our ViT-LSLA. The ViT-LSLA outperforms all of the other SOTA counterparts. The ViT-LSLA is +2.4% higher than the most relevant Swin-T Transformer in top-1 accuracy; +2.6%, 3.4%, and 1.2% better than CSwin-T, CoarT-Lite-Small, MPViT-S respectively. Furthermore, the ViT-LSLA can save the number of parameters and computation cost significantly; 18.9M (-8.7M) of ViT-LSLA vs. 27.6M of Swin-T, which has almost one-third fewer parameters. Meanwhile, among these models, ViT-LSLA has the fewest FLOPs, which is almost three-quarters of Swin-T. To summarize, the ViT-LSLA achieves a better efficiency-quality trade-off compared with other SOTA Vision Transformers.

IP102 trained models				
Model	Input	Params	FLOPs	Top-1 acc
Swin-T(2021)	224^2	27.6M	4.5G	69.2
CSwin-T(2021)	224^2	21.9M	4.3G	69.0
CoaT-Lite-S(2021)	224^2	19.4M	4.0G	68.2
MPViT-S(2022)	224^2	22.6M	4.8G	70.4
ViT-LSLA (Ours)	224^2	18.9M	3.5G	**71.6**

Table 1: The performance comparison of different backbones on IP102 classification.

Food101: This dataset contains 101,000 real-world food images for 101 of the most popular dishes. There are 750 training images and 250 test images for each class. Table 2 demonstrates that the performance of ViT-LSLA is still the most competitive compared to other Transformer counterparts. Because of the marginal utility, the accuracy of ViT-LSLA is not much better than others, but it achieves the most promising efficiency-quality trade-off.

Food101 trained models				
Model	Input	Params	FLOPs	Top-1 acc
Swin-T(2021)	224^2	27.6M	4.5G	88.3
CSwin-T(2021)	224^2	21.9M	4.3G	89.1
CoaT-Lite-S(2021)	224^2	19.4M	4.0G	85.1
MPViT-S(2022)	224^2	22.6M	4.8G	88.3
ViT-LSLA (Ours)	224^2	18.9M	3.5G	**89.4**

Table 2: The performance comparison of different backbones on Food101 classification.

Mini-ImageNet: This dataset with 100 classes used in few-shot learning research is a subset of ImageNet, which has 600 examples for each category. In this paper, a preprocessed version of miniImageNet in which images are not resized to any particular size is used in supervised learning research; there are 480 images for training and 120 images for tests in each class. As shown in Table 3, the ViT-LSLA can not only obtain competitive performance on fine-grained classification datasets of IP102 and Food101 but also on generic one of Mini-ImageNet.

Mini-ImageNet trained models				
Model	Input	Params	FLOPs	Top-1 acc
Swin-T(2021)	224^2	27.6M	4.5G	83.5
CSwin-T(2021)	224^2	21.9M	4.3G	84.4
CoaT-Lite-S(2021)	224^2	19.4M	4.0G	83.7
MPViT-S(2022)	224^2	22.6M	4.8G	86.1
ViT-LSLA (Ours)	224^2	18.9M	3.5G	**87.2**

Table 3: The performance comparison of different backbones on Mini-ImageNet classification.

4.2 Ablation Study

In this subsection, the ablation studies are designed to prove the effectiveness of each component in the ViT-LSLA using the IP102 and Mini-ImageNet classification datasets. They are respectively fine-grained and generic classification datasets. It shows that the important elements of LSLA can deal with the different kinds of datasets.

Light Self-Attention Mechanism: In subsection 3.2, the effect of LSA has been demonstrated by mathematical techniques. As shown in Table 4, there are several comparisons between LSA and original MSA on the ViT-LSLA.

QXX or QKV: Comparing the QXX with QKV, the LSA can significantly save the number of parameters and computation cost by large margins of over 20% (18.9M vs. 24.0M on Parameters and 3.5GFLOPs vs. 4.4GFLOPs on computation); meanwhile, the accuracy is almost unchanged. The QXX has the same accuracy as QKV on IP102 and a -0.5% decrease on Mini-ImageNet. Through this control experiment, the result shows the LSA achieves a very promising efficiency-quality trade-off.

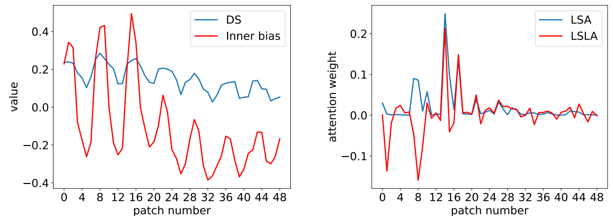
Comparison of different self-attention							
Datasets	Q X X	Q X V	Q K V	NP	Params (M)	FLOPs (G)	Top-1 acc
IP102	✓			✓	16.4	3.1	69.8
	✓				18.9	3.5	**71.6**
		✓			18.9	3.5	71.3
			✓		24.0	4.4	71.6
Mini-Image Net	✓			✓	16.4	3.1	84.8
	✓				18.9	3.5	**87.2**
		✓			18.9	3.5	87.2
			✓		24.0	4.4	87.7

Table 4: This Table shows the performance of QXX, QXV, and QKV on IP102 and Mini-ImageNet classification. NP means the last projection (as shown in Figure 2(a)) of self-attention is discarded.

QXX or QXV: As demonstrated in Table 4, the ViT-LSLA with QXX has the same number of parameters and computation cost as that with QXV; meanwhile, their performances are almost identical. But the QXX without the last projection will lose accuracy substantially (-1.8% on IP102 and -2.4% on Mini-ImageNet). Therefore, the result proves that the QXX with the last projection is equivalent to the QXV (see Figure 2).

Since the better performance of the parallel process in QXV, it can save memory and training time over QXX. But on IP102, the QXV has a slightly inferior accuracy compared to the QXX. Thus, in this paper, the QXX is selected as LSA; research on QXV self-attention mechanism is left for future work.

Self-Limited-Attention Mechanism: The SLA consists of a positional information module and a limited-attention module. The former contains a dynamic scale and an inner position bias to enhance the positional information, which can mark the patches that need to be limited. The latter adopts an outer position bias after the softmax function to



(a) Please see this figure combining with Figure 3(a)

(b) Please see this figure combining with Figure 3(b)

Figure 4: The visualization of the Figure 3. The data of Figures (a) and (b) comes from a local window (window size is 7) of ViT-LSLA trained on IP102.

limit the values of attention weights marked before.

Positional Information Module: As illustrated in Figure 4(a), the blue line and red line mean that patches close to the query patch have larger values of scale and inner position bias, which infers that the DS and inner position bias contains positional information.

Limited-Attention Module: The blue line and red line are severally the attention weights of LSA and LSLA (LSA has no outer position bias compared with LSLA). As shown in Figure 4(b), the outer position bias can powerfully limit neighbors' large attention weights and retain the truly meaningful ones.

Ablation study of ViT-LSLA on IP102 and Mini-ImageNet

Dataset	FS	DS	inner bias	outer bias	Top-1 acc
IP102	✓		✓		70.6
	✓			✓	70.9
	✓		✓	✓	71.5
		✓	✓		70.5
		✓		✓	71.6
		✓	✓	✓	**71.6**
Mini-ImageNet	✓		✓		86.1
	✓			✓	86.9
	✓		✓	✓	87.0
		✓	✓		86.2
		✓		✓	87.1
		✓	✓	✓	**87.2**

Table 5: The FS and DS are severally the fixed scale and dynamic scale; the inner bias and outer bias are respectively inner and outer relative position bias.

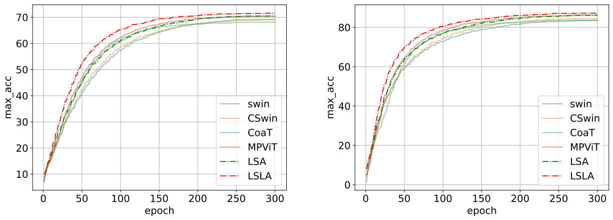
As shown in Table 5, there are several control experiments on IP102 and Mini-ImageNet datasets. It can be observed that the outer position bias can significantly improve the performance of ViT-LSLA.

In the past works, the common sense is that injecting positional information into self-attention is important. However, the self-attention makes queries naturally prone to focus on similar patches; after adding to the inner position bias, the queries pay more attention to the near and similar areas. This process is a local information enhancement but may cause a lack of information diversity.

First of all, even though there is not any positional information in the LSA, relying only on the outer position bias can obtain better performance. (see the comparison of inner bias and outer bias with a fixed scale in Table 5)

Secondly, adopting inner position bias or dynamic scale to enhance the positional information can mark the patches that need to be limited. (see Figures 3(b), 4(b)). Based on it, the outer position bias can precisely limit the values of attention weights marked before, which is conducive to retaining the diversity of information for each query patch.

Lastly, the LSLA can increase performance and the speed of convergence. In Figure 5, the convergent speed of LSLA is faster than other counterparts. Note that simply dropping the outer position bias of LSLA, the LSA can result in substantially inferior performance and convergence speed.



(a) The accuracy of different models on IP102

(b) The accuracy of different models on Mini-ImageNet

Figure 5: The LSLA is our ViT-LSLA; the LSA is the ViT-LSLA discarding the outer position bias.

5 Conclusion

This paper presents a hierarchical vision Transformer with the light self-limited-attention (ViT-LSLA). The LSLA consists of a light self-attention mechanism (LSA) and a self-limited-attention mechanism (SLA).

Firstly, the LSA changes self-attention calculation from Q, K, V to Q, X, X, which can be conveniently applied in current vision Transformers. The LSA can reduce the parameters and computation cost significantly while maintaining performance.

Secondly, this paper introduces the SLA containing a positional information module and a limited-attention module. The former adopts a dynamic scale (DS) and an inner relative position bias to enhance the positional information. The latter uses an outer relative position bias depending on the former to powerfully restrict the large values of attention weight near the query patches. It is beneficial for queries to focus on important and meaningful patches rather than that close and similar ones, which can retain the diversity of information in each query patch.

Finally, the ViT-LSLA demonstrates a competitive performance on several classification datasets and obtains the best efficiency-quality trade-off compared to other counterparts. Therefore, it is hoped that the LSLA will inspire more competitive and novel attention mechanisms.

References

Bao, H.; Dong, L.; Wei, F.; Wang, W.; Yang, N.; Liu, X.; Wang, Y.; Gao, J.; Piao, S.; Zhou, M.; et al. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *International Conference on Machine Learning*, 642–652. PMLR.

Bello, I. 2021. Lambdanetworks: Modeling long-range interactions without attention. *arXiv preprint arXiv:2102.08602*.

Beltagy, I.; Peters, M. E.; and Cohan, A. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Bossard, L.; Guillaumin, M.; and Van Gool, L. 2014. Food-101 – Mining Discriminative Components with Random Forests. In *European Conference on Computer Vision*.

Choromanski, K.; Likhosherstov, V.; Dohan, D.; Song, X.; Gane, A.; Sarlos, T.; Hawkins, P.; Davis, J.; Mohiuddin, A.; Kaiser, L.; et al. 2020. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*.

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

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, X.; Bao, J.; Chen, D.; Zhang, W.; Yu, N.; Yuan, L.; Chen, D.; and Guo, B. 2021. Cswin transformer: A general vision transformer backbone with cross-shaped windows. *arXiv preprint arXiv:2107.00652*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

Fang, J.; Xie, L.; Wang, X.; Zhang, X.; Liu, W.; and Tian, Q. 2021. Msg-transformer: Exchanging local spatial information by manipulating messenger tokens. *arXiv preprint arXiv:2105.15168*.

Howard, J.; and Ruder, S. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Hu, H.; Gu, J.; Zhang, Z.; Dai, J.; and Wei, Y. 2018. Relation networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3588–3597.

Huang, Z.; Ben, Y.; Luo, G.; Cheng, P.; Yu, G.; and Fu, B. 2021. Shuffle transformer: Rethinking spatial shuffle for vision transformer. *arXiv preprint arXiv:2106.03650*.

Katharopoulos, A.; Vyas, A.; Pappas, N.; and Fleuret, F. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, 5156–5165. PMLR.

Kitaev, N.; Kaiser, Ł.; and Levskaya, A. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Lee, Y.; Kim, J.; Willette, J.; and Hwang, S. J. 2022. MPViT: Multi-Path Vision Transformer for Dense Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

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

Park, N.; and Kim, S. 2022. How Do Vision Transformers Work? In *International Conference on Learning Representations*.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding with unsupervised learning. *Technical report, OpenAI*.

Ravi, S.; and Larochelle, H. 2017. Optimization as a Model for Few-Shot Learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Roy, A.; Saffar, M.; Vaswani, A.; and Grangier, D. 2021. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9: 53–68.

Shen, Z.; Zhang, M.; Zhao, H.; Yi, S.; and Li, H. 2021. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3531–3539.

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

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, H.; Zhu, Y.; Green, B.; Adam, H.; Yuille, A.; and Chen, L.-C. 2020. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, 108–126. Springer.

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

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

Wu, S.; Wu, T.; Tan, H.; and Guo, G. 2021b. Pale Transformer: A General Vision Transformer Backbone with Pale-Shaped Attention. *arXiv preprint arXiv:2112.14000*.

Wu, X.; Zhan, C.; Lai, Y.-K.; Cheng, M.-M.; and Yang, J. 2019. Ip102: A large-scale benchmark dataset for insect pest recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8787–8796.

Xu, W.; Xu, Y.; Chang, T.; and Tu, Z. 2021. Co-scale conv-attentional image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9981–9990.

Yang, J.; Li, C.; Zhang, P.; Dai, X.; Xiao, B.; Yuan, L.; and Gao, J. 2021. Focal self-attention for local-global interactions in vision transformers. *arXiv preprint arXiv:2107.00641*.

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

Zhang, A.; Lipton, Z. C.; Li, M.; and Smola, A. J. 2021. Dive into Deep Learning. *arXiv preprint arXiv:2106.11342*.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*.