

LocalViT: Analyzing Locality in Vision Transformers

Yawei Li¹, Kai Zhang¹, Jiezhong Cao¹, Radu Timofte^{1,2}, Michele Magno³, Luca Benini^{4,5}, Luc Van Gool^{1,6}

Abstract—The aim of this paper is to study the influence of locality mechanisms in vision transformers. Transformers originated from machine translation and are particularly good at modelling long-range dependencies within a long sequence. Although the global interaction between the token embeddings could be well modelled by the self-attention mechanism of transformers, what is lacking is a locality mechanism for information exchange within a local region. In this paper, locality mechanism is systematically investigated by carefully designed controlled experiments. We add locality to vision transformers into the feed-forward network. This seemingly simple solution is inspired by the comparison between feed-forward networks and inverted residual blocks. The importance of locality mechanisms is validated in two ways: 1) A wide range of design choices (activation function, layer placement, expansion ratio) are available for incorporating locality mechanisms and proper choices can lead to a performance gain over the baseline, and 2) The same locality mechanism is successfully applied to vision transformers with different architecture designs, which shows the generalization of the locality concept. For ImageNet2012 classification, the locality-enhanced transformers outperform the baselines Swin-T [1], DeiT-T [2] and PVT-T [3] by 1.0%, 2.6% and 3.1% with a negligible increase in the number of parameters and computational effort. Code is available at <https://github.com/ofsoundof/LocalViT>.

I. INTRODUCTION

Recent advances in machine learning (ML) research, such as computer vision and natural language processing, have been driven by backbone models that can be adapted to different problems [4], [5]. However, the foundation models in vision and language tend to be heavy and computational expensive, thus hindering their applicability in edge devices such as drones and robotics. Thus, how to improve the efficiency of neural models such as convolutional neural networks (CNNs) and transformers becomes important.

CNNs are based on locality in that convolutional filters only perceive a local region of the input image, *i.e.* the receptive field. By stacking multiple layers, the effective receptive fields of a deep neural network can be enlarged progressively. This design enables the network to learn a hierarchy of deep features, which is essential for the success of CNNs. Meanwhile, the local, repetitive connections save many parameters compared with fully connected layers. Yet,

one problem is that a larger receptive field can only be achieved by combining layers, despite alternative attempts at enlarging the receptive field [6].

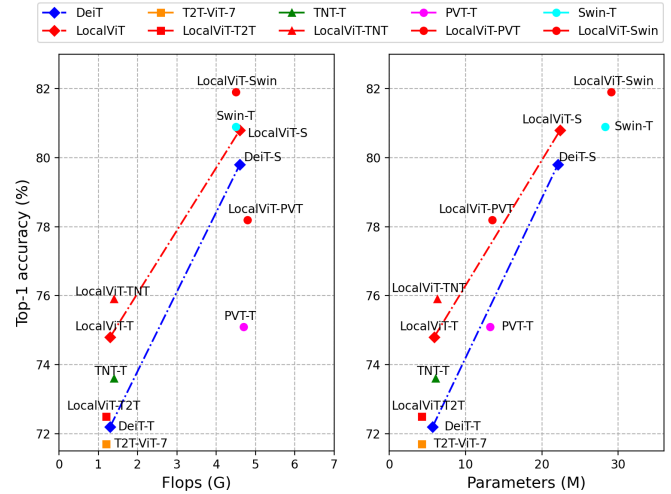


Fig. 1: Comparison between LocalViT and the baseline transformers. The transformers enhanced by the proposed locality mechanism outperform their baselines

A parallel research strand incorporates global connectivity into the network via self-attention [4], [7], [8], [9], [10]. This family of networks, *i.e.* transformer networks, originates from machine translation and is very good at modelling long-range dependencies in sequences. There also is a rising interest in applying transformers to vision [11], [5], [2]. Vision transformers have already achieved performances quite competitive with their CNN counterparts.

To process 2D images with transformers, the input image is first converted to a sequence of tokens which correspond to patches in the image. Then the attention module attends to all tokens and a weighted sum is computed as the tokens for the next layer. In this way, the effective receptive field is expanded to the whole image via a single self-attention layer. Yet, the problem of visual transformers is that global connectivity contradicts the convolutional idea.

Considering the merits of CNNs vs. transformers, a natural question is *whether we can efficiently combine the locality of CNNs and the global connectivity of vision transformers to improve performance while not increasing model complexity.*

This aim of this paper is aligned with other works that try to answer this interesting question, *i.e.* taking the advantage of both convolution and transformers [12], [13], [14], [15], [16], [17]. Differently, we provide a systematic analysis

¹Computer Vision Lab, D-ITET, ETH Zürich, Switzerland. E-mail: yawei.li@vision.ee.ethz.ch.

²Center for Artificial Intelligence and Data Science (CAIDAS), the University of Würzburg, Germany.

³Center for Project-Based Learning, D-ITET, ETH Zürich, Switzerland.

⁴Integrated Systems Laboratory, D-ITET, ETH Zürich, Switzerland.

⁵Department of Electrical, Electronic and Information Engineering, University of Bologna, Italy.

⁶Processing Speech and Images (PSI), KU Leuven, Belgium.

of various design choices of the locality mechanism by rigorous and controlled experiments. Beyond that, we try to generalize the conclusions from the ablation study to a bunch of vision transformers. Thus, the aim of this paper is to thoroughly investigate a single component (locality mechanism) in vision transformers.

To conduct the investigation, we start with a mechanism that injects locality into the feed-forward network of transformers, which is inspired by examining the feed-forward network and inverted residuals [18], [19]. The feed-forward network of transformers consists of two fully connected layers and the hidden dimension between them is expanded to extract richer features. Similarly, in inverted residual blocks, the hidden channel between the two 1×1 convolutions is also expanded. The major difference between them is the efficient depth-wise convolution in the inverted residual block. Such convolution can provide precisely the mechanism for local information aggregation which is missing in the feed-forward network of vision transformers. To cope with the convolution, the image tokens of the sequence from the self-attention module must be rearranged to a 2D feature map, which is processed by the feed-forward network. The class token is split out and bypasses the feed-forward network. The derived new feature map is converted back to image tokens and concatenated with the bypassed class token. The concatenated sequence is processed by the next transformer layer.

Through the empirical study, we derive two sets of conclusions. **Firstly**, four important properties of the investigated locality mechanism are revealed. *i.* Locality mechanism alone can already improve the performance of the baseline transformer. *ii.* A better activation function can result in a significant performance gain. *iii.* The locality mechanism is more important for lower layers. *iv.* Expanding the hidden dimension of the feed-forward network leads to a larger model capacity and a higher classification accuracy. **Secondly**, as shown in Fig. 1, the locality mechanism is successfully applied to 5 vision transformers, which underlines its generality. The contributions of this paper are three-fold:

- 1) We study a locality mechanism that enhances vision transformers. The modified transformer architecture combines a self-attention mechanism for global relationship modelling and a locality mechanism for local information aggregation.
- 2) We analyze the basic properties of the introduced locality mechanism. The influence of each component (depth-wise convolution, non-linear activation function, layer placement, and hidden dimension expansion ratio) is singled out.
- 3) We apply these ideas to vision transformers incl. DeiT [2], Swin transformers [1], T2T-ViT [20], PVT [3], and TNT [21]. Experiments show that the simple technique proposed in this paper generalizes well to various transformer architectures.

II. RELATED WORK

A. Transformers

Transformers were first introduced in [4] for machine translation. The proposed attention mechanism aggregates information from the whole input sequence. Thus, transformers are especially good at modelling long-range dependencies between elements of a sequence. Since then, there have been several attempts to adapt transformers towards vision and robotics [11], [22], [23], [24], [25], [26], [27]. Most recently, transformers are proposed to solve robotic problems by learning the mapping from language and vision observations to robot actions [23].

B. Locality vs. global connectivity

Both local information and global connectivity help to reason about the relationships between image contents. The convolution operation applies a sliding window to the input and local information is inherently aggregated to compute new representations. Thus, locality is an intrinsic property of CNNs [28]. Although CNNs can extract information from a larger receptive field by stacking layers and forming deep networks, they still lack global connectivity [29], [30], [31]. To overcome this problem, some researchers add global connectivity to CNNs with non-local blocks [32], [33].

By contrast, transformers are especially good at modelling long-range dependencies within a sequence owing to their attention mechanism [4]. But, in return, a locality mechanism remains to be added for visual perception. Some works already contributed towards this goal [20], [1], [34], [35], [36], [37]. Those works mainly focus on improving the tokenization and self-attention parts. There are also some works that introduce hybrid architectures of CNNs and transformers [15], [16]. In the meanwhile, we also noticed some other works introducing convolutions to different parts of transformers [17], [12]. The difference between our work and the other works is that we systematically investigate locality mechanism and single out its importance to transformer architectures. This is inspired by the comparison between vision transformers and the inverted residual blocks in MobileNets.

C. Inverted residuals

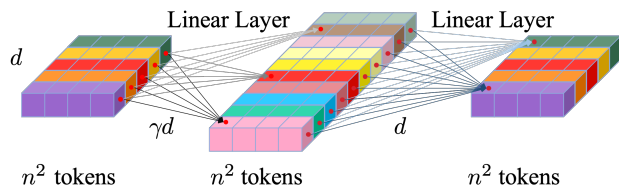
Compared with normal convolution, the computations of depth-wise convolution are only conducted channel-wise. That is, to obtain a channel of the output feature map, the convolution is only conducted on one input feature map. Thus, depth-wise convolution is efficient both in terms of parameters and computation. Thus, Howard *et al.* first proposed the MobileNet architecture based on depth-wise separable convolutions [38]. This lightweight and computationally efficient network is quite friendly for mobile devices. Since then, depth-wise convolution has been widely used to design efficient models. Inverted residual blocks are based on depth-wise convolution and were first introduced in MobileNetV2 [18]. The inverted residual blocks are composed of a sequence of 1×1 - depth-wise -1×1 convolutions. The hidden dimension between the two 1×1 convolutions

is expanded. The utilization of depth-wise convolution avoids the drastic increase of model complexity brought by normal convolution. Due to the efficiency of this module, it is widely used to form the search space of neural architecture search (NAS) [19], [39], [40]. The expansion of the hidden dimension of inverted residuals is quite similar to the feed-forward network of vision transformers. This motivates us to think about the connection between them (See Sec. III-B).

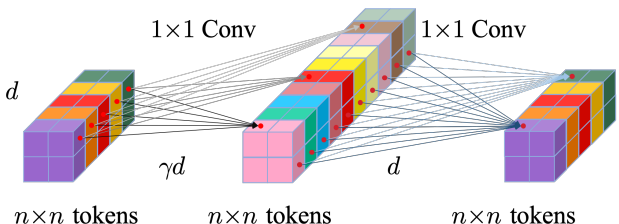
III. METHODOLOGY

Transformers are usually composed of encoders and decoders with similar building blocks. For the image classification task considered here, only the encoders are included in the network. Thus, we mainly describe the operations in the encoder layers. The encoders have two components, *i.e.* the self-attention mechanism that relates a token to all of the tokens and a feed-forward network that is applied to every token. We specifically explain how to introduce locality into the feed-forward network.

A. Input interpretation



(a) The input is regarded as a sequence of tokens.



(b) An equivalent perspective is to still rearrange the tokens as a 2D lattice.

Fig. 2: Visualization of the feed-forward network in transformers from different perspectives. In this figure, $n = 2$, $\gamma = 2$, $d = 5$.

Sequence perspective. Inherited from language modelling, transformers regard the input as a sequence that contains elements of embedded vectors. Consider an input image $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$, where C and $H \times W$ denote the channel and spatial dimension of the input image, resp. The input image is first converted to tokens $\{\hat{\mathbf{X}}_i \in \mathbb{R}^d | i = 1, \dots, N\}$, where $d = C \times p^2$ is the embedding dimension and $N = \frac{HW}{p^2}$. The tokens can be aggregated into a matrix $\hat{\mathbf{X}} \in \mathbb{R}^{N \times d}$.

Self-attention. In the self-attention mechanism, the relationship between the tokens is modelled by the similarity between the projected query-key pairs, yielding the attention

score. The new tokens are computed as the weighted sum of the project values. That is,

$$\mathbf{Z} = \text{Softmax}(\mathbf{Q}\mathbf{K}^T/\sqrt{d})\mathbf{V}, \quad (1)$$

where the Softmax function is applied to the rows of the similarity matrix and d provides a normalization. The query, key, and value are a projection of the tokens, *i.e.* $\mathbf{Q} = \hat{\mathbf{X}}\mathbf{W}_Q$, $\mathbf{K} = \hat{\mathbf{X}}\mathbf{W}_K$, $\mathbf{V} = \hat{\mathbf{X}}\mathbf{W}_V$. The projection matrices \mathbf{W}_Q and \mathbf{W}_K have the same size while \mathbf{W}_V could have a different size. In practice, the three projection matrices usually have the same size, *i.e.* $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times d}$.

Feed-forward network. After the self-attention layer, a feed-forward network is appended. The feed-forward network consists of two fully-connected layers and transforms the features along the embedding dimension. The hidden dimension between the two fully-connected layers is expanded to learn a richer feature representation. That is,

$$\mathbf{Y} = f(\mathbf{Z}\mathbf{W}_1)\mathbf{W}_2, \quad (2)$$

where $\mathbf{W}_1 \in \mathbb{R}^{d \times \gamma d}$, $\mathbf{W}_2 \in \mathbb{R}^{\gamma d \times d}$, and $f(\cdot)$ denotes a non-linear activation function. For the sake of simplicity, the bias term is omitted. The dimension expansion ratio γ is usually set to 4. As shown in Fig. 2a, the input to the feed-forward network is regarded as a sequence of embedding vectors.

Lattice perspective. Since the feed-forward network is applied position-wise to a sequence of tokens $\mathbf{Z} \in \mathbb{R}^{N \times d}$, an exactly equivalent representation is to rearrange the sequence of tokens into a 2D lattice as shown in Fig. 2b. Then the reshaped feature representation is

$$\mathbf{Z}^r = \text{Seq2Img}(\mathbf{Z}), \mathbf{Z}^r \in \mathbb{R}^{h \times w \times d}, \quad (3)$$

where $h = H/p$ and $w = W/p$. The operation Seq2Img converts a sequence to a 2D feature map. Each token is placed to a pixel location of the feature map. The benefit of this perspective is that the proximity between tokens is recovered, which provides the chance to introduce locality into the network. The fully-connected layers could be replaced by 1×1 convolutions, *i.e.*

$$\mathbf{Y}^r = f(\mathbf{Z}^r \otimes \mathbf{W}_1^r) \otimes \mathbf{W}_2^r, \quad (4)$$

$$\mathbf{Y} = \text{Img2Seq}(\mathbf{Y}^r), \quad (5)$$

where $\mathbf{W}_1^r \in \mathbb{R}^{d \times \gamma d \times 1 \times 1}$ and $\mathbf{W}_2^r \in \mathbb{R}^{\gamma d \times d \times 1 \times 1}$ are reshaped from \mathbf{W}_1 and \mathbf{W}_2 and represent the convolutional kernels. The operation Img2Seq converts the image feature map back to a token sequence which is used by the next self-attention layer.

B. Locality

Since only 1×1 convolution is applied to the feature map, there is a lack of information interaction between adjacent pixels. Besides, the self-attention part of the transformer only captures global dependencies between all of the tokens. Thus, the transformer block does not have a mechanism to model the local dependencies between nearby pixels. It would be interesting if locality could be brought to transformers in an efficient way.

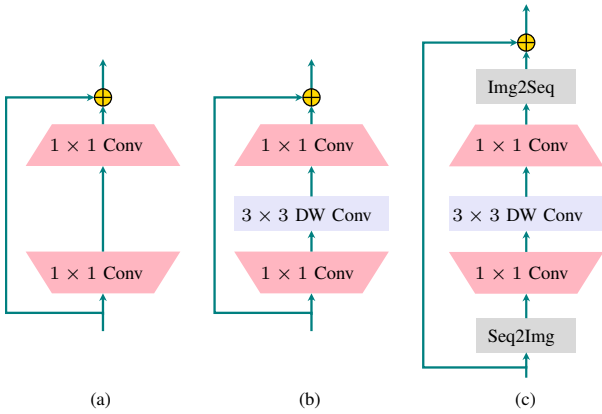


Fig. 3: Comparison between the (a) convolutional version of the feed-forward network in vision transformers, the (b) inverted residual blocks, and (c) the utilized network that brings locality mechanism into transformers. “DW” denotes depth-wise convolution. To cope with the convolution operation, the conversion between sequence and image feature map is added by “Seq2Img” and “Img2Seq” in (c). Note that after each convolution, there are activation functions. In Table II, we systematically investigate influence of the activation function after the depthwise convolution in (c).

The expansion of the hidden dimension between fully-connected layers and the lattice perspective of the feed-forward network remind us of the inverted residual block proposed in MobileNets [18], [19]. As shown in Fig. 3, both of the feed-forward network and the inverted residual expand and squeeze the hidden dimension by 1×1 convolution. The only difference is that there is a depth-wise convolution in the inverted residual block. Depth-wise convolution applies a $k \times k$ ($k > 1$) convolution kernel per channel. The features inside the $k \times k$ kernel is aggregated to compute a new feature. Thus, depth-wise convolution is an efficient way of introducing locality into the network. Considering that, we reintroduce depth-wise convolution into the feed-forward network of transformers. And the computation could be represented as

$$\mathbf{Y}^r = f(f(\mathbf{Z}^r \circledast \mathbf{W}_1^r) \circledast \mathbf{W}_d) \circledast \mathbf{W}_2^r, \quad (6)$$

where $\mathbf{W}_d \in \mathbb{R}^{\gamma d \times 1 \times k \times k}$ is the kernel of the depth-wise convolution. The finally used network is shown in Fig. 3c. The input, *i.e.* a sequence of tokens is first reshaped to a feature map rearranged on a 2D lattice. Then two 1×1 convolutions along with a depth-wise convolution are applied to the feature map. After that, the feature map is reshaped to a sequence of tokens which are used as by the self-attention of the network transformer layer.

Note that the non-linear activation functions are not visualized in Fig. 3. Yet, they play a quite important role in enhancing the network capacity, especially for efficient networks. In particular, we try ReLU6, h-swish [19], squeeze-and-excitation (SE) module [41], efficient channel attention (ECA) module [42], and their combinations. A thorough

analysis of the activation function is discussed in the experiments section.

C. Class token

To apply vision transformers to image classification, a trainable class token is added and inserted into the token embedding, *i.e.*

$$\hat{\mathbf{X}} \leftarrow \text{Concat}(\mathbf{X}_{cls}, \hat{\mathbf{X}}), \quad (7)$$

where \leftarrow denotes the assignment operation, $\mathbf{X}_{cls} \in \mathbb{R}^{1 \times d}$ is the class token. The new matrix has the dimension of $(N + 1) \times d$ and $N + 1 = \frac{HW}{p^2} + 1$ tokens. In the self-attention module, the class token exchanges information with all other image tokens and gathers information for the final classification. In the feed-forward network, the same transformation is applied to the class and image tokens.

When depth-wise convolution is introduced into the feed-forward network, the sequence of tokens needs to be rearranged into an image feature map. Yet, the additional dimension brought by the class token makes the exact rearrangement impossible. To circumvent this problem, we split the $N + 1$ tokens in Eqn. (1) into a class token and image tokens again, *i.e.*

$$(\mathbf{Z}_{cls}, \mathbf{Z}) \leftarrow \text{Split}(\mathbf{Z}). \quad (8)$$

Then the new image token is passed through the feed-forward network according to Eqns. (3), (6), and (5), leading to \mathbf{Y} . The class token is not passed through the feed-forward network. Instead, it is directly concatenated with \mathbf{Y} , *i.e.*

$$\mathbf{Y} \leftarrow \text{Concat}(\mathbf{Z}_{cls}, \mathbf{Y}). \quad (9)$$

The split and concatenation of the class token is done for every layer. Although the class token \mathbf{Z}_{cls} is not passed through the feed-forward network, the performance of the overall network is not adversely affected. This is because the information exchange and aggregation is done only in the self-attention part. A feed-forward network like Eqn. (2) only enforces a transformation within each token.

IV. EXPERIMENTAL RESULTS

This section gives the experimental results for image classification. We first study how the locality brought by depth-wise convolution can improve the performance of transformers. Then we investigate the influence of several design choices including the non-linear activation function, the placement of the locality, and the hidden dimension expansion ratio γ . All those experiments are based on DeiT-T [2]. Then, we study the generalization to other vision transformers including T2T-ViT [20], PVT [3], TNT [21], Swin transformer [1] for image classification. The transformers that are equipped with locality are denoted as LocalViT followed by the suffix that denotes the basic architecture.

A. Implementation details

We introduce the locality mechanism into five vision transformers including DeiT [2], Swin transformers [1], T2T-ViT [20], PVT [3], TNT [21]. Since those transformers have different architectures, slightly different considerations should be made. First of all, a Tokens-to-Token (T2T) module is designed in T2T-ViT [20] and is inserted into the head of the network. Basically, the T2T module is also a transformer block with feed-forward networks. Thus, the same modification is also applied to the T2T module. Secondly, TNT introduced an inner transformer block for the image tokens along with the outer transformer block. Yet, we observed huge increase of GPU memory. Thus, the locality mechanism is only applied to the outer transformer block of TNT [21]. Thirdly, for PVT [3], the class token is only introduced in the final stage of the pyramid. Thus, the split and concatenation of the class token for the feed-forward network is only applied in the final stage. Fourthly, there is no class token in Swin transformers [1]. The classification is done based on an averaged pooled feature map. Thus, the special treatment of the class token is not needed in our modified Swin transformers.

For fast experiment, we shrink TNT and Swin transformers and get smaller versions of them. TNT-T is derived by reducing the embedding dimension from 384 to 192. Swin-M is derived by reducing the number of transformer blocks in the third stage from 6 to 2.

Experimental setup. The ImageNet2012 dataset [43] is used in this paper. The dataset contains 1.28M training images and 50K validation images from one thousand classes. We follow the same training protocol as DeiT [2]. The input image is randomly cropped with size 224×224 . Cross-entropy is used as the loss function. Label smoothing is used. The weight decay factor is set to 0.05. The AdamW optimizer is used with a momentum of 0.9. The training continues for 300 epochs. The batch size is set to 1024. The initial learning rate is set to 1×10^{-3} and decreases to 1×10^{-5} following a cosine learning rate scheduler. During validation, a center crop of the validation images is conducted. We use 8 NVIDIA TITAN RTX GPUs to run the experiments.

B. Influence of the locality

TABLE I: Investigation of the locality brought by depth-wise convolution. *ReLU6 is used as the activation function after depth-wise convolution. †Results derived by modifying the DeiT architecture and training the network with the same training protocol

Network	γ	Depthwise Conv	Params (M)	Top-1 Acc. (%)
DeiT-T [2]	4	No	5.7	72.2
LocalViT-T	4	No	5.7	72.5 (0.3†)
LocalViT-T*	4	Yes	5.8	73.7 (1.5†)
DeiT-T [2]	6	No	7.5	73.1†
LocalViT-T	6	No	7.5	74.3 (1.2†)
LocalViT-T*	6	Yes	7.7	76.1 (3.0†)

We first study how the local information could help to improve the performance of vision transformers in Table I.

TABLE II: Investigation of the non-linear activation function. The combination of HS, ECA [42], and SE [41] is studied. “HS” means h-swish activation. “SE-***” means the reduction ratio in the squeeze-and-excitation module. γ is set to 4.

Activation	Params (M)	Top-1 Acc. (%)
DeiT-T [2]	5.7	72.2
ReLU6	5.8	73.7 (1.5†)
HS	5.8	74.4 (2.2†)
HS + ECA	5.8	74.5 (2.3†)
HS + SE-192	5.9	74.8 (2.6†)
HS + SE-96	6.0	74.8 (2.6†)
HS + SE-48	6.1	75.0 (2.8†)
HS + SE-4	9.4	75.8 (3.6†)

Different hidden dimension expansion ratios γ are investigated. First of all, due to the change of the operations in the feed-forward network (Sec. IV-A), the Top-1 accuracy of LocalViT-T is slightly increased even without the depth-wise convolution. The performance gain is 0.3% for $\gamma = 4$ and is increased to 1.2% for $\gamma = 6$. Note that compared with DeiT-T, no additional parameters and computation are introduced for the improvement. When locality is incorporated into the feed-forward network, there is a significant improvement of the model accuracy, *i.e.* 1.5% for $\gamma = 4$ and 3.0% for $\gamma = 6$. Compared with the baseline, there only is a marginal increase in the number of parameters and a negligible increase in the amount of computation. *Thus, the performance of vision transformers can be significantly improved by the incorporation of a locality mechanism and the adaptation of the operation in the feed-forward network.*

C. Activation functions

The non-linear activation function after depth-wise convolution used in the above experiments is simply ReLU6. The benefit of using other non-linear activation functions is also studied. In Table II, the ablation study based on LocalViT-T is done. First of all, by replacing the activation function from ReLU6 to h-swish, the gain of Top-1 accuracy over the baseline is increased from 1.5% to 2.2%. This shows the benefit of h-swish activation functions can be easily extended from CNNs to vision transformers. Next, the h-swish activation function is combined with other channel attention modules including ECA [42] and SE [41]. The ECA and SE modules are placed directly after the h-swish function. By adding ECA, the performance is further improved by 0.1%. Considering that only 60 parameters are introduced, this improvement is still considerable under a harsh parameter budget.

Another significant improvement is brought by a squeeze-and-excitation module. When the reduction ratio in the squeeze-and-excitation module is reduced from 192 to 4, the gain of Top-1 accuracy is gradually increased from 2.6% to 3.6%. The number of parameters is also increased accordingly. Note that, for all of the networks, the computational complexity is almost the same. This implies that if there is no strict limitation on the number of parameters, advanced non-linear activation functions could be used. In the following

TABLE III: Influence of the placement of locality. “All” means all of the transformer layers are enhanced by depth-wise convolution. “Low”, “Mid”, and “High” mean the lower, middle, and higher transformer layers are equipped with depth-wise convolution, respectively. The study is based on LocalViT-T.

Layer Placement	Params (M)	FLOPs (G)	Top-1 Acc. (%)
High: 9~12	5.78	1.26	69.1
Mid: 5~8	5.78	1.26	72.1
Low: 1~4	5.78	1.26	73.1
Low: 1~8	5.84	1.27	74.0
All: 1~12	5.91	1.28	74.8

TABLE IV: Investigating the expansion ratio of hidden layers in the feed-forward network.

γ	SE	Params (M)	FLOPs (G)	Top-1 Acc. (%)
1	No	3.1	0.7	65.9
	Yes	3.1	0.7	66.2
2	No	4.0	0.9	70.1
	Yes	4.0	0.9	70.6
3	No	4.9	1.1	72.9
	Yes	5.0	1.1	73.1
4	No	5.8	1.3	74.4
	Yes	5.9	1.3	74.8

experiments, we use the combination of h-swish and SE as the non-linear activation function after depth-wise convolution. Additionally, the reduction ratio of the squeeze-and-excitation module is chosen such that only 4 channels are kept after the squeeze operation. This choice of design achieves a good balance between the number of parameters and the model accuracy. *Thus, local information is also important in vision transformers. A wide range of efficient modules could be introduced into the feed-forward network of vision transformers to expand the network capacity.*

D. Placement of locality, expansion ratio, and discussion

The transformer layers where the locality is introduced can also influence the performance of the network. Thus, an ablation study based on LocalViT-T is conducted to study their effect. The results is reported in Table III. There are in total 12 transformer layers in the network. We divide the 12 layers into 3 groups corresponding to “Low”, “Mid”, and “High” stages. For the former 3 rows of Table III, we independently insert locality into the three stages. As the locality is moved gradually from lower stages to the higher stages, the accuracy of the network is decreased. This shows that local information is especially important for the lower layers. This is also consistent with our intuition. When the depth-wise convolution is applied to the lower layers, the local information aggregated there could also be propagated to the higher layers, which is important to improve the overall performance of the network.

When the locality is introduced only in the higher stage, the Top-1 accuracy is even lower than DeiT-T. To investigate whether locality in the higher layers always has an adverse effect, we progressively allow more lower layers to have

TABLE V: Image classification results for different CNNs and vision transformers. The locality functionality is enabled for five different vision transformers

Network	Params (M)	FLOPs (G)	Top-1 Acc. (%)	Top-5 Acc. (%)
CNNs				
ResNet-18 [31]	11.7	1.8	69.8	89.1
ResNet-50 [31]	25.6	4.1	76.1	92.9
DenseNet-169 [44]	14.2	3.4	75.6	92.8
RegNet-4GF [45]	20.7	4.0	80.0	–
MobileNetV1 [38]	4.2	0.6	70.6	–
MobileNetV2 [18]	6.9	0.6	74.7	–
EfficientNet-B0 [40]	5.3	0.4	77.1	93.3
Transformers				
DeiT-T [2]	5.7	1.3	72.2	91.1
LocalViT-T	5.9	1.3	74.8 (2.6 \uparrow)	92.6
DeiT-T $\frac{1}{8}$ [2]	5.9	1.3	74.5	–
DeiT-S [2]	22.1	4.6	79.8	95.1
LocalViT-S	22.4	4.6	80.8 (1.0 \uparrow)	95.4
DeiT-S $\frac{1}{8}$ [2]	22.4	4.6	81.2	–
T2T-ViT-7 [20]	4.3	1.2	71.7	–
LocalViT-T2T	4.3	1.2	72.5 (0.8 \uparrow)	–
TNT-T [21]	6.1	1.4	73.6	91.9
LocalViT-TNT	6.3	1.4	75.9 (2.3 \uparrow)	93.0
PVT-T [3]	13.2	4.7	75.1	92.3
LocalViT-PVT	13.5	4.8	78.2 (3.1 \uparrow)	94.2
Swin-M [1]	21.2	3.0	79.2	94.5
LocalViT-Swin-M	21.7	3.0	80.4 (1.2 \uparrow)	95.0
Swin-T [1]	28.3	4.5	80.9	95.3
LocalViT-Swin	29.1	4.5	81.9 (1.0 \uparrow)	95.7

depth-wise convolution until locality is enabled for all layers. This corresponds to the last three rows of Table III. Starting from the lower layers, the performance of the network could be gradually improved as locality is enabled for more layers. *Thus, introducing the locality to the lower layers is more advantageous compared with higher layers.*

The effect of the expansion ratio of the hidden dimension of the feed-forward network is also investigated. The results are shown in Table IV. Expanding the hidden dimension of the feed-forward network can have a significant effect on the performance of the transformers. As the expansion ratio is increased from 1 to 4, the Top-1 accuracy is increased from less than 70% to nearly 75%. The model complexity is also almost doubled. *Thus, the network performance and model complexity can be balanced by the hidden dimension expansion ratio γ . Squeeze-and-excitation can be more beneficial for smaller γ .*

E. Generalization and comparison with state-of-the-art

Finally, we try to incorporate locality into vision transformers including DeiT [2], Swin transformers [1], T2T-ViT [20], TNT [21], PVT [3] and compare their performance with CNNs. We draw three major conclusions from Table V. *Firstly, the effectiveness of locality can be generalized to a wide range of vision transformers based on the following observations. 1)* Compared with DeiT, LocalViT can yield a higher classification accuracy for both the tiny and small version of the network. The increase of Top-1 accuracy is 2.6% and 1.0%, resp. LocalViT-T even outperforms DeiT-T $\frac{1}{8}$ which is enhanced by knowledge distillation from RegNetY-160 [45]. The small version LocalViT-S is slightly worse

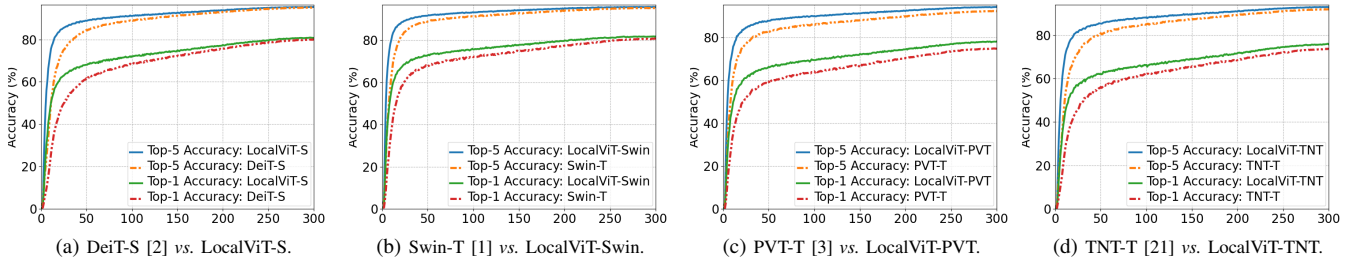


Fig. 4: Comparison of Top-1 and Top-5 accuracy between the baseline transformers and the locality enhanced LocalViT

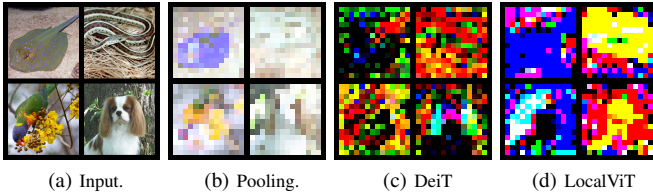


Fig. 5: Comparison of feature maps. (a) Random sampled input images from ImageNet (b) Max pooled images with kernel size 16×16 . (c) & (d) Feature map of the last transformer layer from DeiT and LocalViT

than DeiT-S₈ by 0.4%. 2) LocalViT-T2T outperforms T2T-ViT-7 by 0.8%. Note that T2T-ViT already tries to model the local structure information in the tokens-to-token module. 3) In TNT, an additional transformer block is used to extract local features for the image tokens. Thus, the locality is also considered in TNT. The modified network, *i.e.* LocalViT-TNT could still improve the classification accuracy by a large margin of 2.3%. 4) The biggest improvement comes from PVT. Introducing the locality module leads to a gain of 3.1% over PVT-T. 5) Swin transformer already adopts shifted windows that constrain attention in a local region. Yet, adding locality processing module into the network could still improve the performance of Swin transformers. 6) The comparison of the training log between the baseline transformers and LocalViT is shown in Fig. 4. As shown in Fig. 4, during the training phase, LocalViT outperforms the baseline transformers consistently in terms of both Top-1 and Top-5 accuracy. The gap between LocalViT and the baseline transformers is more obvious in the early training phase. For example, the gap of the Top-1 accuracy between LocalViT-T and DeiT-T could be as large as 10% during the early training phase (at about Epoch 25). Thus, this confirms that the locality mechanism introduced by LocalViT can enlarge the capacity of vision transformers and lead to better performances.

Secondly, some versions of the enhanced vision transformer LocalViT are already quite comparable or even outperform CNNs. This conclusion can be drawn by making the pairwise comparison, *i.e.* LocalViT-T vs. MobileNetV2 (1.4), LocalViT-S vs. ResNet-50, LocalViT-T2T vs. MobileNetV1, LocalViT-PVT vs. DenseNet-169 *etc.*

Thirdly, by comparing the feature maps of transformers with and without locality mechanism in Fig. 5c and Fig. 5d, it is clear that LocalViT does a better job at localizing the objects in the presented input images.

Discussion of limitation. As shown in Table V, introducing locality mechanism increases the complexity of the network. As a result, the inference of the network could be slowed down. We report the throughput of different methods on one NVIDIA TITAN Xp GPU in Table VI. When comparing with the DeiT-T and TNT-T, there is only a marginal decrease (less than 10%) of throughput for LocalViT. The throughput of LocalViT-T and DeiT-T is almost comparable. For PVT-T and Swin-T, the locality enhanced transformer LocalViT is faced with a larger throughput decrease. But the decrease is still within 24%. Yet, considering the non-trivial improvement of the locality enhanced transformer network, we think this is acceptable. The value of this paper is to show the importance of locality mechanism in vision transformers rather than achieving state-of-the-art performances. And we believe that the conclusion derived under rigorous and controlled experiments could help the community to understand the locality mechanism in vision transformers. Combination with other design choices towards more efficient networks could be done in follow-up works.

TABLE VI: Throughput comparison between the baseline networks and those enhanced by the locality mechanism. The inference is conducted on a single NVIDIA TITAN Xp GPU

Network	Throughput images/s	Network	Throughput images/s
DeiT-T	346.3	PVT-T	318.4
LocalViT-T	336.9 (2.7% ↓)	LocalViT-PVT	248.3 (22.0% ↓)
TNT-T	222.0	Swin-T	206.3
LocalViT-TNT	208.3 (6.2% ↓)	LocalViT-Swin	158.2 (23.3% ↓)

V. CONCLUSION

In this paper, we investigated the influence of locality mechanism in the feed-forward of vision transformers. We introduced the locality mechanism into vision transformers by incorporating 2D depth-wise convolutions followed by a non-linear activation function into the feed-forward network of vision transformers. To cope with the locality mechanism, the sequence of tokens embedding is rearranged into a lattice as a 2D feature map, which is used as the input to the enhanced feed-forward network. To enable the rearrangement,

the class token is split before the feed-forward network and concatenated with other image embeddings after the feed-forward network. A series of studies were made to investigate various factors (activation function, layer placement, and expansion ratio) that might influence of performance of the locality mechanism. The proposed locality mechanism is successfully applied to five different vision transformers, which validates its generality.

REFERENCES

- [1] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv preprint arXiv:2103.14030*, 2021.
- [2] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," *arXiv preprint arXiv:2012.12877*, 2020.
- [3] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," *arXiv preprint arXiv:2102.12122*, 2021.
- [4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.
- [5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [6] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
- [7] H. Wang, Z. Wu, Z. Liu, H. Cai, L. Zhu, C. Gan, and S. Han, "Hat: Hardware-aware transformers for efficient natural language processing," *arXiv preprint arXiv:2005.14187*, 2020.
- [8] S. Mehta, M. Ghazvininejad, S. Iyer, L. Zettlemoyer, and H. Hajishirzi, "Delight: Very deep and light-weight transformer," *arXiv preprint arXiv:2008.00623*, 2020.
- [9] Z. Wu, Z. Liu, J. Lin, Y. Lin, and S. Han, "Lite transformer with long-short range attention," *arXiv preprint arXiv:2004.11886*, 2020.
- [10] Y. Li, Y. Fan, X. Xiang, D. Demandolx, R. Ranjan, R. Timofte, and L. Van Gool, "Efficient and explicit modelling of image hierarchies for image restoration," in *Proc. CVPR*, 2023, pp. 18 278–18 289.
- [11] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. ECCV*. Springer, 2020, pp. 213–229.
- [12] J. Guo, K. Han, H. Wu, C. Xu, Y. Tang, C. Xu, and Y. Wang, "Cmt: Convolutional neural networks meet vision transformers," *arXiv preprint arXiv:2107.06263*, 2021.
- [13] Y. Liu, G. Sun, Y. Qiu, L. Zhang, A. Chhatkuli, and L. Van Gool, "Transformer in convolutional neural networks," *arXiv preprint arXiv:2106.03180*, 2021.
- [14] K. Yuan, S. Guo, Z. Liu, A. Zhou, F. Yu, and W. Wu, "Incorporating convolution designs into visual transformers," *arXiv preprint arXiv:2103.11816*, 2021.
- [15] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," *arXiv preprint arXiv:2101.11605*, 2021.
- [16] C. Li, T. Tang, G. Wang, J. Peng, B. Wang, X. Liang, and X. Chang, "Bossnas: Exploring hybrid cnn-transformers with block-wisely self-supervised neural architecture search," *arXiv preprint arXiv:2103.12424*, 2021.
- [17] S. d'Ascoli, H. Touvron, M. Leavitt, A. Morcos, G. Biroli, and L. Sagan, "Convit: Improving vision transformers with soft convolutional inductive biases," *arXiv preprint arXiv:2103.10697*, 2021.
- [18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. CVPR*, 2018, pp. 4510–4520.
- [19] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, "Searching for mobilenetv3," in *Proc. ICCV*, 2019, pp. 1314–1324.
- [20] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, F. E. Tay, J. Feng, and S. Yan, "Tokens-to-token vit: Training vision transformers from scratch on imagenet," *arXiv preprint arXiv:2101.11986*, 2021.
- [21] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer," *arXiv preprint arXiv:2103.00112*, 2021.
- [22] M. Saleh, Y. Wang, N. Navab, B. Busam, and F. Tombari, "Clou-dattention: Efficient multi-scale attention scheme for 3d point cloud learning," in *Proc. IROS*. IEEE, 2022, pp. 1986–1992.
- [23] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.
- [24] C. Shi, Y. Zheng, and A. M. Fey, "Recognition and prediction of surgical gestures and trajectories using transformer models in robot-assisted surgery," in *Proc. IROS*. IEEE, 2022, pp. 8017–8024.
- [25] S. Lee, E. Yi, J. Lee, J. Yoo, H. Lee, and S. H. Kim, "Fully convolutional transformer with local-global attention," in *Proc. IROS*. IEEE, 2022, pp. 552–559.
- [26] E. V. Mascaró, S. Ma, H. Ahn, and D. Lee, "Robust human motion forecasting using transformer-based model," in *Proc. IROS*. IEEE, 2022, pp. 10 674–10 680.
- [27] A. Buckler, L. Figueredo, S. Haddadin, A. Kapoor, S. Ma, and R. Bonatti, "Reshaping robot trajectories using natural language commands: A study of multi-modal data alignment using transformers," in *Proc. IROS*. IEEE, 2022, pp. 978–984.
- [28] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.
- [29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NeurIPS*, 2012, pp. 1097–1105.
- [30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.
- [32] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. CVPR*, 2018, pp. 7794–7803.
- [33] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, "Non-local recurrent network for image restoration," *arXiv preprint arXiv:1806.02919*, 2018.
- [34] J. Li, Y. Yan, S. Liao, X. Yang, and L. Shao, "Local-to-global self-attention in vision transformers," *arXiv preprint arXiv:2107.04735*, 2021.
- [35] W. Wang, L. Yao, L. Chen, D. Cai, X. He, and W. Liu, "Crossformer: A versatile vision transformer based on cross-scale attention," *arXiv e-prints*, pp. arXiv–2108, 2021.
- [36] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, and C. Shen, "Twins: Revisiting the design of spatial attention in vision transformers," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [37] P. Zhang, X. Dai, J. Yang, B. Xiao, L. Yuan, L. Zhang, and J. Gao, "Multi-scale vision longformer: A new vision transformer for high-resolution image encoding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2998–3008.
- [38] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [39] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, "Mnasnet: Platform-aware neural architecture search for mobile," in *Proc. CVPR*, 2019, pp. 2820–2828.
- [40] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *arXiv preprint arXiv:1905.11946*, 2019.
- [41] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.
- [42] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *Proc. CVPR*, 2020.
- [43] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, 2009, pp. 248–255.
- [44] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, 2017.
- [45] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proc. CVPR*, 2020, pp. 10 428–10 436.