Channel Propagation Networks for Refreshable Vision Transformer

Junhyeong Go
Ajou University
gojunhyeong6545@gmail.com

Jongbin Ryu *
Ajou University
jongbinryu@ajou.ac.kr

Abstract

In this paper, we introduce the Channel Propagation method, which aims to increase the channels of the Vision Transformer systematically. Skip connections are commonly acknowledged as a propagation approach that improves the stability of the performance in Vision Transformers. Nevertheless, it is important to note that these skip connections may give rise to the problem of over-smoothing, wherein similar features are represented in multiple layers. To tackle this matter, our proposed approach for Channel Propagation in Vision Transformers retains the present signal information while concurrently propagating location-specific signals in a newly introduced channel dimension. On the other hand, the proposed Channel Propagation method effectively maintains the integrity of identity representation while simultaneously including patch-wise location-specific supervision by introducing a new channel dimension. The inclusion of this approach in Vision Transformers mitigates the issue of over-smoothing while also improving the performance of visual recognition tasks. In our experiments, we confirm that the proposed method is effective for various visual recognition tasks. Specifically, our method demonstrates enhanced performance when implemented on Vision Transformer models; the classification accuracy is increased considerably for plain and hierarchical architectures on the ImageNet dataset.

1. Introduction

Vision Transformer(ViT) employs a self-attention operation to learn the global relation of an image from different spatial patches. This enables ViT to capture long-range dependencies effectively, even with shallow attention layers. The successful implementation of this operation has led to the widespread utilization of ViT in various visual recognition tasks.

However, ViT continues to struggle with a limitation known as the over-smoothing problem, where the network learns highly similar representations across different layers. Previous studies have shed light on the underlying causes of this over-smoothing issue in ViT architectures.

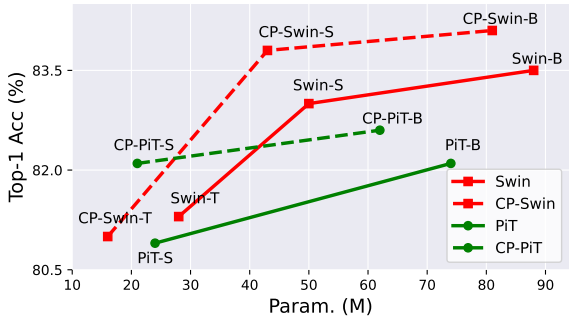

Figure 1. Experimental comparison of the proposed Channel Propagation method with the baseline networks. Our method demonstrates notable enhancements in performance while incurring minimum computational overhead. This improvement is observed across two distinct baseline networks, such as PiT [14] and Swin Transformer [19].

Gong *et al*. [10] conducted a study on the decline in feature diversity for image patches within ViT, leading to their increased similarity. DeepViT [41] delved into the concept of 'attention collapse', which refers to the convergence of attention maps as the model's depth progressively grows. In addition, it can be argued that ViT has a limitation in learning feature diversity in the frequency domain. This is supposed by the finding of Wang *et al*. [28], who observed that as the depth of ViT increases, the self-attention operation tends to disregard high-frequency information, causing a failure to preserve diverse feature representation.

This study investigates the application of skip connections in ViT to enhance the network's ability to capture a broader range of feature representations at each layer through these refreshable skip connections. While the skip connections have a vital role in ensuring the steady training of deep ViTs [7, 22], they also present a significant issue of propagating identical non-refreshing feature representations. This non-refreshing feature propagation is particularly evident in ViT, as it incorporates two skip connections

*Corresponding author.

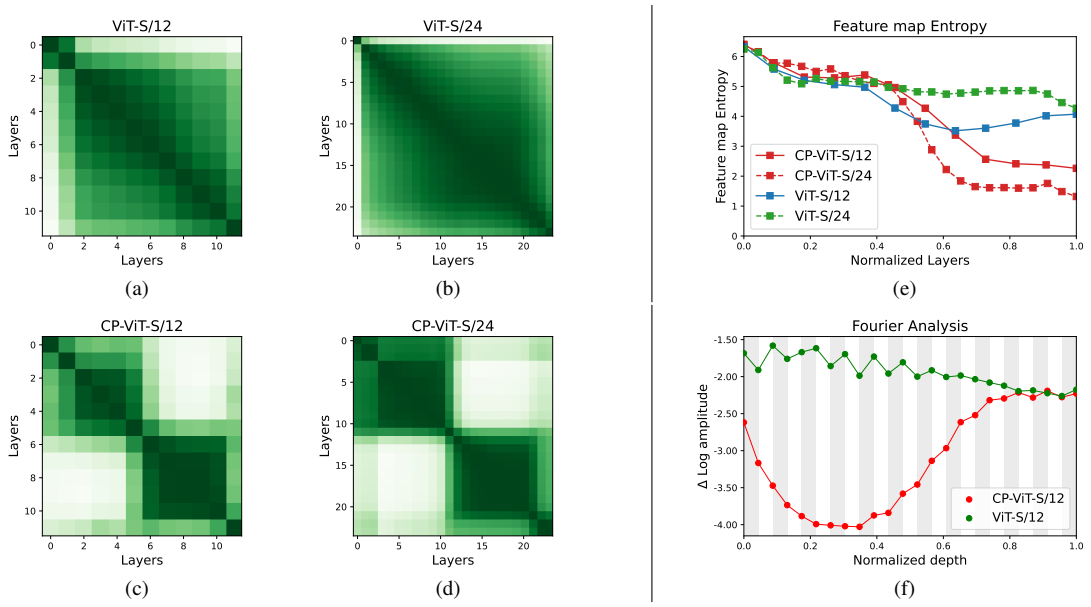

Figure 2. We analyze the degree of similarity in features across different layers of the ViT model [7]. **(a)** and **(b)**: We provide the CKA [17] similarity of the baseline ViT model. The x and y axes of the graph are the indices of the layers. Most layer combinations demonstrate high similarity in these subfigures. **(c)** and **(d)**: The similarity across different layers is presented for our CP-ViT. It shows low similarity in certain layer combinations. Therefore, our CP-ViT has less redundancy compared to the baseline ViT. **(e)**: The spatial entropy of the input features of attention blocks is compared. The x-axis and y-axis denote the normalized layer depth and spatial entropy, respectively. **(f)**: We present a visual representation of the Fourier frequency domains. The y-axis represents the Δ log amplitude.

within a single attention block. To support this claim, we investigate the feature similarity of each layer in ViT through various analyses. We observed that as the depth of ViT increases, the feature similarity also gradually increases. This trend on the feature similarity has also been mentioned in previous studies [3, 6, 22, 34].

Therefore, we propose a novel method for propagating refreshable features on a skip connection in ViT. The proposed method adds refreshable spatial information to the channel direction before the Multi-Head Self-Attention (MHSA) block and subsequently propagates it through skip connections. This enables consolidating diverse attention maps and the progressive learning of refreshable features inside a network. In addition, the proposed approach of employing a gradually expanding channel dimension can effectively address the information loss arising from sudden changes in channel size during the downsampling stage in hierarchical networks, as discussed in the prior work by PyramidNet [11]. The lightweight design of the proposed approach also facilitates the optimization of computational overhead in most ViT architectures.

We validate the image classification performance of the proposed Channel Propagation method; it achieves a high level of performance improvement not only in plain ViTs [25] but also in hierarchical ViTs [14, 19] as shown in Fig. 1. Furthermore, we adopt an affordable channel dimension, resulting in minimal overhead in terms of parameters and FLOPs. We assess the image classification performance of the Channel Propagation method, demonstrating significant improvements in both plain [7, 25] and hierarchical networks such as SwinTransformer [19] and PiT [14]. In addition, we perform empirical studies on frequency and entropy analysis to evaluate the effectiveness of the proposed method in addressing the feature redundancy problem.

2. Method

2.1. Preliminary

The ViT divides the input image into fixed-size patches and augments the positional information of these patches. The input signal is sequentially propagated through Multi-Head Self-Attention (MHSA) blocks and Feed-Forward Network (FFN) blocks, with two skip connections added after each block as follows:

$$\mathrm{X'}_l = \mathrm{X}_l + \mathrm{MHSA}(\mathrm{X}_l), \tag{1}$$

$$\mathrm{X}_{l+1} = \mathrm{X'}_l + \mathrm{FFN}(\mathrm{X'}_l), \tag{2}$$

where l denotes the layer index.

ViT partitions the input image into a predetermined number of patches and enhances the positional information associated with each patch. The input signal undergoes sequential propagation through multiple attention blocks, includ-

ing the two skip connections in an attention block. ViT's attention blocks will inevitably use these skip connections twice, increasing feature redundancy so that each layer in the network learns similar features. We present the following three analyses to investigate the feature representation redundancy of the ViTs due to the skip connection.

CKA similarity. In order to examine the depth-wise propagation signals within the ViT, we first explore the Centered Kernel Alignment (CKA) similarity [17]. As shown in Fig. 2a and Fig. 2b, the skip connection demonstrates notable high similarity throughout all layers. It is noteworthy that the above finding becomes more evident with deeper layers.

Feature entropy. In addition, we measure the spatial entropy $\mathbb{S}$ of the skip connection to assess the spatial entropy of pixels using the spatial distribution of the input images of the blocks as:

$$\mathbb{S} = -\sum p(X_{i,j}) \cdot \log(p(X_{i,j})), \tag{3}$$

where $X_{i,j}$ represents the feature values at i, j position. Higher spatial entropy indicates detailed and local information, whereas lower values signify signals with spatially monotonous and global information [1]. Since these two different information are complementary, learning both higher and lower levels of spatial entropy at different layers can enhance the network due to the diversified feature representation. However, as seen in the right part of Fig. 2e, as the depth gradually increases, the signals propagated within the ViT cannot utilize a diverse range of information during training. This observation implies that the ViT consistently transmits a narrow range of spatial entropy throughout the entire network.

Frequency. Finally, we analyze the network's propagation signal in the frequency domain. Our analysis follows [21], focusing on the features obtained from the MHSA and FFN operations to investigate the frequency. Generally, the self-attention operation of the MHSA is regarded as the low-pass filters that remove high-frequency features [21, 28]. Due to this reason, as shown in Fig. 2f, ViTs tend to retain a biased range of frequency values, which induces redundancy in the frequency domain. The frequency and entropy analyses for the latest models and deeper networks can be found in the supplementary materials.

In summary, the narrow spectrum of feature entropy and frequency observed in ViT suggests that it utilizes only redundant representational information for learning. Consequently, the ViT becomes excessively smoothed, leading to a redundant model. To ensure the diversity of representational information and mitigate redundancy, our proposed Channel Propagation method accumulates refreshable feature representation, allowing the network to learn richer feature representations. In the following section, we introduce the **Channel Propagation** method to prevent excessive smoothing and enhance representational capacity by refreshing the skip connection.

2.2. Channel Propagation

In this subsection, we introduce a Channel Propagation (CP) method that integrates layer-specific supervision features. The objective is to enhance the diversity of feature representation, so the proposed method is formulated using channel dimensions that progressively increase, as illustrated in Fig. 3. In our approach, the initial step involves extracting layer-specific features by a DW-Conv performed after reshaping to a 2D grid. In this operation, akin to positional embedding, we utilize the learnable parameters called Positional Bias term (Pos. Bias) to encode the embedding information regarding the token positions [16]. In addition, we exploit a single MLP layer to associate the channel features.

After extracting the layer-specific features, we concatenate them with the input features along the channel dimension. In other words, we combine both the preserved input and the newly added layer-specific features to refresh the features in each attention layer. The process of the proposed Channel Propagation method operates as follows.

$$\mathrm{X'}_l = \psi(\mathrm{X}_l) + \mathrm{MHSA}(\psi(\mathrm{X}_l)), \tag{4}$$

$$\psi(\mathrm{X}_l) = \mathit{concat}\left(\mathrm{RC}_l, \mathrm{X}_l\right), \tag{5}$$

where X' is the input feature and RC denotes the refreshable layer-specific feature, which is computed as:

$$\mathrm{RC'}_l = \mathrm{DWBlock}(\mathrm{X}_l) + \mathrm{P_b}, \tag{6}$$

$$\mathrm{RC}_l = \mathrm{MLP}(\mathrm{RC}'_l) + \mathrm{RC}'_l, \tag{7}$$

where $\mathrm{P_b}$ represents the learnable positional bias, and DWBlock is the depth-wise convolution layer. This layer-specific refreshable feature is propagated to the subsequent layer over a skip connection without over-smoothing, therefore maintaining the integrity of the original feature signals. As a result, the properties of the receptive field specific to each layer are acquired by leveraging the newly propagated refreshable channels. This facilitates the integration of the refreshable channels at each layer level to learn unique features from the different receptive fields of each layer.

Enhanced Feature Diversity. Fig. 2c and 2d show the CKA similarity of ViTs adopting CP at depths of 12 and 24, respectively. The baseline network [7] learns similar representations across all layers, whereas the proposed Channel Propagation reduces the similarity in representation across

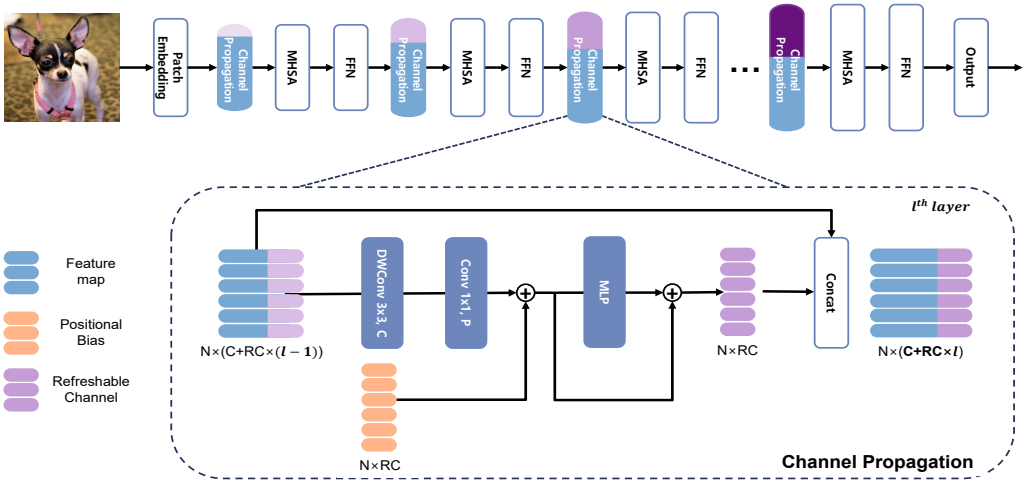

Figure 3. The workflow of the proposed Channel Propagation method. The process involves the aggregation of layer-specific feature representations immediately prior to the Multi-Head Self-Attention (MHSA) block. Subsequently, these features are propagated through MHSA operations and two skip connections. To clarify, our method retains and transfers the location-specific representation information that has been aggregated via skip connections to the subsequent layer.

layers by learning layer-specific features, enhancing diversity. Additionally, in Fig. 2e, the baseline network captures only a narrow range of spatial entropy, whereas the proposed Channel Propagation learns a broader range of spatial entropy. This observation indicates that Channel Propagation learns not only detailed and local information but also monotonic and global information simultaneously. Additionally, as shown in Fig. 2f, the proposed Channel Propagation can acquire diverse frequency spectra compared to the baseline. In particular, due to our architectural design, which preserves information from previous layers, the low-level frequency from MHSA progressively accumulates, leading to lower values in the early-stage layers. At the same time, as the convolutional channels in the proposed method increase in size, the model can capture high-frequency features, facilitating its ability to learn a wide spectrum of frequency levels.

Architectural Design Choice. We implement our proposed Channel Propagation to plain [25], hierarchical [14, 19] ViT architectures. In Tab 1, **Init dim.**, **Heads**, and **RC** (Refreshable Channel size) represent the embedding channel size, the number of heads, and the channel size that progressively increases by the proposed Channel Propagation method. To determine the progressive channel size, we follow the guidance from [23], which introduces an effective

Table 1. Architectural design of proposed Channel Propagation.

Networks	Init dim.	Heads	RC
ViT-S [25]	384	6	-
CP-ViT-T	128	4	8
CP-ViT-S	288	8	16
PiT-S [14]	144	{3, 6, 12}	-
CP-PiT-T	64	{2, 4, 8}	16
CP-PiT-S	144	{2, 4, 8}	32
Swin-S [19]	96	{3, 6, 12, 24}	-
CP-Swin-T	100	{4, 4, 5, 10}	30
CP-Swin-S	100	{4, 4, 10, 20}	20
CP-Swin-B	100	{5, 10, 15, 30}	30

setting of channel dimension as:

$$C = L/\rho, \tag{8}$$

where C is the channel dimension and L denotes the number of layers. In our implementation, we use the optimal setting of [23] as $\rho \in [0.1, 2.0]$, which is the user-configured hyperparameter for the favorable results.

In the case of hierarchical ViTs, we simply modify the downsampling layer to adjust the spatial resolution. This is because our method progressively grows the channel dimensions for every single attention block, so it is not necessary to increase the channel dimension in the downsampling layer. More details of our architectural designs can be found in the supplementary materials.

Table 2. Experimental study on the ImageNet-1K dataset. We compare the classification performance of plain, hierarchical, and deeper ViT models. The image size for training is 224×224. We use RTX-3090 for measuring the throughput.

Networks	#Param. (M)	FLOPs (G)	throughput (image / s)	Top-1 (%)	Top-5 (%)
Plain ViT					
ConViT-S [8]	6	1.3	2022	73.1	91.7
ConViT-S [8]	28	5.8	921	81.3	95.7
ConViT-B [8]	87	17.5	449	82.4	95.9
ViT-T [25]	6	1.3	2882	72.2	91.1
ViT-S [25]	22	4.6	1300	79.8	95.0
ViT-B [25]	86	17.5	604	81.7	95.6
CP-ViT-T	6	2.3	2157	77.8	93.9
CP-ViT-S	24	6.1	1148	82.5	95.9
ViT-S24	43	9.2	841	80.2	94.8
DeepViT-24B [41]	36	7.9	432	80.1	-
DiversePatch [10]	44	-	-	82.2	-
FeatScale [28]	43	9.1	651	81.3	-
CP-ViT-S24	46	9.7	656	81.9	95.7
Hierarchical ViT					
PVT-T [30]	13	1.9	1965	75.1	-
PVT-S [30]	25	3.8	1112	79.8	-
PVT-M [30]	44	6.7	769	81.2	-
PVT-L [30]	61	9.9	541	81.7	-
CVT-13 [32]	20	4.5	658	81.6	-
CVT-21 [32]	32	7.1	445	82.5	-
PiT-T [14]	5	0.7	3420	73.0	91.4
PiT-S [14]	24	2.9	1498	80.9	95.3
CP-PiT-T	5	2.0	1911	77.6	94.0
CP-PiT-S	21	4.5	1130	82.2	95.8
Swin-T [19]	28	4.5	951	81.3	95.5
Swin-S [19]	50	8.8	577	83.0	96.2
Swin-B [19]	88	15.5	412	83.5	96.5
CP-Swin-T	16	5.7	605	82.1	96.2
CP-Swin-S	43	10.1	520	83.8	96.6
CP-Swin-B	81	16.7	301	84.1	96.9

3. Experiments

3.1. Image Classification on ImageNet-1K

Experiment Details. We implement the proposed Channel Propagation method using PyTorch framework and Timm [31] library. For ViT [25] and PiT [14] models, we follow the training recipe of their original setup. The models are trained with a batch size of 1024 for 300 epochs, employing a weight decay of 0.05 and the AdamW optimizer. The initial learning rate is set to 1e-3, and we use a 5-epoch warm-up with a cosine learning rate scheduler. Additionally, data augmentation methods such as Mixup [37], Cutmix [36], Random erasing [39], and Rand augment [5] are applied in our training. To train variants of the Channel Propagation method on the Swin Transformer [19], we refer to the official repository's training recipe for the Swin Transformer [19][1]. A total of 300 epochs is used with a batch size of 1024. This training process utilizes the AdamW optimizer with a weight decay value of 0.05. We utilize cosine lr scheduler, starting from an initial value of 0.001.

[1]https://github.com/microsoft/Swin-Transformer

Table 3. Ablation studies on the component analysis for DW-Conv. We utilize the Swin-S [19] with 100 epoch training setup.

Method	#Param. (M)	Top-1 (%)
Baseline [19]	50	81.0
+ MLP	43	81.1
+ DWConv (k=3)	43	**81.7**
+ DWConv (k=5)	43	81.6

Classification Results. We validate the classification performance of the Channel Propagation method on the ImageNet-1K dataset. We verify its performance for various ViTs, such as plain, hierarchical, and deeper architectures. As shown in Tab. 2, the Channel Propagation exhibits minimal or even lower overhead in terms of parameters and FLOPs, surpassing the performance of other ViT networks. Specifically, our method enhances the classification performance of a plain ViTs, achieving 82.5% compared to the baseline's 79.8%. Ours improves the performance of the Swin Transformer [19], achieving 83.0%, an improvement of 0.8%, and the PiT [14] network's performance from 80.9% to 82.2%. Our method enhances the classification performance of Swin-B from 83.5% to 84.1%, with very low overhead in terms of computational budget. The efficacy of the CP persists even in the case of deeper ViT with 24 attention layers (CP-ViT-S24).

3.2. Ablation Studies

Ablation studies are performed on the Channel Propagation method utilizing the Swin-S [19] and ViT-S [25] models. The experiments conducted in these studies involve training on the ImageNet-1K dataset for 100 epochs.

Effective of Network Design. To verify the effectiveness of the proposed design solely that concatenates refreshable channels progressively, we only use a single MLP layer in our CP-MLP. As shown in Tab. 4, CP-MLP outperforms the baseline without any other components, such as convolution or positional encoding. Furthermore, as shown in Fig. 4, CP-MLP also learns diverse feature entropy that can reduce feature redundancy.

Component Analysis. The Channel Propagation generates layer-specific representations before the MHSA operation to propagate refreshable features in the skip-connection operation. We first investigate the components of the proposed method in Tab. 3. We evaluate DW-Conv operation in

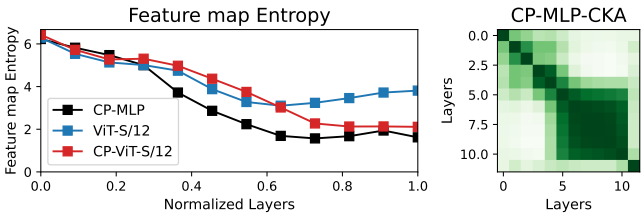

Figure 4. Analysis of **Left**) Entropy, **Right**) Layer-wise cosine similarity. Even with a simple strategy of ours (CP-MLP), it learns more diverse levels of feature map entropy.

Table 4. Ablation study on the effectiveness of our CP-MLP design. We use 100 epochs for training on the ImageNet-1k.

Methods	#Param (M)	FLOPs (G)	Top-1 Acc.
ViT-S	22.1	4.6	75.8
CP-MLP	23.4	4.8	**77.6**
CP-ViT-S	23.6	5.0	**78.9**

Table 5. Ablation studies on the components of Channel Propagation. We train all networks with 100 epochs training setup.

MLP	DWConv	Pos. Bias	#Param. (M)	Top-1 (%)
✗	✗	✗	22	75.8
✓	✗	✗	23	78.3
✓	✓	✗	23	78.7
✓	✓	✓	24	**78.9**

Eq. 6 to validate the efficacy of the layer-specific receptive fields. We replace the 3×3 DW-Conv by MLP and 5×5 DW-Conv. It is confirmed that the 3×3 DW-Conv shows the best performance improvement with minimal overhead. This result suggests that layer-specific features should learn the appropriate size of receptive fields.

We conduct further ablation studies on components of the Channel Propagation: DW-Conv, a single MLP layer, and positional bias. Tab. 5 shows the results of this ablation study. In our architectural design, each component outperforms its respective baseline.

Channel Propagation with original hyperparameters. Our networks progressively increase the channel size, so we need to adjust the channel dimension and the head numbers of the original ViT. In this respect, one can be concerned that the performance improvement comes from the adjustment of hyperparameters such as the channel dimension and head numbers. Therefore, to validate the performance of our Channel Propagation with the same hyperparameter, we insert the CP into a plain network identical to the original design. We maintain the same channel dimension by using a single linear embedding layer. We reduce the channel dimension after the CP block to be the same as the original design from the embedding layer. As shown in Tab. 6, even without using the progressive channel design utilizing the Channel Propagation, the proposed approach works favorably against the original baseline ViTs.

Table 6. Ablation studies on the original ViT [25] network architecture with our Channel Propagation method. The hyperparameters of the channel dimension and number of heads are the same as the original ViT with ours. We use 100 epochs for training all networks.

Networks	Dim.	# Heads	#Param. (M)	FLOPs (G)	Top-1 (%)
ViT-T [25]	192	3	6	1.3	67.6
CP-ViT-T	192	3	6	1.3	69.3
ViT-S [25]	384	6	22	4.6	75.8
CP-ViT-S	384	6	24	5.0	77.5

Table 7. Results of CP-blocks with comparable throughput. We redesign our Channel Propagation method to have a similar throughput compared to the baseline. The redesigned architecture for PiT and ViT is faster while reducing parameters and FLOPs than the baseline.

Networks	#Param. (M)	FLOPs (G)	Top-1 (%)	Throughput (image / s)
PiT-S [14]	24	2.9	80.9	1536
CP-PiT-S	16	2.9	81.4	1552
ViT-S [25]	22	4.6	79.8	1311
CP-ViT-S	16	4.4	81.6	1314

Comparable Resource. To verify the efficiency of the proposed approach, we redesign the proposed network by adjusting the channel dimensions and the size of the RC to improve throughput. As shown in Tab. 7, the redesigned CP-PiT-S and CP-ViT-S outperform PiT-S [14] and ViT-S [25] by 0.5% and 1.8%, respectively. Additionally, our Channel Propagation significantly reduces the parameters of the baseline network while achieving faster inference speeds, leading to overall performance improvements.

Refreshable Channel in FFN. We apply our Channel Propagation after MHSA operations, but there is another skip connection in FFN layers. Thus, we compare ours by applying it to both MHSA and FFN. In this ablation, we compare using 1) 16 dimensions of RC for only MHSA, 2) 8 dimensions for both MHSA and FFN, and 3) 16 dimensions of RC for only FFN. As shown in Tab. 8, applying our Channel Propagation to MHSA yields significantly better performance because the refreshable channels enhance the diversity of the attention maps in MHSA.

Attention map Visualization. To evaluate the effectiveness of refreshable channels, we visualize the attention

Table 8. Performance comparison of MHSA and FFN for our CP method. As outlined in Tab. 1, our method transmits a refreshable channel (RC) of size 16. For comparing MHSA and FFN, we evaluate the following configurations: 16 RC with MHSA, 8 RC with MHSA and FFN, and 16 RC with FFN.

Networks (RC)	#Param. (M)	FLOPs (G)	Top-1 (%)
ViT-S [25]	22	4.6	79.8
MHSA (16)	25	6.1	82.5
MHSA (8) & FFN (8)	24	6.0	82.2
FFN (16)	24	6.0	82.0

Table 9. Semantic segmentation results on the ADE20k. We use UperNet as the segmentation backbone network.

Methods	#Param. (M)	FLOPs (G)	mIoU (%)
Swin-T [19]	59.0	237	44.0
CP-Swin-T	45.1	243	44.9
Swin-S [19]	80.3	261	48.3
CP-Swin-S	73.3	268	48.6

maps across layers for both the baseline ViT and CP-ViT, each with 12 layers[2]. As illustrated in Fig. 5, the attention maps of the baseline model exhibit growing similarity from Block 8 to Block 12, suggesting a decrease in diversity as the depth increases. In contrast, the Channel Propagation method, which integrates refreshable channels alongside attention operations, generates more diverse attention maps, hence reducing redundancy in the network.

3.3. Object Detection

Experimental Settings. We validate the object detection performance of our Channel Propagation on the COCO dataset [18] using the MMDetection library [2]. Mask R-CNN [12] is used a head network with the training recipe of from the Swin Transformer [19]. Specifically, AdamW optimizer using a 0.001 initial learning rate is employed with $1\times$ and $3\times$ training schedules, training for a total of 12 and 36 epochs, respectively. Input image size for all detection networks is configured by 800×1333.

Results. We report the results in Tab. 10. There are performance improvements of 1.1 in AP^{bb} and 0.7 in AP^{mk} scores by our CP-Swin-T. These performance improvements are consistently shown in the $3\times$ scheduler setting.

3.4. Semantic Segmentation

Experimental Settings. We utilize UperNet [33] and MMSegmentation [4] to evaluate the semantic segmentation performance on the ADE20K dataset [40]. The training script follows the Swin Transformer. Specifically, we use 512×512 cropped images and Adam optimizer with a 6×10^{-5} initial learning rate of 0.01 weight decay.

Results. Tab. 9 presents the parameters and mIoU scores of CP and the baseline network [19] for the downstream task. There is a 0.9% performance improvement for Swin-T with our Channel Propagation. Ours also achieves 0.3% performance improvement at a larger scale.

4. Related Works

4.1. Vision Transformer

The transformer architecturehas proven to be highly effective not just in language data but also in visual recognition tasks. This is supported by various studies such as DeiT [25], CaiT [26], Swin [19], and T2T [35]. In contrast to traditional convolutional neural networks networks [13, 20, 38], which primarily focus on capturing local associations, ViTs adopt a different approach. They partition the input image into patches of fixed size and process them sequentially through MHSA and FFN blocks. This approach enables the effective aggregation of long-range dependencies in the image. Nevertheless, ViTs are subject to a notable limitation referred to as an over-smoothing problem, wherein propagated features show excessive similarity. Several studies [3, 10, 22, 28, 41] have addressed the problem of redundant feature representation being generated in ViTs due to the self-attention operation. This over-smoothing issue becomes notably more prominent when developing ViTs with deeper depth, presenting a substantial obstacle. As a result, several studies endeavors aimed at recalibrating the self-attention operation [3, 28, 41] and augmenting the variability of image patches [6, 10].

4.2. Skip Connection in Vision Transformer

Skip connections are widely used as essential elements in deep neural networks to facilitate stable learning [13, 24, 27, 29]. Skip connections allow the direct transmission of identity features of input signals to output signals, hence alleviating potential optimization challenges that may develop during very deep networks. Within the context of ViTs, skip connections play a comparable role by facilitating consistent learning and mitigating the degradation of high-frequency signals that arise from the self-attention mechanism's low-pass filter [28]. Upon careful examination of the occurrence of duplicate representations in ViTs, it becomes apparent that the traditional approach of incorporating residuals through skip connections does not play a significant role in mitigating the issue of excessive smoothing. We introduce a new residual scheme with the Channel Propagation to improve the impact of skip connections. This scheme serves the dual purpose of preserving high-frequency signals and alleviating over-smoothing.

[2]We use [41] for the attention map visualization.

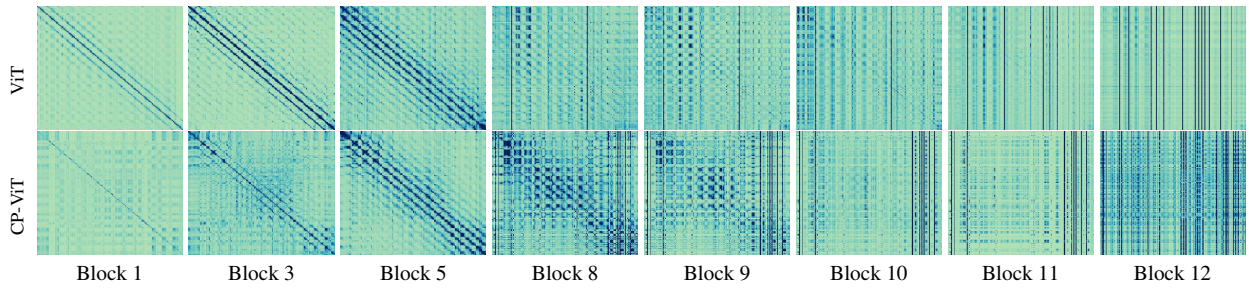

Figure 5. The layer-wise attention map visualization of the baseline and our Channel Propagation. The presence of redundant attention maps is apparent in the initial state, and this tendency grows more prominent, particularly at deeper levels. On the other hand, the proposed Channel Propagation shows more diverse patterns of attention maps.

Table 10. Experimental study on the object detection task. We compare the object detection performance of the proposed Channel Propagation with the Swin Transformer on the MS-COCO dataset.

Schedule	Backbone	#Params. (M)	FLOPs (G)	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
$\times 1$	Swin-T [19]	48	257	42.7	64.6	46.9	39.4	61.8	42.3
	CP-Swin-T	36	279	43.8	65.7	48.2	40.1	62.8	43.2
	Swin-S [19]	69	343	45.3	67.2	49.6	41.1	64.3	44.3
	CP-Swin-S	63	369	46.6	68.4	51.3	42.2	65.7	45.7
$\times 3$	Swin-T [19]	48	257	46.0	68.2	50.3	41.6	65.3	44.7
	CP-Swin-T	36	279	47.0	68.8	51.8	42.3	65.9	45.4
	Swin-S [19]	69	343	48.0	69.5	52.8	42.9	66.9	46.5
	CP-Swin-S	63	369	49.0	70.3	54.0	43.5	67.6	47.0

4.3. Feature Channels in Network

The dimensionality of the feature map plays a critical role in constructing robust deep neural networks. It is evident that effectively modeling the distinct characteristics of each channel dimension enhances the network's capacity to learn diverse feature representations. Previous studies have placed significant emphasis on the efficient configuration of channel dimensions and improving network efficiency by implementing carefully crafted channel designs. The recalibration of channel features was performed by Hu *et al*. [15] through the learning of inter-channel relationships. Jun *et al*. [9] introduced a lightweight spatial bias term that is concatenated in channel dimensions as a potential substitute for self-attention. PyramidNet [11] addressed the issue of information loss in downsampling layers by implementing gradual expansion of the channel dimensions. PiT [14] was able to establish a balance between lightweight design and high performance by emphasizing the utilization of hierarchical channel dimension reduction methods within CNNs. DeepMAD [23] optimizes structural parameters such as the depth and width of the network based on information entropy, achieving state-of-the-art performance with minimal computational overhead.

5. Conclusion

In this paper, we propose a novel Channel Propagation method to address the issue of feature redundancy in ViTs by introducing layer-specific features. Our analysis reveals the presence of redundant feature representations within ViTs, and to alleviate this, we incorporate layer-specific feature representations to refresh the skip connections within each attention block. To validate the effectiveness of our approach, we measure the classification performance for both plain and hierarchical ViT architectures, demonstrating that our design brings substantial performance improvements with low computational overhead. Furthermore, from various analyses of the proposed method, we confirm that our method alleviates over-smoothing to ensure feature diversity in a network. Our proposed method exhibits significant performance improvement in image classification and its downstream tasks. It is posited that the proposed approach holds significant potential in facilitating the building of deep neural networks.

Acknowledgment. This paper was supported in part by the Electronics and Telecommunications Research Institute (ETRI) Grant funded by Korean Government (Fundamental Technology Research for Human-Centric Autonomous Intelligent Systems) under Grant 24ZB1200, under the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2024-RS-2023-00255968), Artificial Intelligence Innovation Hub under Grant RS-2021-II212068, NRF from the Korea Government (MSIT) under Grant RS-2024-00356486.

References

[1] Turgay Celik. Spatial entropy-based global and local image contrast enhancement. *IEEE Transactions on Image Processing*, 23(12):5298–5308, 2014. 3

[2] 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. 7

[3] Tianlong Chen, Zhenyu Zhang, Yu Cheng, Ahmed Awadallah, and Zhangyang Wang. The principle of diversity: Training stronger vision transformers calls for reducing all levels of redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12020–12030, 2022. 2, 7

[4] MMSegmentation Contributors. Mmsegmentation: Openmmlab semantic segmentation toolbox and benchmark, 2020. 7

[5] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 5

[6] Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, pages 2793–2803. PMLR, 2021. 2, 7

[7] 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. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2, 3

[8] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning*, pages 2286–2296. PMLR, 2021. 5

[9] Junhyung Go and Jonngbin Ryu. Spatial bias for attention-free non-local neural networks. *Expert Systems with Applications*, page 122053, 2023. 8

[10] Chengyue Gong, Dilin Wang, Meng Li, Vikas Chandra, and Qiang Liu. Vision transformers with patch diversification. *arXiv preprint arXiv:2104.12753*, 2021. 1, 5, 7

[11] Dongyoon Han, Jiwhan Kim, and Junmo Kim. Deep pyramidal residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5927–5935, 2017. 2, 8

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 7

[13] 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*, pages 770–778, 2016. 7

[14] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11936–11945, 2021. 1, 2, 4, 5, 6, 8

[15] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 8

[16] Md Amirul Islam, Sen Jia, and Neil DB Bruce. How much position information do convolutional neural networks encode? *arXiv preprint arXiv:2001.08248*, 2020. 3

[17] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019. 2, 3

[18] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 7

[19] 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*, pages 10012–10022, 2021. 1, 2, 4, 5, 7, 8

[20] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 7

[21] Namuk Park and Songkuk Kim. How do vision transformers work? *arXiv preprint arXiv:2202.06709*, 2022. 3

[22] 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*, 34:12116–12128, 2021. 1, 2, 7

[23] Xuan Shen, Yaohua Wang, Ming Lin, Yilun Huang, Hao Tang, Xiuyu Sun, and Yanzhi Wang. Deepmad: Mathematical architecture design for deep convolutional neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6173, 2023. 4, 8

[24] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence*, 2017. 7

[25] 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*, pages 10347–10357. PMLR, 2021. 2, 4, 5, 6, 7

[26] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 32–42, 2021. 7

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 7

[28] Peihao Wang, Wenqing Zheng, Tianlong Chen, and Zhangyang Wang. Anti-oversmoothing in deep vision transformers via the fourier domain analysis: From theory to practice. *arXiv preprint arXiv:2203.05962*, 2022. 1, 3, 5, 7

[29] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao. Learning deep transformer models for machine translation. *arXiv preprint arXiv:1906.01787*, 2019. 7

[30] 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*, pages 568–578, 2021. 5

[31] Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021. 5

[32] 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*, pages 22–31, 2021. 5

[33] 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)*, pages 418–434, 2018. 7

[34] Fuzhao Xue, Jianghai Chen, Aixin Sun, Xiaozhe Ren, Zangwei Zheng, Xiaoxin He, Xin Jiang, and Yang You. Deeper vs wider: A revisit of transformer configuration. *arXiv preprint arXiv:2205.10505*, 2022. 2

[35] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 558–567, 2021. 7

[36] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 5

[37] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 5

[38] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2736–2746, 2022. 7

[39] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 13001–13008, 2020. 5

[40] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. 7

[41] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 1, 5, 7