Introducing Feedback Connections for Vision Transformers

Vatsal Agarwal
University of Maryland, College Park
vatsalag@terpmail.umd.edu

Abstract

The introduction of Transformer networks in computer vision has resulted in rapid progress of deep models in a variety of vision tasks. These performance gains are strongly tied to the core component of Transformers, namely self-attention that enables such models to capture important long-range spatial interactions. While several methods have been developed to improve the efficiency or modeling capacity of Vision Transformers, most of them rely on a feedforward architecture where lower-level features are sequentially processed to form higher-level features that contain semantically-rich information. Such a design is prohibiting and inefficient as the network is unable to use the already learned high-level features to better extract relevant information from the lower-level features. The human visual system (HVS) in contrast relies on a series of feedforward and feedback connections to efficiently process visual stimuli. In this work, we propose a Adaptive Gated Attention Block which adds lightweight top-down and bottom-up connections to enhance information flow between features at different levels. In addition, our module is flexible and can be integrated with both local and global-attention based methods. Experiments on benchmark datasets demonstrate that our Adaptive Gated Attention Block consistently outperforms existing state-of-the-art with a negligible increase in parameters and FLOPs.

1. Introduction

The inception of Transformers [40] has revolutionized the field of deep learning for a variety of problems ranging from natural language processing (NLP) to computer vision. While much focus has been placed on using Transformers for NLP tasks, there has been a recent widespread effort to transfer these advances to the vision field [3–5, 9, 13, 16, 20, 39, 41–45, 49]. The first major breakthrough in this effort came from Vision Transformer (ViT), which demonstrated incredible results in image-classifications tasks using a purely Transformer architecture [8]. Inspired by this, many papers proposed different Transformer-based architectures and applied them to a diverse set of vision tasks including fine-grained classification, object detection, and semantic segmentation.

While these approaches have focused on improving the efficiency and modeling capacity of the core self-attention mechanism, they still mostly utilize a bottom-up architecture. This design limits information processing to one direction where the model learns to aggregate lower-level features to form semantically-rich high-level features. In contrast, the human visual system (HVS) utilizes a combination of feed-forward, feedback, and lateral connections to process information [18]. The feedback connections are crucial for efficient processing as they allow the HVS to contextualize raw sensory inputs and better focus on objects of interest [22, 47]. Such top-down processing is critical for scenes that contain occlusions or many distracting objects [11, 12, 27]. In these settings, the HVS can employ high-level information to adaptively select and combine relevant features from both the top-down and bottom-up signals [6, 15, 22, 23].

There has been considerable effort to translate these capabilities to deep vision models [1, 2, 10, 28, 29, 32–34, 38]. A common strategy that has been used is separating the propagation of bottom-up and top-down signals into two separate networks which are then trained iteratively [10, 32]. These have then been extended to include lateral connections between the bottom-up and top-down networks oftentimes by concatenating features from both layers and processing them with a 1×1 convolution [29, 34]. More recently, in [28], a recursive top-down module is introduced to propagate multi-scale and top-down feature information across the network.

In order to address this limitation, we introduce a lightweight module to add top-down and bottom-up connections. The core component of this module is our **Adaptive Gated Attention Block** which is responsible for the generation and propagation of top-down information. First, top-down attention maps are generated from the low-level and high-level features. These attention maps are then used to modulate the low-level features via a gating mechanism. Finally, we feed the gated low-level features to a Transformer

block to obtain refined features at each scale. These features are then processed for downstream tasks. This module is then easily extended to incorporate bottom-up connections by simply adding the gated low-level features to their corresponding high-level features.

Our Adaptive Gated Attention Block is easily integrable to different backbone Vision Transformer architectures. In this work, we apply our module to the PvT-v2 backbone [42]. Furthermore, we demonstrate the efficacy of our proposed module for the image classification task using the ImageNet dataset [7]. We find that adding our Adaptive Gated Attention Block boosts classification performance while adding a negligible number of parameters and FLOPs.

2. Related works

2.1. Vision Transformers

Convolutional neural networks have been the de-facto method for approaching general vision tasks [14, 19, 21, 30, 31, 35]. The development of ViT [8] though has showcased the potential for Transformer backbones to further improve performance. The seminal work proposed splitting an image into fixed-length patches and feeding them to a sequence of Transformer layers and demonstrated the efficacy of a pure Transformer architecture for vision tasks. One caveat, however, is that ViT requires much more training data compared to CNNs.

Since then, there have been several methods that have been proposed to improve the Vision Transformer design. Many of these works focus on integrating a hierarchical structure into the Vision Transformer architecture [9,24,41]. Another avenue of work has been centered on applying the inductive biases of CNNs to Transformers [5, 24, 39, 43, 44, 49]. In an orthogonal direction to these works, there has been a considerable effort to develop more efficient Transformer blocks by addressing the quadratic complexity of self-attention. Specifically, works such as ReST [49], PVT [41, 42] and MViT [9] utilize global self-attention, while [13, 24] apply local self-attention. [4] aims to combine both global and local attention for improved context modeling.

Our work can also be considered as a method for improving hierarchical and multi-scale processing. Swin, PVT, and MViT [9, 24, 41] are some of the first works to propose breaking up the stack of Transformer layers into stages, where at each stage the spatial and channel dimensions are reduced and expanded respectively. Cross-ViT [3] extends this idea further by enabling feature-level interactions across different image scales. It does so by employing two parallel streams for processing smaller and larger image tokens respectively and then using a cross-attention layer to fuse information from both scales.

The Adaptive Gating Attention Block is inspired by CoaT [44]. Rather than using parallel streams, CoaT proposes a co-scale mechanism that allows for feature interactions across different scales. This allows for both local-global and global-local modeling. This mechanism is implemented via a Parallel Block that effectively adds features between each stage and then processes them with a Transformer layer. While this module provides the model with both lateral and top-down connections, it is limited in its ability to selectively modulate features at each scale. A concurrent work [20] similarly works to integrate features across each stage via concatenation and thereby utilizes both local and global features for the final classification prediction. Our Adaptive Gating Attention Block differs from these two works by instead using a gating mechanism to fuse and modulate features across scales rather than addition [44] and concatenation [20].

2.2. Top-Down Attention

Our work is part of a long line of research aiming to introduce feedback connections in deep models. [2] is one of the first works applying feedback connections to CNNs. It realizes this via an iterative process where the model alternates between using its feedforward and feedback connections. While the feedforward layers compute the predictions, the feedback layers route the high-level information to the lower-level features. The low-level neurons that are directly relevant to the target predictions are more highly activated. In a similar vein, [29] utilizes top-down connections to refine initial predictions made from the bottom-up network. [34] further extended the concept of top-down modulation for the object detection task by introducing a lateral and top-down network to generate ROI proposals. Top-down feedback has been utilized for other vision tasks as well including crowd-counting [32] and visual question answering [1]. More recently, [28] integrates bottom-up and top-down interactions across multiple scales via a recursive module that takes in inputs at different image scales and propagates attention features across the network stages.

Our gating mechanism is inspired by [38]. In this work, a two-stream network is proposed for boundary detection and semantic segmentation. The first stream produces content features and the second stream produces shape features. The low-level shape features are then gated by the high-level content features using the introduced Gated Convolutional Layer (GCL). Additionally, the integration strategy of our gating mechanism resembles [17]. This work proposes using propagator and modulator gates to generate and implement top-down attention along each stage of the bottom-up network for segmentation task. There are key differences between these works and our gating mechanism. First, rather than simply using spatial attention, we also introduce a channel attention mechanism to gate specific low-

level information such as textures and edges. Next, our block can be integrated without introducing extra recurrent connections which greatly increase FLOPS.

3. Method

3.1. Self-Attention Revisited

The core component of the Transformer is self-attention [40]. First an input sequence of features $X \in \mathbb{R}^{N \times d}$ is projected into query, key, and value embeddings. These vectors can be represented as XW_q, XW_k, and $XW_v \in \mathbb{R}^{Nxd}$ where W_q, W_k, and $W_v \in \mathbb{R}^{d \times d}$ are projection matrices. [40] then formulates self-attention as:

$$A(X) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{1}$$

Intuitively, the attention mechanism calculates the similarity between query features and key features and then reweighs the value features accordingly. The queries can thus be considered to capture what is needed to be compared, the keys contain the information to be matched against and the values represent the information to be selected and propagated downstream.

3.2. Adaptive Gated Attention Block

The Adaptive Gated Attention Block performs two key tasks. First, it modulates low- and high-level features via top-down gating and bottom-up residual connections respectively. Second, it feeds the refined features to a Transformer layer to further mine contextual information. The overall architecture is shown in Fig. 1.

3.2.1 Top-Down Gating Module

We first discuss the design of our top-down gating module, which is inspired by [38]. Given a pair of low- and high-resolution features (X_l, X_h), where $X_l \in \mathbb{R}^{C_l \times H_l \times W_l}$ and $X_h \in \mathbb{R}^{C_h \times H_h \times W_h}$, we first obtain spatial and channel attention maps for the low-level features. To compute the spatial attention, we generate a compact representation of the high-level features via a 1×1 convolution to obtain $X'_h \in \mathbb{R}^{1 \times H_h \times W_h}$; we then upsample the feature map and concatenate it with the low-level features and process it with another 1×1 convolution to generate $A_{spa} \in \mathbb{R}^{1 \times H_l \times W_l}$. We then normalize this with the tanh operation to obtain the final attention map. The formal equation is as follows:

$$A_{spa} = \tanh(C_{1\times1}(X_l || X'_h)) \tag{2}$$

To generate the channel attention map, we concatenate the low-level features and high-level features and apply average-pooling to aggregate information in the spatial dimension. We then feed this fused feature to a 1×1 convolution to obtain $A_{cha} \in \mathbb{R}^{C_l \times 1 \times 1}$, which we then normalize as well with tanh function. This is shown below:

$$A_{cha} = \tanh(C_{1\times1}(\text{AvgPool2d}(X_l || X_h))) \tag{3}$$

We obtain the new low-level features by applying each attention map independently with a residual connection and then fusing both via addition. To refine the high-level features, we can simply down-sample the processed low-level features such that the spatial dimensions match and apply a 1×1 convolution to match the channel dimensions. We then add them to the original high-level features. These can be considered as bottom-up connections from low-to-high. Both are shown below:

$$X_l = (1 + A_{spa}) * X_l + (1 + A_{cha}) * X_l \tag{4}$$

$$X_h = X_h + C_{1\times1}(X_l) \tag{5}$$

Here we explain the intuition behind this design. In the top-down gating module, we fuse low-level and high-level information to obtain both spatial and channel attention maps. The spatial attention guides the low-level features to focus on important regions in the scene while the channel attention aids the low-level features to focus on specific low-level attributes (e.g. texture and edge patterns) that greater correspond with high-level semantic information. Finally, the addition of the refined low-level features to the high-level information enables the high-level features to contain more fine-grained details and other structural information.

3.2.2 Cross-QKV Attention Layer

In order to maximize the effectiveness of our top-down gating module, we carefully design the structure of the adaptive gated attention block. The key intuitions behind our approach is that the queries, keys, and values capture diverse contextual information at each attention layer and more specifically that the queries, keys, and values at higher layers can guide the generation of those at lower layers. We discuss the details of this block below.

Given a pair of intermediate features extracted at each stage of the bottom-up network, (X_i, X_j), we compute query, key, and value features with the standard linear embedding function: (Q_i, K_i, V_i) and (Q_j, K_j, V_j). Deciding how to pair the low-level and high-level features is non-trivial. It is fairly straightforward to gate the low-level keys and values with the corresponding high-level features. However, it may not be effective to apply such an approach for the query features since the query features at each layer may not necessarily focus on relevant information. Instead, we utilize the high-level values to guide lower-level queries. Intuitively, this can be seen as using what the network has

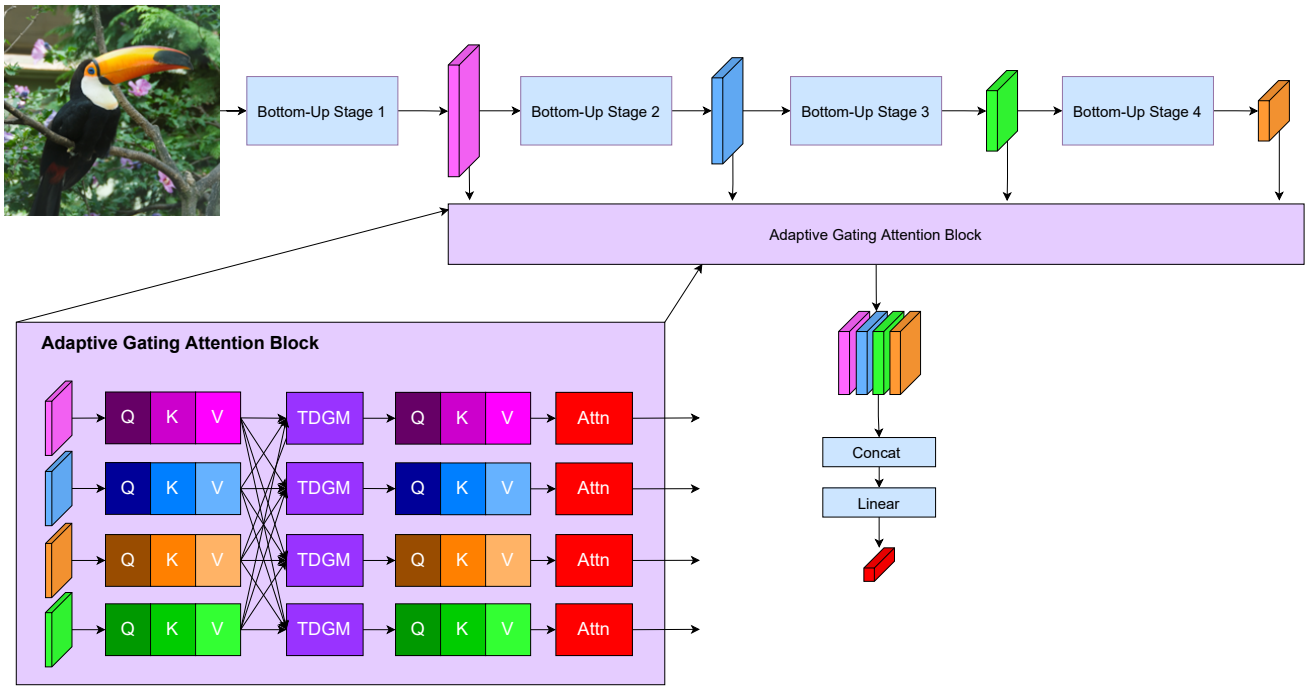

Figure 1. **Adaptive Gating Attention Block Architecture.** For any given backbone architecture, we extract intermediate features at each scale and feed them to the adaptive gating attention block to propagate top-down and bottom-up connections.

already selected to be most relevant to constrain the query features. An important caveat is that while there are top-down and bottom-up interactions for the key and value features, we only apply top-down connections for the query features. Furthermore, we order the gating such that the values are refined before they are used to modulate the low-level queries. For a four-stage network, we apply this gating mechanism across each pair of intermediate features and then add the refined features for each pair. This is shown below (TDGM is the top-down gating module):

$$Q_i = \sum_{j>i}^{n\_stages} TDGM(Q_i, V_j) \tag{6}$$

$$K_i = \sum_{j>i}^{n\_stages} TDGM(K_i, K_j) \tag{7}$$

$$V_i = \sum_{j>i}^{n\_stages} TDGM(V_i, V_j) \tag{8}$$

The modulated query, key, and value features are then fed into a Transformer layer to extract more relevant contextual information. Specifically, we utilize the spatial-reduction attention layer utilized in the PvT-v2 backbone [42] with one modification. This module reduces the computational burden of self-attention by down-sampling the features to a fixed size prior to computing the keys and values. Such a design however would impede the effectiveness of our gating mechanism as there would only be a single-scale representation of the key and value features. Therefore, we simply switch the order of the down-sampling and key/value embedding, so that we first compute the key and value features first. We next apply the top-down gating module as described earlier and then perform the down-sampling.

Our Cross-QKV Attention Layer shares similarities to a variant of the Parallel Block [44] (direct cross-layer attention) with several important distinctions. While both approaches rely on the expressivity of the query, key, and value features, our proposed module utilizes gating to fuse the corresponding features across different scales rather than using the corresponding features directly. Additionally, we explicitly use high-level features to gate the query features at each scale to ensure that the network focuses on relevant regions and low-level semantics at each scale.

3.3. Model Architecture

We demonstrate the effectiveness of our Adaptive Gating Attention Block by integrating it with the PvT-v2 backbone. Please refer to [41, 42] for more details regarding the backbone architecture. For consistency, we make the same modification detailed earlier to the backbone and compute the key/value embedding prior to down-sampling. Prelimi-

Method	#Param (M)	GFLOPs	Top-1 Acc (%)
PVTv2-B0 [42]	**3.7**	**0.6**	71.7
PVTv2-B0-Gated-Only (ours)	4.1	**0.6**	72.2
PVTv2-B0-Gated-Light (ours)	5.6	0.8	**74.3**
PVTv2-B0-Gated (ours)	6.0	1.2	**74.3**
ResNet18 [14]	11.7	1.8	69.8
DeiT-Tiny/16 [39]	**5.7**	**1.3**	72.2
PVTv1-Tiny [41]	13.2	1.9	75.1
PVTv2-B1 [42]	13.1	2.1	**78.7**

Table 1. **Image Classification on ImageNet Validation Set**. Our method shows competitive performance compared to the PvT-v2 model without significantly increasing the number of parameters and FLOPs.

nary experiments show that this doesn't significantly impact performance.

3.3.1 Model variants

In order to better understand the key components of our Adaptive Gating Block, we design two variant architectures. The first design consists of only the top-down gating module and applies it to the intermediate features directly as modulation rather than the query/key/value features. Furthermore, it does not rely on an additional Transformer layer for processing. We call this variant **Gated-Only**. With our second variant, we aim to identify which top-down connections are most crucial. Thus, we limit the propagation of top-down connections to only occur between the query and value features from the second, third, and fourth stages. We call this variant **Gated-Light**. We choose to only focus on gating the query features since these would likely have the most variance across different layers. Thus, gating the query features with the values would help ensure that the network only queries the relevant spatial and feature information. We examine the effectiveness of our model and these variants in the next section.

4. Experiments

4.1. Experimental Setup

Image Classification: We use ImageNet [7] for our experiments. This benchmark consists of 1.3M images for training and 50K images for validation spanning 1000 object classes. We follow the same training paradigm as [42]. Specifically, we apply the following data augmentations: random erasing [50], random horizontal flipping [36], random cropping, label-smoothing [37], Cutmix [46], and mixup [48]. We use the AdamW [26] optimizer and set the momentum to 0.9, the batch-size to 128, and weight-decay to 5×10^{-2}. We use an initial learning rate of 1×10^{-3} and use cosine scheduling [25]. We train all models using $4\times$ V100 GPUs for 300 epochs.

4.2. Experimental Results

Our image classification results are displayed in Tab. 1. We compare our model with the baseline PvT-v2 model for the B0 scale (refer to [42] for more details). We use the publicly available repository and train the baseline models as well. We find that the addition of one gating block improves accuracy by **2.6**% with a negligible increase in parameters and FLOPs. Surprisingly, we also see that we can retain this performance when we constrain our Top-Down Gating Module to only focus on the query features for certain scales. Furthermore, if we completely remove the Cross-QKV Attention Layer and use the Top-Down Gating Module directly, the performance still is superior to that of the baseline with less than 1M parameters added.

4.2.1 Top-Down Gating Module Comparison

In order to investigate the efficacy of our Top-Down Gating Mechanism, we compare its performance to several competitive baselines. The results are shown in Tab. 2. We first examine the potential improvement of just adding an extra Transformer layer for each scale's intermediate features. Next, we integrate the proposed Parallel Block from [44] and examine the gains in accuracy. We first confirm that the inclusion of the gating mechanism is crucial for performance as the one Gated-Only block outperforms even two Parallel Attention blocks. When comparing to the Parallel Block, we see that while our models outperform a single Parallel Block, they are not as competitive against two Parallel Blocks.

Table 2. **Effectiveness of Top-Down Gating Module.** All experiments are performed with the PvT-v2-B0 architecture using the ImageNet-1K validation set.

Model	#Params	Input	#GFLOPs	Top-1 Acc.
PvT-v2 (base)	3.7M	224^2	0.6	71.7%
PvT-v2 w/ Attention Block				
- Parallel Attention	5.4M	224^2	0.6	72.0%
- Parallel Attention (2)	6.8M	224^2	1.1	72.0%
- Parallel Block	5.4M	224^2	0.6	71.5%
- Parallel Block (2)	7.0M	224^2	1.1	**75.8**%
- Gated-Only	**4.1M**	224^2	0.6	72.2%
- Gated-Light	5.6M	224^2	0.8	74.3%
- Gated	6.0M	224^2	1.2	74.3%

5. Conclusions and Future Work

In this work, we take inspiration from the human visual system to design a general framework for integrating top-down connections in Vision Transformers. Namely, we introduce the Adaptive Gated Attention Block with a Top-Down Gating Module and a Cross-QKV Attention layer. We apply this framework for the PvT-v2 backbone and

showcase promising results on the competitive ImageNet benchmark. Future work would aim to apply this block to larger models and other Vision Transformer backbones to demonstrate its general effectiveness. Additionally, more experiments need to be done to determine whether our module's performance gains can be realized on popular downstream tasks such as object detection and semantic segmentation.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018. 1, 2

[2] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2956–2964, 2015. 1, 2

[3] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 357–366, 2021. 1, 2

[4] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2

[5] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021. 1, 2

[6] Marvin M Chun and Yuhong Jiang. Top-down attentional guidance based on implicit learning of visual covariation. *Psychological Science*, 10(4):360–365, 1999. 1

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 5

[8] 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

[9] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6824–6835, 2021. 1, 2

[10] Carlo Gatta, Adriana Romero, and Joost van de Veijer. Unrolling loopy top-down semantic feedback in convolutional deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 498–505, 2014. 1

[11] Adam Gazzaley and Anna C Nobre. Top-down modulation: bridging selective attention and working memory. *Trends in cognitive sciences*, 16(2):129–135, 2012. 1

[12] Charles D Gilbert and Mariano Sigman. Brain states: top-down influences in sensory processing. *Neuron*, 54(5):677–696, 2007. 1

[13] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2

[14] 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. 2, 5

[15] Joseph B Hopfinger, Michael H Buonocore, and George R Mangun. The neural mechanisms of top-down attentional control. *Nature neuroscience*, 3(3):284–291, 2000. 1

[16] Zilong Huang, Youcheng Ben, Guozhong Luo, Pei Cheng, Gang Yu, and Bin Fu. Shuffle transformer: Rethinking spatial shuffle for vision transformer. *arXiv preprint arXiv:2106.03650*, 2021. 1

[17] Rezaul Karim, Md Amirul Islam, and Neil DB Bruce. Distributed iterative gating networks for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2844–2853, 2020. 2

[18] Victor AF Lamme, Hans Super, and Henk Spekreijse. Feedforward, horizontal, and feedback processing in the visual cortex. *Current opinion in neurobiology*, 8(4):529–535, 1998. 1

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[20] Nannan Li, Yaran Chen, Weifan Li, Zixiang Ding, and Dongbin Zhao. Bvit: Broad attention based vision transformer. *arXiv preprint arXiv:2202.06268*, 2022. 1, 2

[21] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[22] Grace W Lindsay. Attention in psychology, neuroscience, and machine learning. *Frontiers in computational neuroscience*, page 29, 2020. 1

[23] Grace W Lindsay, Daniel B Rubin, and Kenneth D Miller. A simple circuit model of visual cortex explains neural and behavioral aspects of attention. *bioRxiv*, page 875534, 2019. 1

[24] 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. 2

[25] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[27] Behrad Noudoost, Mindy H Chang, Nicholas A Steinmetz, and Tirin Moore. Top-down control of visual attention. *Current opinion in neurobiology*, 20(2):183–190, 2010. 1

[28] Bo Pang, Yizhuo Li, Jiefeng Li, Muchen Li, Hanwen Cao, and Cewu Lu. Tdaf: Top-down attention framework for vision tasks. In *AAAI*, 2021. 1, 2

[29] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *European conference on computer vision*, pages 75–91. Springer, 2016. 1, 2

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[32] Deepak Babu Sam and R Venkatesh Babu. Top-down feedback for crowd counting convolutional neural network. In *Thirty-second AAAI conference on artificial intelligence*, 2018. 1, 2

[33] Abhinav Shrivastava and Abhinav Gupta. Contextual priming and feedback for faster r-cnn. In *European conference on computer vision*, pages 330–348. Springer, 2016. 1

[34] Abhinav Shrivastava, Rahul Sukthankar, Jitendra Malik, and Abhinav Gupta. Beyond skip connections: Top-down modulation for object detection. *arXiv preprint arXiv:1612.06851*, 2016. 1, 2

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[36] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 5

[37] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 5

[38] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5229–5238, 2019. 1, 2, 3

[39] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers amp; distillation through attention. In *International Conference on Machine Learning*, volume 139, pages 10347–10357, July 2021. 1, 2, 5

[40] 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. 1, 3

[41] 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. 1, 2, 4, 5

[42] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, pages 1–10, 2022. 1, 2, 4, 5

[43] 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. 1, 2

[44] Weijian Xu, Yifan Xu, Tyler Chang, and Zhuowen Tu. Co-scale conv-attentional image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9981–9990, 2021. 1, 2, 4, 5

[45] 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. 1

[46] 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

[47] Theodore P Zanto, Michael T Rubens, Jacob Bollinger, and Adam Gazzaley. Top-down modulation of visual feature processing: the role of the inferior frontal junction. *Neuroimage*, 53(2):736–745, 2010. 1

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

[49] Qinglong Zhang and Yu-Bin Yang. Rest: An efficient transformer for visual recognition. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2

[50] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020. 5