

Understanding Gaussian Attention Bias of Vision Transformers Using Effective Receptive Fields

Bum Jun Kim
kmbmjn@postech.ac.kr

Hyecheon Choi
hyeyeon@postech.ac.kr

Hyeonah Jang
hajang@postech.ac.kr

Sang Woo Kim
swkim@postech.ac.kr

Department of Electrical Engineering
POSTECH
Pohang, South Korea

Abstract

Vision transformers (ViTs) that model an image as a sequence of partitioned patches have shown notable performance in diverse vision tasks. Because partitioning patches eliminates the image structure, to reflect the order of patches, ViTs utilize an explicit component called positional embedding. However, we claim that the use of positional embedding does not simply guarantee the order-awareness of ViT. To support this claim, we analyze the actual behavior of ViTs using an effective receptive field. We demonstrate that during training, ViT acquires an understanding of patch order from the positional embedding that is trained to be a specific pattern. Based on this observation, we propose explicitly adding a Gaussian attention bias that guides the positional embedding to have the corresponding pattern from the beginning of training. We evaluated the influence of Gaussian attention bias on the performance of ViTs in several image classification, object detection, and semantic segmentation experiments. The results showed that proposed method not only facilitates ViTs to understand images but also boosts their performance on various datasets, including ImageNet, COCO 2017, and ADE20K.

1 Introduction

Vision transformers (ViTs) [1] have achieved remarkable performances in various vision tasks that are often superior to those of convolutional neural networks (CNNs) [2, 3, 4]. Unlike CNNs, ViTs partition an image into a sequence of patches and subsequently combine patch features based on the self-attention (SA) mechanism, enabling the aggregation of rich global information within the image [5, 6, 7].

Despite its effectiveness, SA poses inherent limitations in understanding the order of input patches. However, because 2D images are structured data, understanding the order of patches is important for ViT [8]. To overcome this problem, ViTs employ an explicit

component called positional embedding that enables the identification of the order and the corresponding geometric positions of patches.

However, we claim that simply using positional embeddings does not ensure order-awareness. To validate our claim, we utilize the effective receptive field (ERF) [0, 23] that highlights the pixels actually used in perception, illustrating how ViTs understand images. Using ERF, we demonstrate that ViTs with untrained positional embeddings do not discriminate between near and far patches and that order-awareness is acquired after positional embedding is trained to obtain specific patterns.

Motivated by this observation, to construct a ViT born with the spatial understanding of images, we propose injecting Gaussian attention bias into positional embedding. The innate spatial understanding of images helps ViT capture the nearness and farness of pixels, thereby enhancing the performance of ViT in vision tasks. We observed that using Gaussian attention bias improved the performance of ViTs on several datasets, tasks, and models.

2 Background

Forward propagation of standard ViT Let $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ be the input image to the ViT, where $H \times W$ is the resolution, and C is the number of channels of the image. Using a pre-defined resolution of patch $P \times P$, ViT spatially partitions the image into N nonoverlapping patches, where $N = HW/P^2$. Each patch is linearly projected, yielding $\text{PatchEmbedding}(\mathbf{x})$.¹ Subsequently, an absolute positional embedding \mathbf{E}_{pos} is added, resulting in \mathbf{z}_0 , the input to the first transformer block. Now, transformer blocks containing SAs and multilayer perceptrons (MLPs) are applied in a row to produce \mathbf{z}_L , where L is the number of transformer blocks. Finally, LayerNorm [4] is applied to produce the last feature map \mathbf{y} from which the head produces a classification score. Here, $\text{PatchEmbedding}(\mathbf{x})$, \mathbf{E}_{pos} , \mathbf{z}_L , and \mathbf{y} have the same size of $\mathbb{R}^{N \times D}$, where D is the dimension of the patch features.

Trick to obtain the ERF of ViT Because the ERF of ViT has been rarely discussed and our study is the first to provide a concrete and detailed analysis on this topic, we first formulate the ERF of ViT. The ERF depicts the actual usage of each pixel for determining the target feature in a neural network, representing a generic connection between them. We follow the common tricks to obtain ERFs of CNNs [46]. However, unlike CNNs, to obtain the ERF of ViT, we should focus on the patch unit. To investigate the properties of the entire ViT, the last feature map \mathbf{y} is chosen as the target feature map. First, we target the n th patch corresponding to the central patch. Because our goal is to analyze the spatial relationship between the target patch and pixel units, we ignore other units such as the image channel and the dimension of the patch feature. Thus, the features of the central patch are averaged over its dimensions: $Y = \frac{1}{D} \sum_{d=1}^D \mathbf{y}_{n,d}$. Now, we examine the contribution of each pixel to Y that can be obtained by a gradient $\frac{\partial Y}{\partial \mathbf{x}} \in \mathbb{R}^{H \times W \times C}$. The gradient is averaged over channels to obtain $\mathbf{G} = \frac{1}{C} \sum_{c=1}^C [\frac{\partial Y}{\partial \mathbf{x}}]_c$. At this point, $\mathbf{G} \in \mathbb{R}^{H \times W}$ contains the spatial relationship between the targeted patch feature and pixels. However, because \mathbf{G} arises from the forward and backward operations of a single image, \mathbf{G} is strongly dependent on the input image rather than on the ViT’s properties. To capture the general behavior of the ViT, \mathbf{G} is averaged over a sufficiently large number of images. At this time, because negative values in \mathbf{G} cancel out the positive values, we ignore the negative importance using ReLU [0, 46, 28]. Thus, we obtain $\mathbf{R} = \frac{1}{|S|} \sum_{\mathbf{x} \in S} \text{ReLU}(\mathbf{G})$, where S denotes an image dataset. Because it is averaged

¹The concatenation of the class token is ignored in this study for notational simplicity.

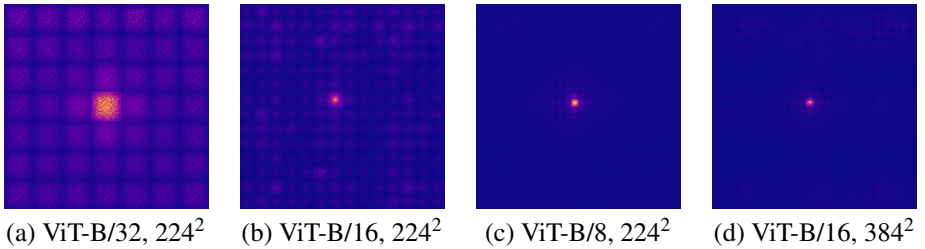


Figure 1: ERFs of ViT with different patch sizes. Because printed figures can be seen improperly, we highly encourage viewing all images electronically with zoom.

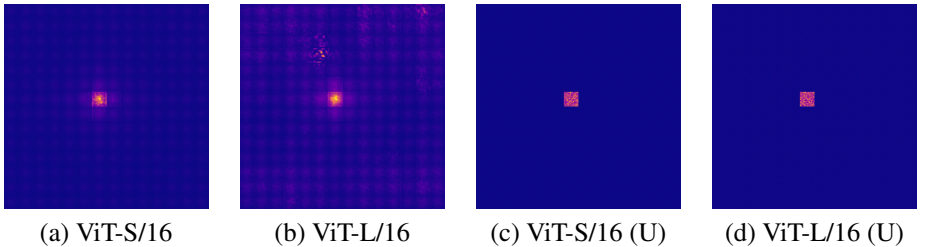


Figure 2: ERFs of ViT with different model sizes. (U) indicates an untrained model.

over numerous images, $\mathbf{R} \in \mathbb{R}^{H \times W}$ represents the general relationship between the pixels and the targeted patch feature of the ViT corresponding to ERF.

3 Effective Receptive Fields of Vision Transformers

3.1 Qualitative Analysis

In this section, we qualitatively analyze the ERFs of ViTs. Although the ERFs of ResNets [13, 57] resemble a 2D Gaussian, the ERFs of ViTs exhibit a different shape owing to nonoverlapping patch partitioning [27]. Figure 1 shows the ERFs of ViT-B with different patch sizes of $\{32, 16, 8\}$. First, the ERF of ViT mainly highlights the targeted central patch and slightly uses information from other patches. This behavior indicates that each patch feature is responsible for representing information in the corresponding patch, while it is combined with certain global information from other patches via SA.

For ViT, a large-sized model was observed to yield a widespread ERF (Figure 1 (b) and Figure 2 (a, b)). This observation is expected because a larger ViT further stacks wider layers. However, for untrained ViTs, the ERFs exhibited no difference (Figure 2 (c, d)). This observation shows that the wider ERF of the ViT-L/16 arises from not only architectural largeness but also pretrained weights.

In addition, we obtained ERFs of other ViT variants (Figure 3). Among them, the DeiT [60], DeiT III [63], and BEiT [9] have nearly identical architectures to the ViT with different pretrained weights, yielding similar but slightly different ERFs. CaiT [62] exhibits architectural modifications, such as class attention, but yielded a similar ERF to that of ViT. Compared with others, XCiT [10] and Swin [20] exhibit wider ERFs. These include explicit modules that allow communication with the neighboring patches, such as the local patch

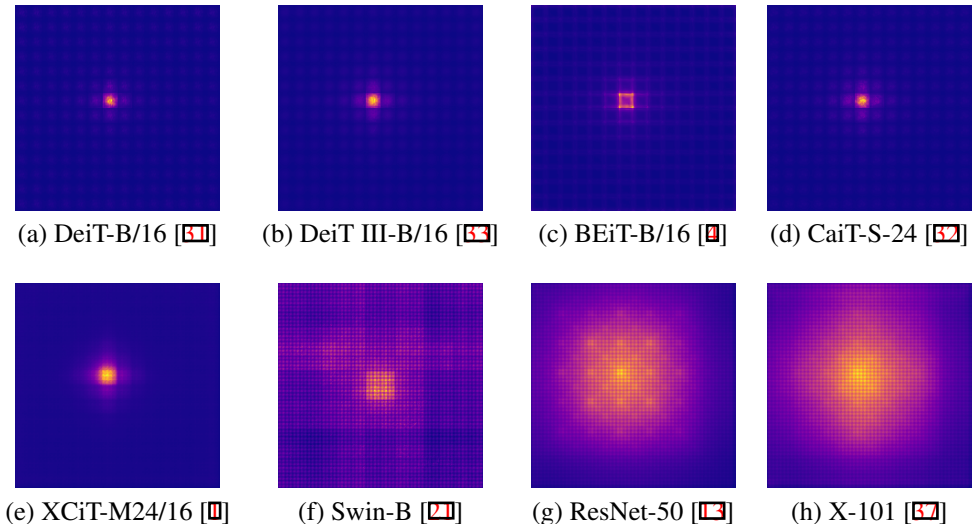


Figure 3: ERFs of ViTs and CNNs. “X” indicates ResNeXt with 32 cardinality.

interaction module, patch merging layer, and shifted window partitioning. In summary, we observed that the ERFs of ViTs were represented as highlights of the targeted patch area with other patches being partially utilized (See the Appendix for further analysis of the ERF of ViT).

3.2 Spatial Understanding of Images by ViTs

The interesting aspect of ERF is that it illustrates how ViTs understand spatial images. Although the ERF of ViT shows that the majority of the activated pixels are in the target patch, adjacent patches are more activated than distant patches, yielding a roughly \oplus -shape. This behavior implies that the ViTs have order-awareness in the sequence of patches, enabling them to use more information from nearby patches and less from far patches, which we refer to as the spatial understanding of images. The spatial understanding of images is one of the critical components for obtaining high-performance ViT. Wu et al. [35] and Liu et al. [21] observed that ViTs without any positional embedding yielded decreased performance.

Positional embeddings in ViTs exist in diverse forms. The absolute positional embedding (APE) can be either a predefined sinusoidal sequence [34] or learnable parameter [4] and is added to the patch embedding. The relative positional embedding (RPE) [24, 29], also called attention bias [22], is added to the attention matrix for each l th layer as

$$\text{Attention}_l(\mathbf{Q}_l, \mathbf{K}_l, \mathbf{V}_l) = \text{softmax} \left(\frac{\mathbf{Q}_l \mathbf{K}_l^\top}{\sqrt{D}} + \mathbf{B}_{\text{rel},l} \right) \mathbf{V}_l, \quad (1)$$

where $\mathbf{Q}_l, \mathbf{K}_l, \mathbf{V}_l \in \mathbb{R}^{N \times D}$ are the query, key, and value in SA, respectively, and $\mathbf{B}_{\text{rel},l} \in \mathbb{R}^{N \times N}$ is the RPE as attention bias. To obtain $\mathbf{B}_{\text{rel},l}$, the original Swin transformer [22] used a learnable table called RelPosBias that provided $\mathbf{B}_{\text{rel},l}$ for each relative coordinate. SwinV2 [22] employed a learnable MLP to obtain $\mathbf{B}_{\text{rel},l}$ from each relative coordinate; this term is also called RelPosMlp. Although the original ViT used APE [4], recent ablation

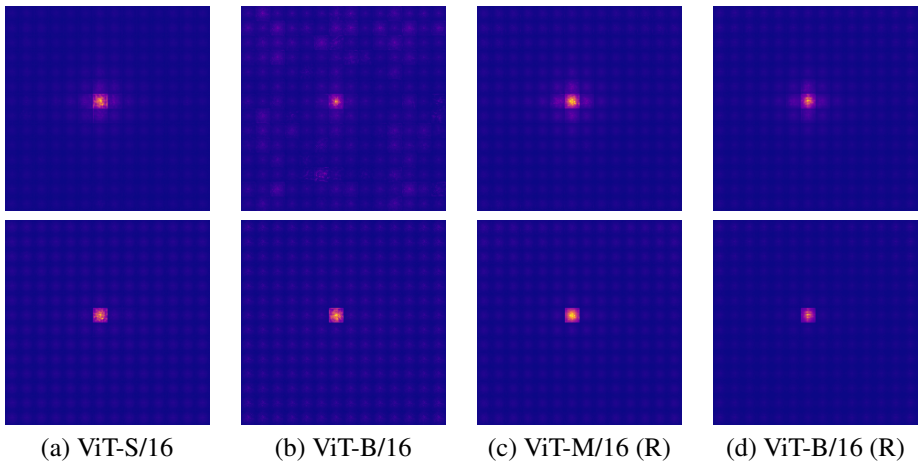


Figure 4: ERFs of ViTs, where (R) indicates the model with RPE. The second row illustrates ERFs when the APE or RPE is re-initialized to random parameters. Note that the \oplus -shape is lost in the second row.

studies [4, 21, 30, 35] have reported that using RPE yielded improved performance. Bello [5] claims that although APE has provided successful results in natural language processing tasks, relative information from RPE is crucial for vision tasks.

We examined the role of positional embeddings. We obtained ERFs before and after the APE or RPE of pretrained ViTs was re-initialized to random parameters (Figure 4). We observed that re-initializing APE or RPE altered ERFs, causing adjacent patches to lose their contribution to the target patch feature and exhibit the same contribution as far patches. This observation clearly demonstrates that SA itself cannot understand the location of patches, and positional embedding plays a significant role in the spatial understanding of images.

To investigate the underlying mechanism, we extracted the learned and untrained RPEs. From $\mathbf{B}_{\text{rel},l} \in \mathbb{R}^{N \times N}$, the RPE of the n th patch $\mathbf{B}_{\text{rel},l,n} \in \mathbb{R}^N$ was obtained for $n \in \{0, 1, \dots, N-1\}$ and was reshaped into $\mathbf{B}'_{\text{rel},l,n} \in \mathbb{R}^{H/P \times W/P}$. For visualization, $\mathbf{B}'_{\text{rel},l,n}$ was averaged over multi-head. The RPE in the first attention layer is visualized in Figure 5. Note that the learned RPE appeared as a sliced 2D Gaussian, distinguishing between near and distant patches. Because softmax computes an exponential ratio, the bias term $\mathbf{B}_{\text{rel},l}$ in Eq. 1 becomes an exponential coefficient that amplifies each element of the attention matrix. Thus, a higher RPE value means a larger amplification of the corresponding patch. Importantly, we observed that untrained RPEs showed distance-independent values. In other words, re-initializing RelPosMlp provides a random RPE that does not discriminate between near and far patches.

We also examined whether RPEs appear as 2D Gaussians across all layers (Figure 6). We fitted the ERF of ViTs to a 2D Gaussian using the `LMfit` [25] library (Table 1). The coefficient of determination R^2 indicates how exactly ERF fits a 2D Gaussian, ideally 1. The standard deviations $\hat{\sigma}_X$ and $\hat{\sigma}_Y$ represent the wideness of the 2D Gaussian. We discovered that, for the majority of layers, RPE fitted to the 2D Gaussian with $R^2 > 0.7$. The exceptions, whose RPE showed no pattern, were found in the last two layers of $l \in \{11, 12\}$ that were close to the classifier head.

We interpret these observations as follows: Initially, an untrained RPE exhibits a ran-

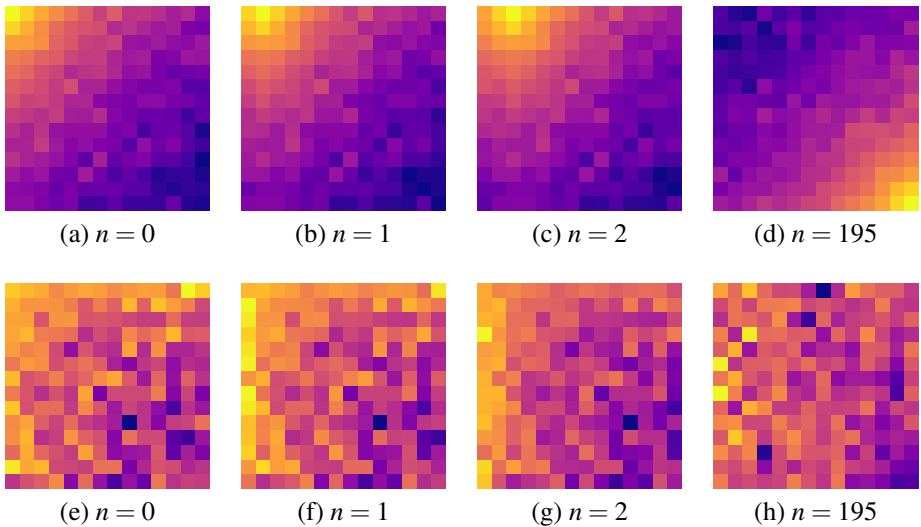


Figure 5: RPE of ViT-B/16 (R) for each patch index. The first row is obtained from the pretrained model, whereas the second row is obtained from the untrained model.

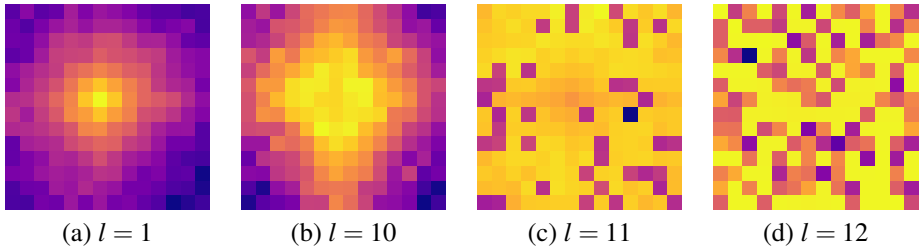


Figure 6: RPE corresponding to the center was extracted for each layer of ViT-B/16 (R).

l	ViT-S/16, 224^2 (R)			ViT-M/16, 224^2 (R)			ViT-B/16, 224^2 (R)		
	R^2	$\hat{\sigma}_x$	$\hat{\sigma}_y$	R^2	$\hat{\sigma}_x$	$\hat{\sigma}_y$	R^2	$\hat{\sigma}_x$	$\hat{\sigma}_y$
1	0.731	6.837	7.063	0.893	4.219	4.113	0.914	4.553	4.394
2	0.798	4.704	4.538	0.728	6.257	5.679	0.573	6.672	6.719
3	0.831	6.185	6.392	0.824	4.715	5.039	0.870	4.649	4.849
4	0.867	4.757	5.020	0.838	5.250	5.355	0.813	4.901	5.404
5	0.753	6.798	5.310	0.795	5.597	4.920	0.853	5.055	4.807
6	0.730	5.624	4.631	0.694	8.054	5.540	0.817	5.421	4.276
7	0.796	5.872	4.848	0.844	5.509	4.660	0.877	6.895	5.020
8	0.805	4.865	5.473	0.798	5.715	5.010	0.825	5.640	4.006
9	0.771	5.668	5.681	0.729	5.472	6.538	0.873	5.328	4.914
10	0.786	5.111	6.125	0.878	4.430	5.348	0.896	5.342	6.132
11	0.231	8.709	272.743	0.359	5.824	298.676	0.012	21.137	702.646
12	0.019	690.530	181.928	0.002	396.639	415.174	0.004	579.639	332.651

Table 1: Results of fitting RPEs to a 2D Gaussian.

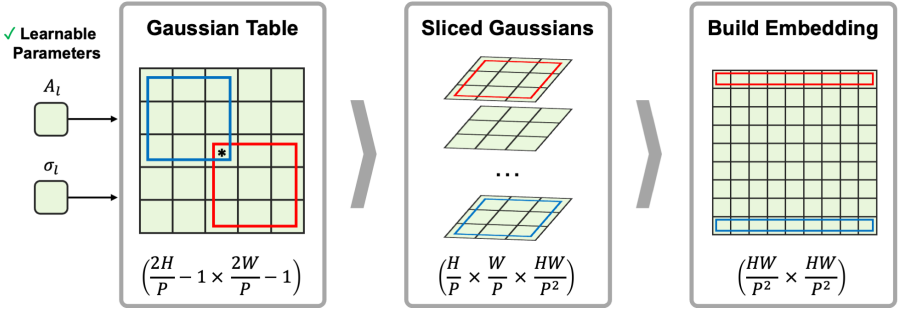


Figure 7: Illustration on how we obtain Gaussian attention bias.

dom pattern and cannot distinguish between near and far patches. However, because RPE is learnable, ViTs can choose to *acquire* an understanding of the different positions of patches. After training, the RPE becomes a pattern close to a 2D Gaussian, discriminating the different positions of patches. The learned RPE allows ViTs to understand near and far patches. These observations motivate us to design a new RPE method.

4 Proposed Method

Our objective is to design an RPE that easily recognizes close and distant patches to facilitate ViTs to acquire spatial understanding of images. In light of the observation that learned RPE fits suitably with a 2D Gaussian, we propose injecting *Gaussian attention bias* into RPE:

$$\text{Attention}_l(\mathbf{Q}_l, \mathbf{K}_l, \mathbf{V}_l) = \text{softmax} \left(\frac{\mathbf{Q}_l \mathbf{K}_l^\top}{\sqrt{D}} + \mathbf{B}_{\text{rel},l} + \mathbf{B}_{\text{Gaussian},l} \right) \mathbf{V}_l. \quad (2)$$

Here, we aim to build $\mathbf{B}_{\text{Gaussian},l}$ so that the bias terms $\mathbf{B}_{\text{rel},l} + \mathbf{B}_{\text{Gaussian},l}$ readily appear as a 2D Gaussian, even in the initial state. By reversing the process of extracting RPE in Figure 5, we build $\mathbf{B}_{\text{Gaussian},l}$ by stacking sliced 2D Gaussians.

First, for the l th layer, we generate a 2D Gaussian table using $A_l, \sigma_l \in \mathbb{R}$:

$$f(x, y) = A_l^2 \exp \left(- \left(\frac{(x - x_c)^2}{2\sigma_l^2} + \frac{(y - y_c)^2}{2\sigma_l^2} \right) \right), \quad (3)$$

where $x = 1, 2, \dots, 2W/P - 1, y = 1, 2, \dots, 2H/P - 1$, and (x_c, y_c) correspond to the central coordinate. Note that the amplitude is set to A_l^2 to ensure a non-negative amplitude for any A_l . The variance σ_l^2 is shared for the horizontal and vertical directions. Thus, we parameterize the 2D Gaussian with only two parameters, A_l and σ_l .

Second, sliced Gaussians are obtained. As shown in Figure 7, the first sliced Gaussian should have the center coordinate of the 2D Gaussian (*) at the top-left (red box), whereas the last sliced Gaussian should exhibit the center coordinate (*) at the bottom-right (blue box). This slicing ensures that it resembles the learned RPE in Figure 5. Finally, each sliced Gaussian is reshaped and stacked to build $\mathbf{B}_{\text{Gaussian},l}$, whose size is the same as $\mathbf{B}_{\text{rel},l}$.

Our design of Gaussian attention bias has several advantages. First, because we designed Gaussian attention bias as an additional bias $\mathbf{B}_{\text{Gaussian},l}$, it can be seamlessly plugged into

Dataset	Model	RPE w/o GAB	RPE w/ GAB	Difference
ImageNet-1K	ViT-S/16 (R)	80.567	80.724	+0.157
	ViT-M/16 (R)	81.224	81.249	+0.025
	ViT-B/16 (R)	81.381	81.484	+0.103

Table 2: Top-1 accuracy on the ImageNet-1K dataset. All the accuracies in this paper are expressed in percentage units. ‘‘GAB’’ indicates Gaussian attention bias.

any type of RPE, including RelPosBias and RelPosMlp. In other words, if we attempt to implement a redesign of RelPosMlp to resemble a 2D Gaussian, it cannot be applied to other RPEs, such as RelPosBias.

Second, our Gaussian attention bias is hyperparameter-free. Note that $\mathbf{B}_{\text{Gaussian},l}$ is parameterized by A_l and σ_l with a differentiable function (Eq. 3). Thus, A_l and σ_l can be set as learnable parameters in gradient descent optimization. We do not need the trial-and-error-based hyperparameter tuning on A_l and σ_l . Because σ_l determines the wideness of the 2D Gaussian, the flexibility of σ_l is beneficial when using ViTs for other datasets or tasks that require different sizes of ERF. Furthermore, different A_l and σ_l values are allowed for each layer. As we observed in Table 1, as the last two layers did not learn to be a 2D Gaussian, it is preferable to allow different behaviors in the last two layers. For example, the last two layers can naturally choose A_l to be zero.

Finally, we benefit from the learnability of the original RPE, such as RelPosBias or RelPosMlp. Indeed, RPEs such as RelPosBias or RelPosMlp have a significant number of parameters that enable enriched expression in SA. Considering this behavior, we allow the degree of freedom of the original RPE.

However, we remove unnecessary degrees of freedom from our Gaussian attention bias. We do not generate multiple Gaussian tables; rather, we use a single Gaussian table to ensure that sliced Gaussians are shifted versions of each other, inspired by the use of relative coordinates in RPE. We do not use a constant term in our Gaussian function at Eq. 3 because softmax is invariant to constant translation [19, 24]: $\text{softmax}(\mathbf{x} + C) = \text{softmax}(\mathbf{x})$. Finally, we choose to share the Gaussian attention bias across multiple heads of SA within the same layer (See the Appendix for the ablation study).

5 Experiments

Now, we investigate the influence of Gaussian attention bias on the performance of ViTs.

ImageNet-1K First, we trained the ViTs on the image classification task using the ImageNet-1K dataset [9] from scratch. ViT-{S, M, L}/16 (R) using RelPosMlp without APE were used. See the Appendix for experimental details, such as the hyperparameters. For each model with and without Gaussian attention bias, the top-1 accuracy was measured (Table 2). The top-1 accuracy of the three models improved after incorporating Gaussian attention bias.

Other Datasets To further examine the performance difference, we targeted image classification on other datasets: Oxford-IIIT Pet [26], Caltech-101 [10], Stanford Cars [18], and Stanford Dogs [15]. Test accuracy was measured for each ViT that used RelPosMlp with and without Gaussian attention bias. We observed that the use of Gaussian attention bias consistently improved the test accuracies of the three ViTs on the four datasets (Table 3). For these

Dataset	Model	RPE w/o GAB	RPE w/ GAB	Difference
Oxford-IIIT Pet	ViT-S/16 (R)	91.486	92.780	+1.294
	ViT-M/16 (R)	92.810	92.960	+0.150
	ViT-B/16 (R)	93.381	93.743	+0.362
Caltech-101	ViT-S/16 (R)	88.403	90.202	+1.799
	ViT-M/16 (R)	89.132	89.983	+0.851
	ViT-B/16 (R)	89.254	89.570	+0.316
Stanford Cars	ViT-S/16 (R)	80.126	83.079	+2.953
	ViT-M/16 (R)	80.731	83.890	+3.159
	ViT-B/16 (R)	80.154	82.612	+2.458
Stanford Dogs	ViT-S/16 (R)	81.535	82.507	+0.972
	ViT-M/16 (R)	85.088	85.714	+0.626
	ViT-B/16 (R)	89.256	90.185	+0.929

Table 3: Test accuracy with and without Gaussian attention bias on other datasets.

Backbone	RPE Method	COCO		ADE20K	
		AP ^{box}	AP ^{mask}	mIoU	aAcc
Swin-S	RelPosBias w/o GAB	48.12	43.03	46.16	81.82
	RelPosBias w/ GAB	48.23	43.13	46.41	82.09
	Difference	+0.11	+0.10	+0.25	+0.27

Table 4: Experimental results in terms of object detection and semantic segmentation.

experiments, each dataset contained objects of various sizes, whose classification requires different sizes of ERF or σ_l . Because we designed σ_l as learnable, the model with Gaussian attention bias can flexibly cope with different ERFs. Indeed, the learned σ_l achieved similar but slightly different values for each dataset (See the Appendix).

Object Detection and Semantic Segmentation Finally, we targeted two downstream tasks: object detection on the COCO 2017 dataset [24] and semantic segmentation on the ADE20K dataset [40]. To further investigate our proposed method with a different setup, we targeted Swin-S with RelPosBias as the backbone. Using the COCO dataset, we measured bounding box mAP (AP^{box}) on object detection and segmentation mAP (AP^{mask}) on instance segmentation. Using the ADE20K dataset, the mean intersection over union (mIoU) and mean accuracy over all pixels (aAcc) were measured. We observed that the Swin transformers with Gaussian attention bias exhibited improvements across all four indices (Table 4).

6 Conclusion

In this study, we analyzed how ViTs understand spatial images. From detailed analyses of the ERF of ViTs, we discovered that ViTs acquired spatial understanding of images during training and that this phenomenon was caused by the underlying transition from randomized positional embedding to a learned one. To guide the understanding of the spatial nearness

and farness of patches, we proposed injecting Gaussian attention bias into ViTs. In several experiments on image classification, object detection, and semantic segmentation, ViTs integrated with Gaussian attention bias achieved superior results.

Acknowledgements

This work was supported by Samsung Electronics Co., Ltd (IO201210-08019-01).

References

- [1] Alaaeldin Ali, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, and Hervé Jégou. XCiT: Cross-Covariance Image Transformers. In *NeurIPS*, pages 20014–20027, 2021.
- [2] André Araujo, Wade Norris, and Jack Sim. Computing receptive fields of convolutional neural networks. *Distill*, 4(11):e21, 2019.
- [3] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization. *CoRR*, abs/1607.06450, 2016.
- [4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. In *ICLR*, 2022.
- [5] Irwan Bello. LambdaNetworks: Modeling long-range Interactions without Attention. In *ICLR*, 2021.
- [6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. In *ECCV (1)*, volume 12346, pages 213–229, 2020.
- [7] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In *WACV*, pages 839–847, 2018.
- [8] Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. On the Relationship between Self-Attention and Convolutional Layers. In *ICLR*, 2020.
- [9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021.
- [11] Li Fei-Fei, Robert Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Comput. Vis. Image Underst.*, 106(1):59–70, 2007.

- [12] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. LeViT: a Vision Transformer in ConvNet’s Clothing for Faster Inference. In *ICCV*, pages 12239–12249, 2021.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016.
- [14] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Ian Simon, Curtis Hawthorne, Noam Shazeer, Andrew M. Dai, Matthew D. Hoffman, Monica Dinulescu, and Douglas Eck. Music Transformer: Generating Music with Long-Term Structure. In *ICLR*, 2019.
- [15] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. CVPR workshop on fine-grained visual categorization (FGVC)*, volume 2, 2011.
- [16] Bum Jun Kim, Hyeyeon Choi, Hyeonah Jang, Dong Gu Lee, Wonseok Jeong, and Sang Woo Kim. Dead pixel test using effective receptive field. *Pattern Recognit. Lett.*, 167:149–156, 2023.
- [17] Jannik Kossen, Neil Band, Clare Lyle, Aidan N. Gomez, Thomas Rainforth, and Yarin Gal. Self-Attention Between Datapoints: Going Beyond Individual Input-Output Pairs in Deep Learning. In *NeurIPS*, pages 28742–28756, 2021.
- [18] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D Object Representations for Fine-Grained Categorization. In *ICCV Workshops*, pages 554–561, 2013.
- [19] Anirban Laha, Saneem Ahmed Chemmengath, Priyanka Agrawal, Mitesh M. Khapra, Karthik Sankaranarayanan, and Harish G. Ramaswamy. On Controllable Sparse Alternatives to Softmax. In *NeurIPS*, pages 6423–6433, 2018.
- [20] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV (5)*, volume 8693, pages 740–755, 2014.
- [21] 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 *ICCV*, pages 9992–10002, 2021.
- [22] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin Transformer V2: Scaling Up Capacity and Resolution. In *CVPR*, pages 11999–12009, 2022.
- [23] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard S. Zemel. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In *NIPS*, pages 4898–4906, 2016.
- [24] André F. T. Martins and Ramón Fernández Astudillo. From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification. In *ICML*, volume 48, pages 1614–1623, 2016.

- [25] Matthew Newville, Till Stensitzki, Daniel B Allen, Michal Rawlik, Antonino Ingarciola, and Andrew Nelson. LMFIT: Non-linear least-square minimization and curve-fitting for Python. *Astrophysics Source Code Library*, pages ascl-1606, 2016.
- [26] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, pages 3498–3505, 2012.
- [27] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do Vision Transformers See Like Convolutional Neural Networks? In *NeurIPS*, pages 12116–12128, 2021.
- [28] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.*, 128(2):336–359, 2020.
- [29] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-Attention with Relative Position Representations. In *NAACL-HLT (2)*, pages 464–468, 2018.
- [30] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck Transformers for Visual Recognition. In *CVPR*, pages 16519–16529, 2021.
- [31] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, volume 139, pages 10347–10357, 2021.
- [32] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with Image Transformers. In *ICCV*, pages 32–42, 2021.
- [33] Hugo Touvron, Matthieu Cord, and Hervé Jégou. DeiT III: Revenge of the ViT. In *ECCV (24)*, volume 13684, pages 516–533, 2022.
- [34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *NIPS*, pages 5998–6008, 2017.
- [35] Kan Wu, Houwen Peng, Minghao Chen, Jianlong Fu, and Hongyang Chao. Rethinking and Improving Relative Position Encoding for Vision Transformer. In *ICCV*, pages 10013–10021, 2021.
- [36] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *NeurIPS*, pages 12077–12090, 2021.
- [37] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *CVPR*, pages 5987–5995, 2017.
- [38] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring Self-Attention for Image Recognition. In *CVPR*, pages 10073–10082, 2020.

-
- [39] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking Semantic Segmentation From a Sequence-to-Sequence Perspective With Transformers. In *CVPR*, pages 6881–6890, 2021.
- [40] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic Understanding of Scenes Through the ADE20K Dataset. *Int. J. Comput. Vis.*, 127(3):302–321, 2019.