








## Spatial Decomposition and Aggregation for Attention in Convolutional Neural Networks

Meng Zhu , Weidong Min ,<sup>†,‡,§</sup> Hongyue Xiang , Cheng Zha ,  
Zheng Huang , Longfei Li  and Qiyang Fu 

*\*School of Mathematics and Computer Science  
Nanchang University, Nanchang 330031  
P. R. China*

*†Institute of Metaverse, Nanchang University  
Nanchang 330031, P. R. China*

*‡Jiangxi Key Laboratory of Smart City  
Nanchang 330031, P. R. China*

*§minweidong@ncu.edu.cn*

Received 27 May 2023

Accepted 22 October 2023

Published 31 January 2024

Channel attention has been shown to improve the performance of deep convolutional neural networks efficiently. Channel attention adaptively recalibrates the importance of each channel, determining what to attend to. However, channel attention only encodes inter-channel information but neglects the importance of positional information. Positional information is crucial in determining where to attend to. To address this issue, we propose a novel channel-spatial attention method named Spatial-Decomposition-Aggregation Attention (SDAA) method. First, a high-axis spatial direction is decomposed into multiple low-axis spatial directions. Then, a shared transformation sub-unit establishes attention in each low-axis space direction. Next, all the low-axis attention masks are aggregated into a high-axis attention mask. Finally, the generated high-axis attention mask is fused into the input features, thus enhancing the input features. Essentially, our method is a divide-and-conquer process. Experimental results demonstrate that our SDAA method outperforms the existing channel-spatial attention methods.

**Keywords:** Convolutional neural networks; channel-spatial attention; spatial decomposition; spatial aggregation.

### 1. Introduction

Deep convolutional neural networks (CNNs) have made significant success in a variety of tasks.<sup>5,16,20,28,33</sup> A set of filters displays neighborhood spatial connection patterns along input channels at each convolutional layer of the network,

<sup>§</sup>Corresponding author.

thus integrating position-wise information and channel-wise information together within local receptive fields. CNNs can create reliable representations which capture hierarchical patterns and achieve global theoretical receptive fields via stacking a number of convolutional layers, nonlinear activation function layers and downsampling operators.

Recently, attention mechanisms have been gaining attention in research communities as they can enhance feature representations by focusing on essential features while suppressing unnecessary ones.<sup>10,13,17,34</sup> These mechanisms improve the feature representations generated by standard convolutional layers by means of explicitly building dependencies among channels or using weighted spatial masks for spatial attention. The idea behind learning attention weights is to enable the network to learn what and where to focus on.

The Squeeze-and-Excitation (SE)<sup>10</sup> attention remains the most widely used attention mechanism in CNNs. It computes channel attention using global average pooling and offers significant performance gains at a relatively low computational cost. However, SE attention only focuses on encoding inter-channel information and disregards the importance of positional information, which is crucial for capturing object structures in vision tasks.<sup>32</sup> Recent approaches, such as convolutional block attention module (CBAM)<sup>34</sup> and bottleneck attention module (BAM),<sup>23</sup> are aimed to incorporate positional information by reducing the channel dimension of the input tensor and then computing spatial attention using small kernel convolutions. Nevertheless, the small kernel convolution is limited in their ability to model long-range dependencies, which are essential for vision tasks.<sup>37</sup>

To address the above problems, a novel channel-spatial attention method called Spatial-Decomposition-Aggregation Attention (SDAA) method is presented in this paper. Initially, a high-axis spatial direction is broken down into several low-axis spatial directions. Subsequently, a common transformation sub-unit establishes attention in each low-axis spatial direction. Afterwards, the low-axis attention masks are combined into a high-axis attention mask. Finally, the resulting high-axis attention mask is integrated into the input features, thereby augmenting the input features. Fundamentally, our approach is a strategy of divide-and-conquer. In this shared transformation sub-unit, gated self-attention is used to establish long-distance position relationships, the matrix is used to establish channel relationships, and they are stacked together in series.

The main contributions of this paper are concluded as follows:

- (1) We propose SDAA method. Our SDAA method decomposes a high-axis spatial direction into multiple low-axis spatial directions and aggregates correlations established at all low-axis spatial directions.
- (2) Within this shared transformation sub-unit, while gated self-attention is utilized to build long-range position correlations, the matrix is employed to establish channel correlations, and they are stacked together in series.

The rest of this paper is organized as follows. Related work is reviewed in Sec. 2. Our attention method is introduced in Sec. 3. Experimental results are shown in Sec. 4. The discussion is made in Sec. 5. The conclusion is drawn in Sec. 6.

## 2. Related Work

By means of combining the strengths of channel attention and spatial attention, channel-spatial attention is capable of adaptively selecting significant objects and regions.<sup>2</sup> The field of channel-spatial attention was first introduced in the residual attention network,<sup>31</sup> which emphasized the importance of informative features in both channel and spatial dimensions. This approach used a bottom-up structure comprising multiple convolutions to generate a three-axis attention map (channel, height and width). However, it suffered from high computational costs and limited receptive fields. Later works<sup>8,19,22,23,34,36</sup> have improved the discriminative power of features to leverage global spatial information by means of incorporating global average pooling and separating channel attention from spatial attention for better computational efficiency.

Woo *et al.*<sup>34</sup> presented the CBAM which stacked channel attention and spatial attention in series to enhance informative channels and important regions. The channel attention map and spatial attention map were decoupled for computational efficiency, and global pooling was used to leverage spatial global information. By means of combining channel attention and spatial attention sequentially, CBAM could utilize both channel and spatial correlations of features to guide the network on what to focus on and where to focus. Specifically, it emphasized useful channels and enhances informative local regions. However, there is still room for improvement in CBAM. For example, using a convolution to produce the spatial attention in CBAM might result in a limited receptive field for the spatial sub-module. Park *et al.*<sup>23</sup> proposed the BAM alongside CBAM, with the goal of enhancing network representational capability in a computationally efficient manner. BAM used dilated convolutions to expand the receptive field of the spatial attention sub-module and incorporated a bottleneck structure to save on computation. BAM inferred channel attention and spatial attention in parallel streams and then combined the two attention maps after resizing both branch outputs. However, while dilated convolutions were effective in expanding the receptive field, they still fell short in capturing long-range position information and encoding long-range position correlations. The success of attention mechanisms that rely solely on weak supervisory signals from class labels prompted Linsley *et al.*<sup>19</sup> to explore how explicit human supervision could improve the performance and interpretability of attention models. As a proof of concept, they proposed the global-and-local attention (GALA) module, which extended the SE block with a spatial attention mechanism. To address the issue of computational complexity, Zhang *et al.*<sup>36</sup> proposed an efficient Shuffle Attention (SA) module that combined two types of attention mechanisms using Shuffle Units. The SA module grouped channel dimensions

into multiple sub-features and processes them in parallel. For each sub-feature, a Shuffle Unit was used to depict feature dependencies in both channel and spatial dimensions. The resulting sub-features were then aggregated, and a “channel shuffle” operator was employed to enable information communication between different sub-features.

In CBAM and BAM, channel attention and spatial attention were calculated separately, without taking into account the interplay between these two domains.<sup>21</sup> To capture cross-dimension interactions, Misra *et al.*<sup>21</sup> proposed triplet attention (TA), a lightweight yet powerful attention mechanism. In contrast to CBAM and BAM, triplet attention emphasized the significance of capturing cross-domain interactions rather than computing spatial attention and channel attention independently. This enabled the triplet attention to capture more informative and discriminative feature representations.

Yang *et al.*<sup>35</sup> highlighted the significance of learning attention weights that varied across both channel and spatial domains in their proposal of SimAM, a straightforward and parameter-free attention module that could directly estimate three-axis weights instead of expanding one-axis or two-axis weights. The design of SimAM was based on established neuroscience theory, eliminating the requirement for manual fine-tuning of the network structure. Wang *et al.*<sup>30</sup> proposed the fully attentional block (FAB), which eliminated the pooling layer to restore spatial information.

The SE block utilized global average pooling to aggregate global spatial information before modeling channel correlations, however, it failed to consider the importance of positional information. On the other hand, BAM and CBAM used convolutions to capture local position correlations but were incapable to model long-range position dependencies. To address these limitations, Hou *et al.*<sup>8</sup> proposed a novel attention mechanism called coordinate attention (CA). This mechanism embedded positional information into channel attention, allowing the network to focus on important areas. The process of coordinate attention involved two steps: coordinate information embedding and coordinate attention generation. Using coordinate attention, the network could accurately determine the position of a target object, and this approach had a larger receptive field than BAM and CBAM. Similar to the SE block, coordinate attention built channel relationships and effectively enhanced the expressive power of learned features. Ouyang *et al.*<sup>22</sup> presented an innovative and effective multi-scale attention (EMA) module, which was aimed to preserve channel-wise information while reducing computational complexity. To achieve this, EMA partitioned the channel dimensions into several sub-features, ensuring that spatial semantic features were evenly distributed within each feature group. In addition to incorporating global information to recalibrate the channel-wise weight in each parallel branch, EMA employed a cross-dimension interaction method to further combine the output features of the two parallel branches.

### 3. Proposed Attention Method

In this section, we will introduce how our SDAA method performs spatial decomposition and aggregation, and how the shared transformation sub-unit establish both channel correlations and long-range position correlations simultaneously. For ease of understanding, Table 1 shows main symbols used in this section.

#### 3.1. Overview of the proposed attention method

First, we introduce the overview of our SDAA method, as seen in Fig. 1. There are four sub-units in our SDAA method: **Spatial Decomposition, Transformation, Spatial Aggregation, Scale.**

**Spatial decomposition.** The spatial decomposition sub-unit is aimed to decompose a high-axis spatial direction into multiple low-axis spatial directions. Assume that the input tensor is  $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$  (including the two-axis spatial direction), that  $\mathbf{U} \in \mathbb{R}^{H \times C}$  (including the one-axis spatial direction) denotes the average pooled feature tensor by means of reducing the  $W$ -axis of the input tensor, and

Table 1. Main symbols used in this section.

Symbol	Definition
$\sum$	Summation
$\odot$	Broadcast element-wise multiplication
$\sigma(\cdot)$	Sigmoid function
$\text{GAP}(\cdot)$	Global average pooling
$r$	Reduction ratio
$H$	Image height
$W$	Image width
$C$	Number of image channels

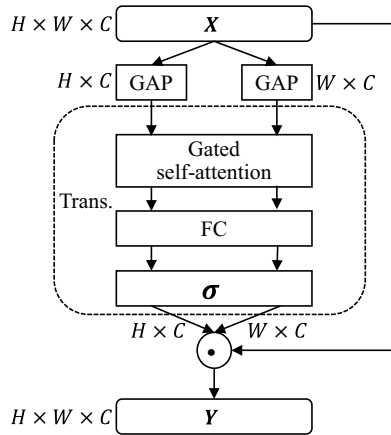


Fig. 1. Overview of our SDAA method. FC denotes fully connection. Trans. denotes transformation.

that  $\mathbf{V} \in \mathbb{R}^{W \times C}$  (including the one-axis spatial direction) denotes the average pooled feature tensor by means of reducing the  $H$ -axis of the input tensor. The spatial decomposition can be formulated as Eq. (1)

$$u_{i,k} = \frac{1}{W} \sum_{j=1}^W x_{i,j,k}, v_{j,k} = \frac{1}{H} \sum_{i=1}^H x_{i,j,k}. \quad (1)$$

Here  $x_{i,j,k}$  is the element at  $(i, j)$  within the  $k$ th feature map.

**Transformation.** The transformation sub-unit is aimed to establish channel correlations and long-range positional correlations for each low-axis spatial direction, and output attention masks, i.e.  $\mathbf{S} = \text{Transformation}(\mathbf{U})$ ,  $\mathbf{T} = \text{Transformation}(\mathbf{V})$ . This transformation sub-unit is shared by all low-axis spatial directions. A detailed introduction of the shared transformation sub-unit is described in Sec. 3.2.

**Spatial aggregation.** The spatial aggregation sub-unit is aimed to aggregate all the generated low-axis attention masks into a high-axis attention mask. Let  $\mathbf{M} \in \mathbb{R}^{H \times W \times C}$  denote the aggregated high-axis attention mask. The spatial aggregation can be formulated as Eq. (2)

$$m_{i,j,k} = s_{i,k} t_{j,k}. \quad (2)$$

**Scale.** The scale is aimed to enhance input feature representation by means of fusing the attention mask into the input features. The scale can be formulated as Eq. (3)

$$\mathbf{Y} = \mathbf{X} \odot \mathbf{M}. \quad (3)$$

### 3.2. Transformation sub-unit

Next, we introduce the transformation sub-unit. The details of the transformation sub-unit are described as Algorithm 1.

---

#### Algorithm 1: Details of the transformation sub-unit.

---

**Input:** The input tensor  $\mathbf{Z} \in \mathbb{R}^{N \times C}$ , the trainable weights

$$\Theta_q, \Theta_k \in \mathbb{R}^{C \times (C/(2r))}, \Theta_v, \Theta_u \in \mathbb{R}^{C \times (C/r)}, \mathbf{W} \in \mathbb{R}^{(C/r) \times C}.$$

**Output:** The output tensor  $\mathbf{T} \in \mathbb{R}^{N \times C}$ .

*/\* Using gated self-attention to build long-distance position correlations. \*/*

$$1 \quad \mathbf{Q} = \mathbf{Z}\Theta_q, \mathbf{K} = \mathbf{Z}\Theta_k, \mathbf{V} = \mathbf{Z}\Theta_v, \mathbf{U} = \mathbf{Z}\Theta_u;$$

$$2 \quad \mathbf{S} = \sigma(\mathbf{U}) \odot (\text{Softmax}(\mathbf{Q}\mathbf{K}^T / \sqrt{C/(2r)})\mathbf{V});$$

*/\* Using the matrix to build channel correlations. \*/*

$$3 \quad \mathbf{O} = \mathbf{S}\mathbf{W};$$

*/\* Output normalization. \*/*

$$4 \quad \mathbf{T} = \sigma(\mathbf{O}).$$


---

**Self-attention.** The convolution is limited by its ability to capture local receptive fields, which in turn limits its ability to establish long-range positional dependencies. Although this limitation can be addressed by stacking more convolutional layers, a more direct approach is to use self-attention<sup>29</sup> to model long-range positional dependencies in one step, i.e.  $\text{Softmax}(\mathbf{QK}^T/\sqrt{C/(2r)})\mathbf{V}$ .

**Gated self-attention.** Inspired by the Gated Attention Unit,<sup>11</sup> we have designed a novel gated self-attention mechanism, defined as  $\mathbf{S} = \sigma(\mathbf{U}) \odot (\text{Softmax}(\mathbf{QK}^T/\sqrt{C/(2r)})\mathbf{V})$ . If matrix  $\text{Softmax}(\mathbf{QK}^T/\sqrt{C/(2r)})$  is equal to the scaled identity matrix  $\alpha\mathbf{I}$ ,  $\mathbf{S}$  becomes the Gated Linear Unit (GLU).<sup>6</sup> Therefore, gated self-attention can be seen as a simple and natural fusion of self-attention and GLU.

**Remark.** FAB,<sup>30</sup> coordinate attention,<sup>8</sup> EMA<sup>22</sup> and our SDAA method share some similarities, but there are also differences. FAB models channel attention directly on the original feature map by means of removing pooling. However, removing pooling results in the inability to capture global information and increases computational complexity. Additionally, FAB, coordinate attention, and EMA implicitly establish positional correlations, i.e. encoding positional information into channel attention. In contrast, our SDAA method explicitly establishes long-range positional dependencies through self-attention.

## 4. Experiments

In this section, we conducted experiments to evaluate our SDAA method across a range of tasks, datasets and model architectures. All compared attention methods were benchmarked under the same experimental settings.

### 4.1. Implementation details

To evaluate our proposed method on CIFAR<sup>15</sup> classification, we employed two widely used CNNs as backbone models, including EfficientNet<sup>26</sup> and REGNET.<sup>25</sup> The parameters of networks were optimized using Lion<sup>3</sup> with layer adaption<sup>27</sup> and mini-batch size of 128. All models were trained from scratch for 80 epochs by means of setting the initial learning rate to  $2e-3$ . If  $\text{step} < 390$ , the learning rate scaling factor linearly increased from 0 to 1. If  $390 \leq \text{step} < 21450$ , the learning rate scaling factor decayed inversely with step from 1 to 0.1. If  $\text{step} \geq 21450$ , the learning rate scaling factor remained constant at 0.1.

We further evaluated our methods on PASCAL VOC<sup>4</sup> using SSDlite<sup>9</sup> and RetinaNet.<sup>18</sup> For training SSDlite, the network was trained from scratch for 80 epochs using Adam<sup>14</sup> with mini-batch size of 128. The initial learning rate was set to  $2e-3$ . If  $\text{step} < 133$ , the learning rate scaling factor linearly increased from 0 to 1. If  $133 \leq \text{step} < 7315$ , the learning rate scaling factor decayed inversely with step from 1 to 0.1. If  $\text{step} \geq 7315$ , the learning rate scaling factor remained constant at 0.1. For training RetinaNet, the network was trained from scratch for 40 epochs

using Adam with mini-batch size of 32. The initial learning rate was set to  $2e - 3$ . If  $\text{step} < 534$ , the learning rate scaling factor linearly increased from 0 to 1. If  $534 \leq \text{step} < 14418$ , the learning rate scaling factor decayed inversely with step from 1 to 0.1. If  $\text{step} \geq 14418$ , the learning rate scaling factor remained constant at 0.1.

In each convolution layer and fully connected layer, the weight was initialized with truncated Xavier normal distribution,<sup>1,7</sup> the bias if having was initialized with zeros. In each batch normalization<sup>12</sup> layer, the  $\gamma$  was initialized with ones and the  $\beta$  if having was initialized with zeros.

All of experimental tasks were accomplished on one computer. Its configurations were as follows: CPU Intel(R) Core(TM) i7-11700K, GPU Nvidia RTX A5000, RAM 32GB, Python 3.11.2, PyTorch<sup>24</sup> 2.0.1, Cudatoolkit 11.8 and Cudnn 8.9.

## 4.2. Image classification

To evaluate the influence of our methods, we first performed experiments on the CIFAR dataset. This dataset is composed of colored natural images with  $32 \times 32$  pixels. The CIFAR-100 dataset is composed of 100 categories with 600 images each category. The CIFAR-10 dataset is composed of 10 categories with 6000 images each category. For training on CIFAR, we used standard data augmentation of random color space transformation and random horizontal flip. We reported the top-1 error and top-5 error as the evaluation metrics. The results on CIFAR-100 are reported in Table 2. The results on CIFAR-10 are reported in Table 3.

**Integration with modern architectures.** SE has become a basic component of EfficientNet and REGNETY. To study the integration of different channel-spatial attention methods with modern network architectures, we chose EfficientNet and REGNETY as benchmark networks, and replaced the original SE in the network with different channel-spatial attention methods. The corresponding results are reported in Table 2. When using different channel-spatial attention methods to replace the original SE in EfficientNet-B0, our SDAA achieved the optimal top-1 and top-5 errors simultaneously on CIFAR-100, reduced the top-1 error by 1.22% compared to SE, and reduced the top-5 error by 1.27% compared to SE on CIFAR-100. When using different channel-spatial attention methods to replace the original SE in RegNETY-800MF, our SDAA also achieved the optimal top-1 and top-5 errors simultaneously on CIFAR-100, reduced the top-1 error by 1.58% compared to SE, and reduced the top-5 error by 1.39% compared to SE on CIFAR-100. It is worth noting that while FAB and SE performed similarly, FAB incurred higher computational costs compared to SE. This suggests that directly removing pooling to restore position information does not lead to accuracy improvement, but instead increases computational cost.

**Different image classification datasets.** To study the performance of channel-spatial attention on different datasets, we selected the CIFAR-100 and CIFAR-10 datasets as the benchmark datasets. The corresponding results are reported



Table 2. Comparison results between different attention methods on CIFAR-100. None means the attention mechanism used in the original backbone network is removed. The **bold** denotes the best result under the same backbone network.

Model	Year	Top-1 Err. (%) ↓	Top-5 Err. (%) ↓	Param. (M) ↓	FLOPs (G) ↓
EfficientNet-B0 <sup>26</sup> + none	ICML2019	38.43	13.92	3.499	0.119
+ SE <sup>10</sup>	CVPR2018	38.24	13.63	4.126	0.120
+ CBAM <sup>34</sup>	ECCV2018	40.15	15.10	4.128	0.121
+ BAM <sup>23</sup>	BMVC2018	38.18	13.74	4.705	0.140
+ FAB <sup>30</sup>	ECCV2018	38.62	13.78	4.136	0.140
+ GALA <sup>19</sup>	ICLR2019	44.28	17.63	4.458	0.131
+ TA <sup>21</sup>	WACV2021	60.03	31.30	3.504	0.131
+ CA <sup>8</sup>	CVPR2021	38.23	13.76	4.459	0.125
+ SA <sup>36</sup>	ICASSP2021	38.23	13.67	3.506	0.121
+ SimAM <sup>35</sup>	ICML2021	39.56	14.34	3.499	0.119
+ EMA <sup>22</sup>	ICASSP2023	38.31	13.90	8.209	0.705
+ SDAA (Ours)	-	<b>37.02</b>	<b>12.36</b>	4.754	0.132
REGNETY-800MF <sup>25</sup> + none	CVPR2020	42.06	16.33	4.880	0.266
+ SE <sup>10</sup>	CVPR2018	41.32	15.75	5.720	0.267
+ CBAM <sup>34</sup>	ECCV2018	42.66	16.63	5.722	0.268
+ BAM <sup>23</sup>	BMVC2018	41.25	15.50	7.848	0.367
+ FAB <sup>30</sup>	ECCV2018	41.27	15.85	5.726	0.305
+ GALA <sup>19</sup>	ICLR2019	43.63	16.79	6.151	0.287
+ TA <sup>21</sup>	WACV2021	48.31	21.60	4.884	0.273
+ CA <sup>8</sup>	CVPR2021	40.31	15.11	6.152	0.276
+ SA <sup>36</sup>	ICASSP2021	41.45	15.65	4.881	0.267
+ SimAM <sup>35</sup>	ICML2021	42.17	16.51	4.880	0.266
+ EMA <sup>22</sup>	ICASSP2023	41.31	15.77	4.917	0.322
+ SDAA (Ours)	-	<b>39.74</b>	<b>14.36</b>	6.561	0.287

in Tables 2 and 3. When integrated with EfficientNet-B0, our SDAA method almost consistently achieved the best results on both CIFAR-100 and CIFAR-10. When integrated with RegNETY-800MF, our SDAA method consistently achieved the best results on both CIFAR-100 and CIFAR-10. Overall, our SDAA method achieved greater robustness than existing channel-spatial attention methods on various image classification datasets.

### 4.3. Object detection

We next conducted experiments on the PASCAL VOC 2007  $\cup$  2012.<sup>4</sup> This dataset is composed of 20 classes. The PASCAL VOC 2012trainval dataset has 11,530 images containing 27,450 ROI annotated objects. We trained all detectors on VOC 2012trainval and evaluated the results on VOC 2007 test for comparison. For training on PASCAL VOC, we used standard data augmentation of random color space transformation and random horizontal flip. We reported the metrics of COCO API as the evaluation metrics. The results are reported in Table 4.

Table 3. Comparison results between different attention methods on CIFAR-10.

Model	Year	Top-1 Err. (%) ↓	Top-5 Err. (%) ↓	Param. ( <i>M</i> ) ↓	FLOPs ( <i>G</i> ) ↓
EfficientNet-B0 <sup>26</sup> + none	ICML2019	12.17	<b>0.39</b>	3.499	0.119
+ SE <sup>10</sup>	CVPR2018	12.15	0.49	4.126	0.120
+ CBAM <sup>34</sup>	ECCV2018	13.76	0.57	4.128	0.121
+ BAM <sup>23</sup>	BMVC2018	11.76	0.45	4.705	0.140
+ FAB <sup>30</sup>	ECCV2018	12.12	0.41	4.136	0.140
+ GALA <sup>19</sup>	ICLR2019	13.54	0.56	4.458	0.131
+ TA <sup>21</sup>	WACV2021	27.90	2.49	3.504	0.131
+ CA <sup>8</sup>	CVPR2021	12.20	0.48	4.459	0.125
+ SA <sup>36</sup>	ICASSP2021	12.10	0.47	3.506	0.121
+ SimAM <sup>35</sup>	ICML2021	12.09	0.56	3.499	0.119
+ EMA <sup>22</sup>	ICASSP2023	11.82	0.49	8.209	0.705
+ SDAA (Ours)	-	<b>11.65</b>	0.40	4.754	0.132
REGNETY-800MF <sup>25</sup> + none	CVPR2020	13.52	0.51	4.880	0.266
+ SE <sup>10</sup>	CVPR2018	13.71	0.55	5.720	0.267
+ CBAM <sup>34</sup>	ECCV2018	13.08	<b>0.43</b>	5.722	0.268
+ BAM <sup>23</sup>	BMVC2018	13.06	0.44	7.848	0.367
+ FAB <sup>30</sup>	ECCV2018	13.80	0.47	5.726	0.305
+ GALA <sup>19</sup>	ICLR2019	19.13	1.07	6.151	0.287
+ TA <sup>21</sup>	WACV2021	19.01	1.02	4.884	0.273
+ CA <sup>8</sup>	CVPR2021	12.99	0.46	6.152	0.276
+ SA <sup>36</sup>	ICASSP2021	14.17	0.55	4.881	0.267
+ SimAM <sup>35</sup>	ICML2021	14.10	0.56	4.880	0.266
+ EMA <sup>22</sup>	ICASSP2023	13.56	0.50	4.917	0.322
+ SDAA (Ours)	-	<b>12.56</b>	<b>0.43</b>	6.561	0.287

**SSDlite320.** When using different channel-spatial attention methods to replace the original SE in the backbone network MobileNetV3-Large for SSDlite, the integration of BAM, FAB, SA, EMA or SDAA improved the performance of object detection by a margin. Compared to BAM, FAB, SA and EMA, SDAA exceeded the baseline of SSDlite320 by performance gaining. Meanwhile, our SDAA outperformed SE by 0.1% in terms of AP, 0.4% in terms of AP@50 and underperformed SE by 0.3% in terms of AP@75.

**RetinaNet50.** When using different channel-spatial attention methods to replace the original SE in the backbone network ResNet50 for RetinaNet50, the integration of CBAM, CA, SA or SDAA improved the performance of object detection by a margin. Our SDAA outperformed SE by 0.2% in terms of AP, 1.3% in terms of AP@50 and underperformed SE by 0.6% in terms of AP@75.

In summary, on the same object detection dataset PASCAL VOC 2007 test, our SDAA method demonstrated greater stability compared to the existing channel-spatial attention methods when integrated into different objection detectors. It was worth noting that our SDAA method did not achieve the optimal AP in large object detection, specifically AP@75.

Table 4. Comparison results between different attention methods on PASCAL VOC 2007 test.

Model	Year	AP (%) $\uparrow$	AP@50 (%) $\uparrow$	AP@75 (%) $\uparrow$	Param. (M) $\downarrow$	FLOPs (G) $\downarrow$
SSDlite320 <sup>9</sup> + none	ICCV2019	14.2	28.4	12.3	1.650	0.463
+ SE <sup>10</sup>	CVPR2018	14.6	28.9	<b>13.3</b>	2.466	0.465
+ CBAM <sup>34</sup>	ECCV2018	12.8	26.3	10.8	2.467	0.466
+ BAM <sup>23</sup>	BMVC2018	14.5	28.3	13.2	4.728	1.056
+ FAB <sup>30</sup>	ECCV2018	14.4	28.6	12.7	2.470	0.675
+ GALA <sup>19</sup>	ICLR2019	9.7	20.9	7.7	2.881	0.571
+ TA <sup>21</sup>	WACV2021	6.1	14.1	4.5	1.652	0.474
+ CA <sup>8</sup>	CVPR2021	13.9	27.7	12.1	2.882	0.487
+ SA <sup>36</sup>	ICASSP2021	14.6	<b>29.3</b>	12.7	1.652	0.465
+ SimAM <sup>35</sup>	ICML2021	13.5	26.9	12.2	1.650	0.463
+ EMA <sup>22</sup>	ICASSP2023	14.5	29.1	12.7	2.671	1.424
+ SDAA (Ours)	-	<b>14.7</b>	<b>29.3</b>	13.0	3.282	0.513
RetinaNet50 <sup>18</sup> + none	ICCV2017	17.1	32.5	15.9	14.542	25.035
+ SE <sup>10</sup>	CVPR2018	18.1	33.8	17.2	15.799	25.046
+ CBAM <sup>34</sup>	ECCV2018	17.1	33.4	15.1	15.801	25.052
+ BAM <sup>23</sup>	BMVC2018	15.5	30.3	14.1	17.870	26.199
+ FAB <sup>30</sup>	ECCV2018	16.7	31.9	15.9	15.808	25.744
+ GALA <sup>19</sup>	ICLR2019	6.8	15.3	5.0	16.444	25.402
+ TA <sup>21</sup>	WACV2021	13.5	28.8	10.9	14.547	25.083
+ CA <sup>8</sup>	CVPR2021	18.1	35.0	<b>17.5</b>	16.445	25.086
+ SA <sup>36</sup>	ICASSP2021	17.9	34.3	16.7	14.545	25.059
+ SimAM <sup>35</sup>	ICML2021	13.8	27.0	12.8	14.542	25.035
+ EMA <sup>22</sup>	ICASSP2023	16.2	31.0	14.9	15.332	28.317
+ SDAA (Ours)	-	<b>18.3</b>	<b>35.1</b>	16.6	17.057	25.144

## 5. Discussion

Our SDAA method essentially decomposes a complex task into several simple and same subtasks. Moreover, the original task and each subtask are homomorphic. Our SDAA method then processes these subtasks separately, and finally aggregates the results of these subtasks processing.

The limitation of our SDAA method is losing H-directional and W-directional interaction information. This limitation may be addressed by means of adding an H-directional and W-directional interaction sub-unit and is worth studying in the future.

## 6. Conclusion








In this paper, we propose SDAA method, an channel-spatial attention method, which is aimed at simultaneously building channel correlations and long-range position correlations with assistance of the spatial decomposition and spatial aggregation. Experimental results show that our SDAA method consistently outperforms the existing attention methods applied into different models across different datasets.

We recommend exploring the possibility of a universal attention framework that leverages all types of attention mechanisms. For instance, a soft selection mechanism, also known as branch attention, can select from channel attention and spatial attention based on the task at hand. Attention mechanisms are inspired by the human visual system and represent a significant step towards building computer vision systems that can be interpreted. Although attention-based models are often explained through attention maps, these maps only provide a general sense of how the model is operating, rather than a precise understanding. However, in applications where security and safety are critical, such as medical diagnostics and automated driving systems, more rigorous requirements are necessary. In such areas, it is essential to better understand how these models work, including their failure modes. Developing attention models that are more interpretable and characterizable can increase their applicability.

### Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No. 62076117) and the Jiangxi Key Laboratory of Smart City (Grant No. 20192BCD40002).

### ORCID

Meng Zhu  <https://orcid.org/0000-0002-4900-8973>  
Weidong Min  <https://orcid.org/0000-0003-2526-2181>  
Hongyue Xiang  <https://orcid.org/0009-0008-9105-5440>  
Cheng Zha  <https://orcid.org/0000-0002-2390-1552>  
Zheng Huang  <https://orcid.org/0009-0003-2994-8030>  
Longfei Li  <https://orcid.org/0009-0003-2602-8941>  
Qiyang Fu  <https://orcid.org/0000-0002-9151-2815>

### References

1. D. R. Barr and E. T. Sherrill, Mean and variance of truncated normal distributions, *Am. Stat.* **53**(4) (1999) 357–361.
2. L. Chen *et al.*, Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2017), pp. 5659–5667.
3. X. Chen *et al.*, Symbolic discovery of optimization algorithms, preprint (2023), arXiv:2302.06675.
4. M. Everingham and J. Winn, The pascal visual object classes challenge 2012 (voc2012) development kit, *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep.* 8, Springer, Berlin, Germany (2011).
5. D. Gai *et al.*, Spatiotemporal learning transformer for video-based human pose estimation, *IEEE Trans. Circuits Syst. Video Technol.* **33** (2023) 4564–4576.
6. J. Gehring *et al.*, Convolutional sequence to sequence learning, in *Proc. Int. Conf. Machine Learning*, Vol. 70 (PMLR, 2017), pp. 1243–1252.

7. X. Glorot and Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in *Proc. Int. Conf. Artificial Intelligence and Statistics*, Vol. 9 (PMLR, 2010), pp. 249–256.
8. Q. Hou, D. Zhou and J. Feng, Coordinate attention for efficient mobile network design, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition* (IEEE, 2021), pp. 13713–13722.
9. A. Howard *et al.*, Searching for mobilenetv3, in *Proc. IEEE/CVF Int. Conf. Computer Vision* (IEEE, 2019), pp. 1314–1324.
10. J. Hu, L. Shen and G. Sun, Squeeze-and-excitation networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2018), pp. 7132–7141.
11. W. Hua, Z. Dai, H. Liu and Q. Le, Transformer quality in linear time, in *Proc. Int. Conf. Machine Learning*, Vol. 162 (PMLR, 2022), pp. 9099–9117.
12. S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *Proc. Int. Conf. Machine Learning*, Vol. 37 (PMLR, 2015), pp. 448–456.
13. M. Jaderberg, K. Simonyan, A. Zisserman and K. Kavukcuoglu, Spatial transformer networks, in *Proc. Advances in Neural Information Processing Systems*, Vol. 28 (Curran Associates, 2015), pp. 2017–2025.
14. D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, in *Proc. Int. Conf. Machine Learning* (PMLR, 2015) pp. 1–15.
15. A. Krizhevsky and G. E. Hinton, Learning multiple layers of features from tiny images, Technical Report TR-2009, University of Toronto, Toronto (2009).
16. A. Krizhevsky, I. Sutskever and G. E. Hinton, Imagenet classification with deep convolutional neural networks, in *Proc. Advances in Neural Information Processing Systems*, Vol. 25 (Curran Associates, 2012), pp. 1097–1105.
17. X. Li, W. Wang, X. Hu and J. Yang, Selective kernel networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2019), pp. 510–519.
18. T. Y. Lin *et al.*, Focal loss for dense object detection, in *Proc. IEEE Int. Conf. Computer Vision* (IEEE, 2017), pp. 2980–2988.
19. D. Linsley, D. Shiebler, S. Eberhardt and T. Serre, Learning what and where to attend with humans in the loop, in *Proc. Int. Conf. Learning Representations* (OpenReview, 2019), pp. 1–21.
20. J. Long, E. Shelhamer and T. Darrell, Fully convolutional networks for semantic segmentation, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2015), pp. 3431–3440.
21. D. Misra, T. Nalamada, A. U. Arasanipalai and Q. Hou, Rotate to attend: Convolutional triplet attention module, in *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision* (IEEE, 2021), pp. 3139–3148.
22. D. Ouyang *et al.*, Efficient multi-scale attention module with cross-spatial learning, in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing* (IEEE, 2023), pp. 1–5.
23. J. Park *et al.*, Bam: Bottleneck attention module, in *Proc. British Machine Vision Conf. (BMVA)*, 2018, pp. 147.
24. A. Paszke *et al.*, Pytorch: An imperative style, high-performance deep learning library, in *Proc. Advances in Neural Information Processing Systems* (Curran Associates, 2019), pp. 8026–8037.
25. I. Radosavovic *et al.*, Designing network design spaces, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition* (IEEE, 2020), pp. 10428–10436.

26. M. Tan and Q. V. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in *Proc. Int. Conf. Machine Learning*, Vol. 97 (PMLR, 2019), pp. 6105–6114.
27. R. Tian and A. P. Parikh, Amos: An adam-style optimizer with adaptive weight decay towards model-oriented scale, preprint (2022), arXiv:2210.11693.
28. A. Toshev and C. Szegedy, Deeppose: Human pose estimation via deep neural networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2014), pp. 1653–1660.
29. A. Vaswani et al., Attention is all you need, in *Proc. Advances in Neural Information Processing Systems*, Vol. 30 (Curran Associates, 2017), pp. 5998–6008.
30. C. Wang et al., Mancs: A multi-task attentional network with curriculum sampling for person re-identification, in *Proc. Eur. Conf. Computer Vision* (Springer, 2018), pp. 365–381.
31. F. Wang et al., Residual attention network for image classification, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2017), pp. 3156–3164.
32. H. Wang et al., Axial-deeplab: Stand-alone axial-attention for panoptic segmentation, in *Proc. Eur. Conf. Computer Vision* (Springer, 2020), pp. 108–126.
33. Q. Wang et al., Dual similarity pre-training and domain difference encouragement learning for vehicle re-identification in the wild, *Pattern Recognit.* **139** (2023) 109513.
34. S. Woo et al., Cbam: Convolutional block attention module, in *Proc. Eur. Conf. Computer Vision* (Springer, 2018) pp. 3–19.
35. L. Yang, R. Y. Zhang, L. Li and X. Xie, SimAM: A simple, parameter-free attention module for convolutional neural networks, in *Proc. Int. Conf. Machine Learning*, Vol. 139 (PMLR, 2021), pp. 11863–11874.
36. Q. L. Zhang and Y. B. Yang, Sa-net: Shuffle attention for deep convolutional neural networks, in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing* (IEEE, 2021), pp. 2235–2239.
37. H. Zhao et al., Pyramid scene parsing network, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (IEEE, 2017), pp. 2881–2890.



**Meng Zhu** received B.E. and M.E. degrees in Computer Science and Technology from the Nanchang University, China in 2018 and 2021, respectively. He is currently pursuing his Ph.D. degree at the Nanchang University, China. His current

research interests include computer vision, and natural language processing.



**Weidong Min** received B.E., M.E. and Ph.D. degrees in Computer Application from the Tsinghua University, China in 1989, 1991 and 1995, respectively. He is currently Professor and Dean, Institute of Meta-verse, the Nanchang University, China. He is

Executive Director of China Society of Image and Graphics. His current research interests include image and video processing, artificial intelligence, big data, distributed system, and smart city information technology.



**Hongyue Xiang** received B.E. degree in Computer Science and Technology from the Hangzhou Dianzi University, China in 2018 and the M.E. degree in Software Engineering from the Nanchang University, China in 2021, respectively. She is currently pursuing Ph.D. degree at the Nanchang University, China. Her current research interests include image encryption, and deep learning.



**Cheng Zha** received M.E. degree in Computer Technology from the Shanghai Ocean University, China in 2020. He is currently pursuing Ph.D. degree at the Nanchang University, China. His current research interests include computer vision, and ship detection.



**Zheng Huang** received B.E. degree in School of Civil Engineering and Communication from the North China University of Water Resources and Electric Power, China, in 2021. He is currently pursuing his M.E. degree at the Nanchang University, China.

His current research interests include computer vision, and deep learning.



**Longfei Li** received the B.E. degree in Software Engineering from Jiangxi Normal University, China in 2021. He is currently pursuing M.E. degree at the Nanchang University, China. His current research interests include computer vision, and vehicle re-identification.



**Qiyan Fu** received M.E. degree in Electronic and Communication Engineering from the Nanchang University, China in 2017. She is currently pursuing Ph.D. degree at the Nanchang University, China. Her current research interests focus on artificial intelligence, and computer vision.