MCA: Moment Channel Attention Networks

Yangbo Jiang[1,2], Zhiwei Jiang[3], Le Han[1,2], Zenan Huang[1,2], Nenggan Zheng[1,2,4,5]*

[1]Qiushi Academy for Advanced Studies, Zhejiang University, Hangzhou, Zhejiang, China
[2]College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang, China
[3]Guangzhou Electronic Technology Co., Ltd., Chinese Academy of Sciences, GuangZhou, China
[4]State Key Lab of Brain-Machine Intelligence, Zhejiang University, Hangzhou, Zhejiang, China
[5]CCAI by MOE and Zhejiang Provincial Government(ZJU), Hangzhou, Zhejiang, China
{jiangyangbo, hanle, lccurious}@zju.edu.cn, j_z_w@163.com, zng@cs.zju.edu.cn

Abstract

Channel attention mechanisms endeavor to recalibrate channel weights to enhance representation abilities of networks. However, mainstream methods often rely solely on global average pooling as the feature squeezer, which significantly limits the overall potential of models. In this paper, we investigate the statistical moments of feature maps within a neural network. Our findings highlight the critical role of high-order moments in enhancing model capacity. Consequently, we introduce a flexible and comprehensive mechanism termed Extensive Moment Aggregation (EMA) to capture the global spatial context. Building upon this mechanism, we propose the Moment Channel Attention (MCA) framework, which efficiently incorporates multiple levels of moment-based information while minimizing additional computation costs through our Cross Moment Convolution (CMC) module. The CMC module via channel-wise convolution layer to capture multiple order moment information as well as cross channel features. The MCA block is designed to be lightweight and easily integrated into a variety of neural network architectures. Experimental results on classical image classification, object detection, and instance segmentation tasks demonstrate that our proposed method achieves state-of-the-art results, outperforming existing channel attention methods.

Introduction

Channel attention mechanisms have aroused wide concern in the field of computer vision due to their remarkable performance on various tasks, including image classification(Hu, Shen, and Sun 2018; Woo et al. 2018), object detection(Dai et al. 2017; Carion et al. 2020), instance segmentation(Yuan et al. 2018; Fu et al. 2019), face recognition(Yang et al. 2017; Wang et al. 2020b), image generation (Gregor et al. 2015; Zhang et al. 2019; Liu et al. 2022) and multi-modal learning (Su et al. 2019; Xu et al. 2018), etc. The objective of channel attention method is to learn the aggregation of global spatial information and adaptively recalibrate the weight of each channel. As an effective method, the channel attention approach has become an essential module in various neural networks and has been widely employed to enhance the representational power of network.

*Corresponding author

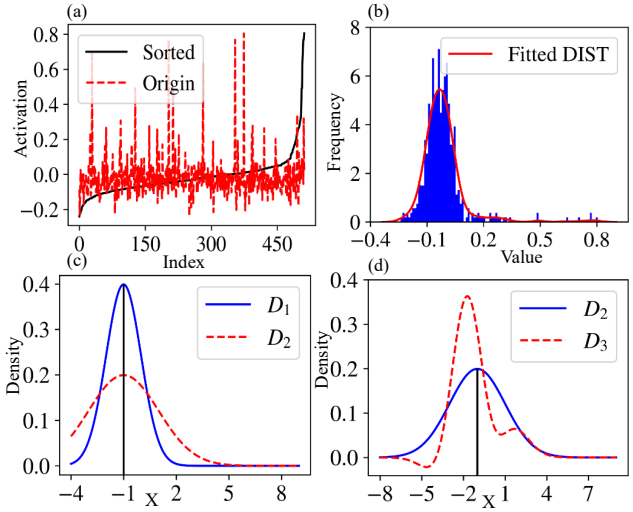

Figure 1: The activation of feature map exhibits a distinct probability distribution, as illustrated in (a) and (b). While the first-order moment is inadequate to represent a standard Gaussian distribution in (c), and combining the first and second-order moment falls short in capturing non-Gaussian distributions (as depicted in (d)). Extensive moment aggregation mechanism offers a viable solution to this challenge.

In recent years, several effective channel attention methods have been presented. Hu et al. (Hu, Shen, and Sun 2018) introduced the Squeeze-and-Excitation Networks (SENet) with squeeze and excitation modules. The squeeze module maps global spatial information with global average pooling (GAP), while the excitation module recalibrates the weights of each channel based on their channel-wise relationships. Since then, other channel attention algorithms (Lee, Kim, and Nam 2019; Wang et al. 2020a; Ruan et al. 2021) have achieved significant improvements to both the squeeze and excitation modules. From a statistical perspective, the squeeze module of SENet (Hu, Shen, and Sun 2018), ECANet (Wang et al. 2020a), and GCT (Ruan et al. 2021) utilize first-order statistics (GAP) to extract global information from the feature map without considering high-order statistics. Despite SRM (Lee, Kim, and Nam 2019)

Blocks	Squeeze Method		Excitation Method			Params	AP
	Pooling Method	Order	Method	Moments Fusion	Channel Interaction		
SE (CVPR,2018)	GAP	1ST	FC	✗	✗	$\frac{2}{r}\sum_{s=1}^{S} N_s \cdot C_s^2$	35.4
SRM (ICCV,2019)	GAP & STD	1ST & 2ND	CFC	✓	✗	$\sum_{s=1}^{S} N_s \cdot C_s * 6$	36.9
ECA (CVPR,2020)	GAP	1ST	Conv1D	✗	✓	$\sum_{s=1}^{S} N_s \cdot \lvert(log_2 C + 1)/2\rvert_{odd}$	36.2
GCT (CVPR,2021)	GAP	1ST	Gaussian	✗	✓	$\sum_{s=1}^{S} N_s$	37.9
MCA-E (Ours)	GAP & VAR	1ST & 2ND	CMC	✓	✓	$\sum_{s=1}^{S} N_s \cdot C_s * 2$	38.0
MCA-S (Ours)	GAP & SKEW	1ST & 3RD	CMC	✓	✓	$\sum_{s=1}^{S} N_s \cdot C_s * 2$	**38.3**

Table 1: Comparison with other SOTA channel attention blocks on COCO val2017 set. Our proposed block is distinguished by two key features: (i) a flexible and comprehensive moment aggregation approach for pooling; (ii) an novel excitation method that serves two efficiencies: moment fusion and local channel interaction. Here, FC, CFC, and CMC refer to fully connected, channel-wise fully connected, and cross moment conv1D layer, respectively. While Gaussian denotes Gaussian function. C denotes the number of channels. r denotes the reduction ratio of SE. $|\cdot|$ odd indicates the nearest odd number of $\cdot$.

involving the utilization of both first-order statistics (GAP or Mean) and second-order statistics (Standard Deviation, STD) for spatial representation, it merely concatenates them with a basic multiply and addition operation. However, the local channel interaction is absent in the methodology of SRM.

To dig deeper into the shortcomings of the squeeze module, we investigate the moment statistics within the probability distribution. The activation of feature map within a neural network can be regarded as samples from a probability distribution, as depicted in Fig. 1(a) and (b). Moments provide a method to represent this probability distribution by employing the moment generating function (Casella and Berger 2002). These moments are derived from the Taylor series expansion of the expectation of a random variable (Athreya and Lahiri 2006; Garthwaite, Jolliffe, and Jones 2002). Specifically, for a Gaussian distribution, the first-order moment corresponds to the Mean (μ), and the second-order moment corresponds to the Variance (VAR, σ^2), and the higher order moments is zero or constant (Patel and Read 1996; Bai and Ng 2005). From a statistical perspective, as shown in the moment generate function, other order moment, except for the first-order moment, second and third-order moment pay a crucial role in represent a probability distribution. For example, in Fig. 1(c), the probability distribution D_1 and D_2 both have the same mean of -1, but the former has a lower STD of 1 compared to the latter's 2. A smaller STD indicates a more concentrated distance from the Mean, while the Mean reflects the degree of concentration of the data. Similarly, in Fig. 1(d), the D_2, being a Gaussian distribution, has a lower SKEW (third-order moment) of 0 compared to the D_3's 2, and the SKEW indicates the asymmetry of the distribution. Therefore, a flexible multiple moment aggregation method is crucial for effectively representing the feature map of networks.

In this paper, we propose a Moment Channel Attention (MCA) method inspired by moment representation in the probability distribution. The MCA method employs Extensive Moment Aggregation (EMA) mechanism as the pooling method for a refined aggregation of extensive moment information. Furthermore, we introduce the Cross Moment Convolution (CMC) method via channel-wise convolution layer to capture the interaction between moments of different orders in the feature map, as well as local cross channels feature. Based on EMA mechanism with base term and other terms, we present two MCA methods: MCA-E and MCA-S. As shown in Table 1, MCA-E and MCA-S slightly increase the model size. Despite this slight increase in computational cost, the accuracy of the model is effectively improved. MCA-E and MCA-S achieve excellent performance compared to other SOTA methods. Moreover, MCA method is easily integrated into network architectures, such as ResNets (He et al. 2016; Xie et al. 2017), and lighter architectures like ShuffleNet (Ma et al. 2018), Inception (Ioffe and Szegedy 2015). Furthermore, MCA can be trained end-to-end, making it convenient and flexible for practical applications. In general, the main contributions of this work are summarized as three-fold:

- In this paper, We regard the activation of feature map as samples following a probability distribution and analyzing the validity of each order moment. Consequently, we introduce the EMA mechanism, revealing the effectiveness of utilizing a constrained set of moments to accurately represent the global spatial features. Building upon this aggregation strategy, we propose the MCA method, which captures feature from multiple moments. Notably, the ECA method can be viewed as a specific instance of our proposed method.
- Furthermore, we introduce the CMC method to moment fusion. The CMC method employs a concise and powerful channel-wise convolution layer that merges information from multiple order moments within the feature map, while concurrently incorporating local cross-channel interactions.
- Compared with existing channel attention methods, our proposed method achieves excellent state-of-the-art results on the COCO benchmarks for object detection and instance segment, as well as the ImageNet classification benchmark. In addition, our method is a lightweight and convenient, with computational and model complexity comparable to ECA method.

Related Work

Attention Mechanisms Attention mechanism is an effective way to enhance the representational power of neural

networks, which can be typically categorized into channel attention (Hu, Shen, and Sun 2018; Gao et al. 2019; Lee, Kim, and Nam 2019; Wang et al. 2020a; Yang et al. 2020; Qin et al. 2021; Ruan et al. 2021), spatial attention (Wang et al. 2018; Chen et al. 2018; Cao et al. 2019; Li et al. 2019), and channel & spatial attention (Park et al. 2018; Woo et al. 2018; Roy, Navab, and Wachinger 2018). We briefly review channel attention methods firstly, which aims to recalibrate the weights of channels to facilitate the representation ability. SENet (Hu, Shen, and Sun 2018) recalibrates feature maps by capturing channel-wise dependencies, which has become a paradigm of channel attention. ECANet (Wang et al. 2020a) uses 1D convolution to capturing local cross-channel interaction while alleviating the negative impact of channel dimension reduction, and SRM (Lee, Kim, and Nam 2019) combines an attention mechanism with style pooling to extract style information from each channel and then recalibrates the weights of each channel through channel-independent style integration. The GCT (Ruan et al. 2021) employs a Gaussian function to directly map global contexts to attention activation, which significantly improves the generalization of the model.

On the other hand, spatial attention can be interpreted as an adaptive spatial region selection mechanism. For example, NLNet (Wang et al. 2018) utilizes non-local spatial structure, which can be viewed as a form of self-attention mechanism(Vaswani et al. 2017), to capture long-range dependencies. A2-Net (Chen et al. 2018) leverages the same structure as NLNet capture long-range relation, but with different details in dimension and computation process. Meanwhile, GcNet (Cao et al. 2019) simplifies the non-local block and incorporates an excitation block from SENet (Hu, Shen, and Sun 2018) to enhance channel interaction. Lastly, SKNet (Li et al. 2019) utilizes selective kernel unit to capture multiple scale feature information for channel representation.

Additionally, there are methods that combine both previous two ways. For instance, BAM (Park et al. 2018) and CBAM (Woo et al. 2018) integrate both channel and spatial attention mechanisms and the previous method use an additive operator to fuse the two blocks, while the latter uses a cascaded operator for integration.

Moment Statistics for Deep Learning Moment statistics is a potent tool for computer vision, including image moment (Flusser 2000; Flusser and Suk 2006), statistical moment (Gonzalez and Woods 2008a), color moment (Smeulders et al. 2000; Yu et al. 2002), and moment texture approaches (Gonzalez and Woods 2008b), etc. In the image moment and color moment method, multiple moments are used to represent the image and color channels. Notably, the integration of moment estimation into deep learning has gained traction in recent years. Moment estimation method is applied for optimization algorithm (Kingma and Ba 2015; Tong, Liang, and Bi 2022; Zou et al. 2019) and self-supervised learning (Li, Liu, and Sun 2021). Moreover, Zellinger et al. introduce a novel regularization technique, Central Moment Discrepancy (CMD), to facilitate domain adaptation from source to target domains (Zellinger et al. 2017).

The Proposed Approach

In this section, we begin by introducing a novel moment aggregation mechanism based on probability distribution. Subsequently, we present our moment channel attention method.

Extensive Moment Aggregation Mechanism

We define the moment representation as an empirical estimate of the moment representation. Only the moments that correspond to the marginal distributions are computed.

Definition 1 (Extensive Moment Aggregation) *Let X be bounded random samples with probability distribution p. The extensive moment aggregation EMA_K is defined as an empirical estimate of the moment representation, by*

$$EMA_K(p) = \alpha_1 \left\| E(X) \right\|_2 + \sum_{k=2}^{K} \alpha_k \left\| M_k(X) \right\|_2 , \quad (1)$$

where $E(X)$ is the empirical expectation on the sample X and $M_k(X) = E((X - E(X))^k)$ encompasses all k-th order central moments of X. And K serves as a parameter that restricts the quantity of moments computed, $\alpha_k \in (0, 1)$ act as weighted parameters.

The bound on the order of moment terms is limited by the parameter K. Proposition 1 demonstrates that the marginal distribution moment terms have an upper bound that strictly decrease with increasing moment order. The theoretical analysis in later section shows that the marginal utility of higher-order terms fade as the order increases. Specifically, within this framework, we can always find a best trade-off between distribution approximation and computational efficiency.

Proposition 1 (Upper Bound) *Let X be bounded random vector with probability distribution p on compact interval $[a, b]^N$. Then, for all positive integers k,*

$$\alpha_k \left\| M_k(X) \right\|_2 \leq \sqrt{N} \left(\frac{1}{k+1} (\frac{k}{k+1})^k + \frac{1}{2^{1+k}} \right). \quad (2)$$

where $M_k(X) = E((x - E(X))^k)$ is the vector of all k-th order moments of the marginal distributions of p. And $\alpha_k \in (0, 1)$ act as weighted parameters.

Proof 1 *Let $X([a, b])$ denote the set of all random variables with values in the interval $[a, b]$. Then, it follows that*

$$\begin{aligned} \alpha_k \left\| M_k(X) \right\|_2 &\leq \left\| \mathbb{E} \left(\left(\left| \frac{X - \mathbb{E}(X)}{b - a} \right| \right)^k \right) \right\|_2 \\ &\leq \sqrt{N} \sup_{X \in \mathcal{X}([a,b])} \mathbb{E} \left(\left| \frac{X - \mathbb{E}(X)}{b - a} \right|^k \right). \end{aligned} \quad (3)$$

Here, the latter term refers to the absolute moment of order k, for which the smallest upper bound is known (Egozcute et al. (Egozcue et al. 2012)):

$$\alpha_k \left\| M_k(X) \right\|_2 \leq \sqrt{N} \sup_{X \in \mathcal{X}([0,1])} x(1 - x)^k + (1 - x)x^k. \quad (4)$$

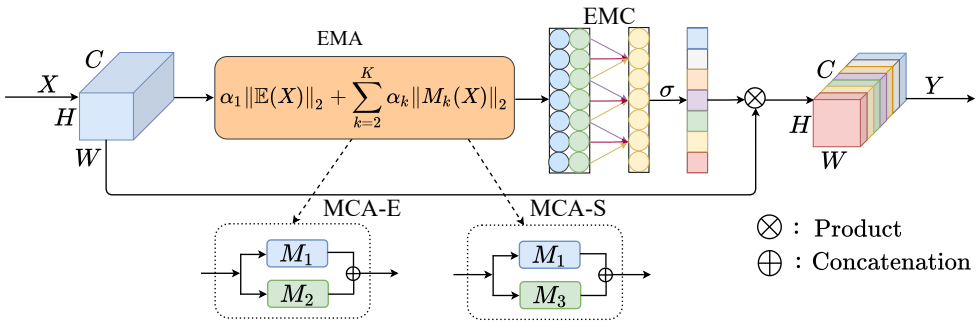

Figure 2: Diagram of the moment channel attention (MCA) networks. We design a extensive moment aggregation (EMA) mechanism to capture global spatial feature, while the cross moment convolution (CMC) method facilitates cross-channel interactions between lower-order and other-order moments as well as different channels. The parameter α denotes as a learnable factor for each moment and σ denotes Sigmoid function. K indicates the order of moments, for MCA-E and MCA-S, K is set to 2.

Then, we can get a more explicit upper bound:

$$\alpha_k \left\| M_k(X) \right\|_2 \leq \sqrt{N} \left(\frac{1}{k+1} (\frac{k}{k+1})^k + \frac{1}{2^{1+k}} \right). \quad (5)$$

The proof of theorem and proposition can also refer to Zellinger et al. (Zellinger et al. 2017, 2019) for detail.

The equation presented in Eq. (1) offers a comprehensive formalization of moment information aggregation, and Eq. (2) demonstrates that a constrained set of moments can effectively represent the distribution. By controlling the parameter K, we can readily balance between distribution approximation accuracy and computational efficiency.

We apply the extensive moment aggregation for the global feature extraction of channel attention mechanism and propose a novel attention method.

Moment Channel Attention Block

In this section, we present the architecture of MCA in detail. The MCA method comprises three modules: (a) moment aggregation; (b) moment fusion; (c) channel recalibration. The overall framework of MCA is illustrated in Fig. 2.

Moment Aggregation From extensive moment aggregation EMA_K, it is evident that moments consist of first-order moment M_1, other-order moments along with their combination. So for M_1, $M_1 = E(X)$; and for other moments, $M_k(X) = E((X - E(X))^k)(k \geq 2)$.

As showed in EMA_K, The moment aggregation module can obtain single moment: the first-order moment M_1 or the second-order moment M_2 or the third-order moment M_3 for each channel. so the M_{mono} is represented as follows:

$$M_{mono} = [M_1 | M_2 | M_3]\,. \quad (6)$$

M_{mono} is a refined representation for this channel and $M_{mono} \in \mathbb{R}^{1\times1\times C}$. M_{mono} will out to the next layer as the basis for weight recalibration.

In the same way, the M_{dual} module combines two of M_1, M_2 and M_3 as the aggregation of global feature and $M_{dual} \in \mathbb{R}^{1\times2\times C}$. The M_{dual} is represented as follows:

$$M_{dual} = [M_1 \wedge M_2] || [M_1 \wedge M_3] || [M_2 \wedge M_3]\,. \quad (7)$$

In addition, the M_{triple} obtains M_1, M_2 and M_3 for each channel as the representation of the channel and $M_{triple} \in \mathbb{R}^{1\times3\times C}$. The specific calculation of M_{triple} is as follows:

$$M_{triple} = [M_1 \wedge M_2 \wedge M_3]\,. \quad (8)$$

To summarize, the steps of M_{mono}, M_{dual}, and M_{triple} represent the detail implementation of extensive moment aggregation. Both M_{dual} and M_{triple} result in an increase in dimension, which is a matter we address in the subsequent section.

Moment Fusion Next, we delve into moment fusion approach aimed at resolving the dimension issue associated with extensive moment aggregation EMA_K and this method effectively captures both the low-order and other-order moment information across the channel.

Note that the dimension of M_{mono}, M_{dual}, and M_{triple} are different. How to fuse the moment vector is a challenge. In the SENet method and the ECANet method, the dimension of the vector is also $1 \times 1 \times C$, and they are processed by a fully connected (Hu, Shen, and Sun 2018) layer and a Conv1D (Wang et al. 2020a) layer, respectively. In the SRM method, the dimension of input is $1 \times 2 \times C$, they use the channel-wise fully connected (CFC) (Lee, Kim, and Nam 2019) layer for fusion. During this study, we employ cross moment convolution (CMC) method, which utilizes channel-wise 1D convolution layer for fusion multiple moments. The input vectors correspond to the output of either M_{mono}, M_{dual}, or M_{triple}., and the output $F \in \mathbb{R}^{1\times1\times C}$. the moment fusion is formulated as:

$$F = \text{CMC}(M). \quad (9)$$

The CMC method fuses information cross the channel through the convolutional kernel, and can fuse different orders moment information in the channel.

Channel Recalibration Finally, through the above moment fusion method, we obtain the extensive moment feature aggregation. The entire algorithm can be represented in

a unified form. The entire MCA block of is expressed as follows:

$$Y = X \cdot \sigma(F). \tag{10}$$

Among them, $X \in \mathbb{R}^{C \times H \times W}$ be an activation feature in a convolutional network, where C, H, and W denote the number of channels, the spatial height and width, respectively. And M can be M_{mono}, M_{dual} or M_{triple}. The dimension of CMC is adjusted according to the dimension of Moment.

Theoretical analysis of extensive moment aggregation indicates that the performance improves as the moment order increase. This observation is further validated by the results of ablation studies. Specifically, M_2 and M_3 demonstrates better than M_1, and M_{dual} outperforms M_{mono}. While M_{triple} exhibits superiority over M_{dual}. It is noteworthy that M_{triple} slightly outperforms performance to $[M_1 \wedge M_3]$. It can be seen that the extensive moment aggregation is an effective way of extracting representative information.

In this paper, we conduct the method using an extensive moment aggregation manner, fundamental term and additional terms. Additionally, computational complexity and performance are also considerations. Taking these factors into account, we have chosen two methods, namely Moment-E and Moment-S, for further investigation. In the Moment-E method, we adopt $M = [M_1 \wedge M_2]$, while in the Moment-S method, we utilize $M = [M_1 \wedge M_3]$. Subsequent experiments and analyses are conducted based on the aforementioned two algorithms.

Parameter and Computational Complexity

This section analyzes the parameter and computational complexity of the MCA algorithm. The number of parameters for MCA(MCA-E, MCA-S) is $\sum_{s=1}^{S} N_s \cdot C_s * 2$, where S denotes the numbers of stages, N_s is the number of repeated blocks in s-th stage, and C_s is the dimension of the output channels for s-th stage. While the extra parameters of SENet is $\frac{2}{r}\sum_{s=1}^{S} N_s \cdot C_s^2$, where r is its reduction ratio. For instance, given ResNet-50 as a baseline for image classification, MCA-E and MCA-S require only 6.06K additional parameters whereas SENet requires 2.53M. MCA also introduces negligible extra computations to the original architecture. A single forward pass of a 224×224 pixel image for MCA-S requires additional 0.011 GFLOPs to ResNet-50 which requires 4.122 GFLOPs. By adding only 0.27% relative computational burden, MCA-E increases the Top-1 accuracy from 74.97% to 76.61%, which indicates that MCA offers a good trade-off between accuracy and efficiency.

Experiments

In this section, we evaluate the performance of the MCA method on the COCO and ImageNet datasets, along with other channel attention methods, such as SENet (Hu, Shen, and Sun 2018), SRM (Lee, Kim, and Nam 2019), ECANet (Wang et al. 2020a), and GCT method (Yang et al. 2020). To ensure a fair comparison, we use the same network architecture, data augmentation strategy, and optimization parameters for all methods without any additional bells and whistles.

Implementation Details

We evaluate the effectiveness of our approach in object detection and instance segmentation tasks using the COCO (Lin et al. 2014) dataset. The training of all models is performed on 8 GPUs, with each mini-batch consisting of eight images. As part of the preprocessing procedure, the shorter edge of the input image is resized to 800 while the longer edge is constrained to a maximum size of 1333 and RandomFlip is used for data augmentation. In terms of optimization, stochastic gradient descent (SGD) was chosen as the optimizer, with a weight decay of 1e-4, and a momentum of 0.9. All models are trained for a total of 12 epochs. The learning rate is set to 0.02 initially and drops to 0.002 and 0.0002 at the 8th and 11th epochs, respectively. For the object detection and instance segmentation tasks, we utilize the mean average precision (AP), $AP_{0.5}$, $AP_{0.75}$, AP_S, AP_M, and AP_L as the evaluation metrics. Additionally, we also test the storage capacity by Params, as well as the computing efficiency in GFLOPs.

Next, we further evaluate the MCA method for image classification tasks on the ImageNet dataset (Russakovsky et al. 2015). All models are trained with a mini-batch size of 256 using 8 GPUs (32 images per GPU). The size of the image is 224×224, and we use RandomFlip in the horizontal direction for data augmentation. In terms of optimization, SGD was selected as the optimizer, with a weight decay of 1e-4, and a momentum of 0.9. The learning rate is initial to 0.1. The model is trained for 100 epochs and the learning rate is decreased by a factor of 0.1 at epochs 30, 60, 90. We insert the attention module after the third convolutional layer of each residual module. For the image classification experiment, we use Top-1 and Top-5 accuracy as the test metrics. Additionally, we evaluate the storage capacity with Params and computing efficiency in terms of GFLOPs. Our research is conducted using PyTorch 1.8.2 and MindSpore 1.7.0 (Huawei Technologies Co. 2022). The source code for this project can be accessed on GitHub at https://github.com/CSDLLab/MCA.

Ablation Study

First, we perform ablation analysis to investigate the effects of the moments selection and coverage of moment cross-channel interaction. The Faster RCNN (Ren et al. 2015) is utilized as the base detector for experimenting on the COCO dataset. The ResNet-50 is employed as the backbone model.

The Selection of Moments The aim of our proposed method is to conduct aggregation of extensive moment feature, so we conduct ablation experiments to explore the impact of moments on the performance, specifically on their selection. Moments include first-order moment M_1, second-order moment M_2, third-order moment M_3, and fusion versions include $M_1 + M_2$, $M_1 + M_3$, $M_2 + M_3$, and $M_1 + M_2 + M_3$. Table 2 shows that M_{triple} $(M_1 + M_2 + M_3)$ has the best performance in the object detection tasks, followed by the MCA-S $(M_1 + M_2)$ method. Several key observations can be derived from Table 2. Firstly, higher-order moments exhibit superior performance compared to lower-order moments. Notably, M_3 outperforms both M_2 and M_1. Sec-

Methods			AP	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L
M_1	M_2	M_3						
			34.9	56.6	37.1	20	38.5	45.6
✓			36.2	58.4	38.6	21.1	39.1	45.9
	✓		37.6	59.4	40.6	21.9	41.0	48.3
		✓	38.1	60.0	41.4	23.1	42.1	48.8
✓	✓		38.0	60.0	40.7	22.4	42.1	48.2
✓		✓	38.3	**60.5**	41.4	22.6	**42.4**	49.4
	✓	✓	38.2	60.2	41.5	22.8	42.1	**49.5**
✓	✓	✓	**38.4**	60.7	**41.6**	**23.2**	42.2	49.3

Table 2: Ablation study of different moments on COCO val2017 set with object detection task and the baseline model is Resnet-50.

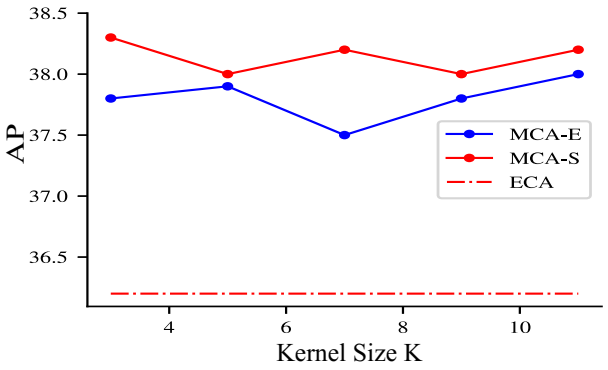

Figure 3: Results of MCA-E and MCA-S with various kernel size k for 3 to 11 using ResNet-50 as backbone model. Here we choose the ECA method as the baseline.

ondly, the combination of multiple moments yields superior results compared to using individual moment. For instance, the combination of M_1+M_2 performs better than employing either M_1 or M_2 alone. Additionally, the combination of M_1+M_3 surpasses the performance achieved by using M_1 or M_3 individually. Thirdly, M_{triple} demonstrates a clear advantage over M_{dual} and M_{mono} in terms of performance. Fourthly, complex combinations of moments M_{triple} show limited improvement. Although extensive moment aggregation has an upper bound k. Considering the computational complexity, we do not investigate further to the fourth moment Kurtosis (Bai and Ng 2005).

Coverage of Moment Cross-Channel Interaction The another aim of our MCA method is capturing moment cross-channel interaction appropriately, so the coverage of interaction (i.e., kernel size k of CMC method) needs to be determined. In our experiments, we set the kernel size k for MCA-E and MCA-S to range from 3 to 11 for simplicity. The results are depicted in Fig. 3, from which we have three observations. (i) MCA block with different k outperforms ECANet block, demonstrating the effectiveness of utilizing moments in channel attention. (ii) The performance of the MCA-E method has a zigzagging upward trend with slight fluctuations; the performance of the MCA-S method has a zigzagging downward trend with slight fluctuations. (iii) For MCA-E and MCA-S, the setting with 11 and 3 achieves the best performance, respectively. In this way, we use these settings in our block and other experiments.

Object Detection on COCO Dataset

We also conduct object detection experiments on the COCO dataset and compared the performance of our method with SENet, SRM, ECANet, and GCT methods. The experimental results are shown in Table 3. For object detection, we use three types of detectors, namely Faster RCNN (Ren et al. 2015), Mask RCNN (He et al. 2017), and RetinaNet (Lin et al. 2017). The ResNet-50 and ResNet-101 models are used as the backbone models.

Overall, the performance of MCA-E and MCA-S is superior to that of other attention blocks on the object detection task. In the Faster RCNN experiment, with ResNet50 as the backbone model, compared with the baseline, SENet, SRM, ECANet, and GCT methods, the MCA method exhibits AP improvements of 3.2%, 2.9%, 1.4%, 2.1%, and 0.4%, correspondingly; Subsequently, using employing ResNet-101 as the backbone model, the MCA method achieves superior AP compared to the baseline, SENet, SRM, ECANet, and GCT method, with improvement of 1.3%, 1.2%, 1.0%, 0.4%, and 0.2%, respectively. In the Mask RCNN experiment, utilizing ResNet-50 as the backbone model, compared with the baseline, SENet, SRM, ECANet, and GCT blocks, the MCA method improves the AP by 1.5%, 1.2%, 1.0%, 0.9%, and 0.5%, respectively; Similarly, with ResNet-101 as the backbone, the AP of MCA methods outperforms the baseline, SENet, SRM, ECANet, and GCT by 1.3%, 0.9%, 1.4%, 0.8%, and 0.2%, respectively. In the RetinaNet experiment, using ResNet-50 is the backbone model, compared with the baseline, SENet, SRM, ECANet, and GCT methods, the MCA method yields the AP by 1.1%, 1.4%, 0.3%, 1.0%, and 0.3%, respectively; with ResNet-101 as the backbone, the MCA method surpasses the performance of the baseline, SENet, SRM, ECANet, and GCT methods by 1.2%, 1.4%, 2.0%, 0.8%, and 0.5%, respectively.

Instance Segmentation on COCO Dataset

Next, we conduct instance segmentation experiments utilizing the COCO dataset, comparing proposed MCA method with other blocks including SENet, SRM, ECANet, and GCT method. we utilize Mask RCNN as the detector and select ResNet-50 and ResNet-101 models as the backbone models. Our proposed method outperforms the other channel attention method in terms of instance segmentation as the results shown in Table 5. (i) With the ResNet-50 as the backbone model, the MCA method enhances the performance of the baseline, SENet, SRM, ECA, and GCT methods by 1.4%, 1.2%, 1.2%, 0.9%, and 0.6%, respectively. (ii) Using ResNet-101 as the backbone model, the MCA method outperforms the baseline, SENet, SRM, ECA, and GCT methods by 1.1%, 0.8%, 1.0%, 0.6%, and 0.1%, respectively. These outcomes show the generalization and effectiveness of our MCA method.

Detector	Methods	Params	GFLOPs	AP	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L
Faster RCNN	ResNet-50	41.53M	207.07	34.9	56.6	37.1	20	38.5	45.6
	+SE (CVPR,2018)	+2.49M	207.18	35.4	57.4	37.7	20.8	39.1	45.9
	+SRM (ICCV,2019)	+0.06M	207.07	36.9	57.5	39.8	21.1	40.4	47.0
	+ECA (CVPR,2020)	+0.05k	207.18	36.2	58.4	38.6	21.1	41.8	49.0
	+GCT (CVPR,2021)	+0.01k	207.18	37.9	59.6	41.1	22.3	40.7	47.9
	+MCA-E (Ours)	+0.03M	207.07	38.0	60.0	40.7	22.4	42.1	**49.6**
	+MCA-S (Ours)	+0.03M	207.07	**38.3**	**60.5**	**41.4**	**22.6**	**42.4**	49.4
	ResNet-101	60.52M	283.14	39.0	60.2	42.1	22.2	43.0	51.0
	+SE (CVPR,2018)	+4.72M	283.33	39.1	60.5	42.3	22.1	43.1	51.3
	+SRM (ICCV,2019)	+0.13M	283.14	39.0	59.6	42.3	22.0	42.9	50.6
	+ECA (CVPR,2020)	+0.12k	283.32	39.9	61.3	43.4	22.7	44.2	52.1
	+GCT (CVPR,2021)	+0.03k	283.32	40.1	61.7	43.5	22.9	42.7	50.5
	+MCA-E (Ours)	+0.06M	283.14	**40.3**	61.9	**43.7**	23.5	44.4	**52.5**
	+MCA-S (Ours)	+0.06M	283.14	40.2	**62.3**	43.2	**23.6**	**44.6**	52.2
Mask RCNN	ResNet-50	44.17M	260.14	37.5	58.7	40.3	22.2	40.9	48.8
	+SE (CVPR,2018)	+2.49M	260.25	37.8	59.2	40.7	22.2	41.5	48.7
	+SRM (ICCV,2019)	+0.05M	260.14	38.0	58.7	41.6	21.9	41.3	49.8
	+ECA (CVPR,2020)	+0.05k	260.25	38.1	59.8	41.0	22.1	42.1	49.4
	+GCT (CVPR,2021)	+0.01k	260.25	38.5	60.1	41.8	22.4	42.4	49.7
	+MCA-E (Ours)	+0.03M	260.14	**39.0**	**61.0**	42.3	22.9	42.7	**50.5**
	+MCA-S (Ours)	+0.03M	260.14	**39.0**	60.9	**42.7**	**23.3**	**42.9**	**50.5**
	ResNet-101	63.16M	336.21	39.7	60.6	43.3	22.9	43.8	52.2
	+SE (CVPR,2018)	+4.72M	336.40	40.1	61.4	43.4	23.6	44.2	52.4
	+SRM (ICCV,2019)	+0.12M	336.21	39.6	60.6	43.0	22.4	43.8	52.4
	+ECA (CVPR,2020)	+0.17k	336.39	40.2	61.6	43.9	23.6	44.5	52.7
	+GCT (CVPR,2021)	+0.03k	336.39	40.8	62.4	44.4	24.1	45.0	53.2
	+MCA-E (Ours)	+0.07M	336.21	40.8	62.4	44.3	23.7	45.0	**52.9**
	+MCA-S (Ours)	+0.07M	336.21	**41.0**	**62.8**	**44.8**	**24.9**	**45.1**	**52.9**
RetinaNet	ResNet-50	37.74M	239.32	35.0	54.2	37.1	20.4	38.6	45.4
	+SE (CVPR,2018)	+2.49M	239.43	34.7	57.4	37.7	20.8	39.1	45.2
	+SRM (ICCV,2019)	+0.06M	239.32	35.8	54.8	**38.3**	20.3	39.6	46.2
	+ECA (CVPR,2020)	+0.09k	239.43	35.1	54.8	37.1	20.5	38.7	45.8
	+GCT (CVPR,2021)	+0.01k	239.43	35.8	55.6	37.9	20.5	39.6	46.6
	+MCA-E (Ours)	+0.03M	239.32	35.4	55.0	37.6	20.3	39.1	46.2
	+MCA-S (Ours)	+0.03M	239.32	**36.1**	**55.9**	38.0	**21.1**	**39.9**	**47.2**
	ResNet-101	56.74M	315.39	37.6	57.0	40.4	21.3	42.1	48.9
	+SE (CVPR,2018)	+4.71M	315.58	37.4	56.7	40.2	20.9	41.8	48.5
	+SRM (ICCV,2019)	+0.12M	315.39	36.8	55.9	39.2	20.7	41.1	47.8
	+ECA (CVPR,2020)	+0.17k	315.57	38.0	57.7	40.3	21.6	42.2	49.1
	+GCT (CVPR,2021	+0.03k	315.57	38.3	58.0	41.0	21.9	42.4	49.6
	+MCA-E (Ours)	+0.06M	315.39	38.0	57.9	40.4	**22.2**	42.2	49.5
	+MCA-S (Ours)	+0.06M	315.39	**38.8**	**58.9**	**41.4**	21.7	**43.0**	**51.3**

Table 3: Comparisons between different methods on the COCO val2017 set with object detection task.

Image Classification on ImageNet Dataset

Finally, we conduct image classification experiments on the ImageNet dataset. And we choose 4 models as the backbone, including: ResNet (He et al. 2016) and ShuffleNetV2 (Ma et al. 2018) model. The experimental results of image classification are shown in Table 4.

ResNet We first evaluate our MCA module on popular ResNet model. All attention modules are placed after the last BatchNorm layer inside each bottleneck of ResNet. MCA-E performs better than SENet, SRM, ECA, and GCT across different depths and backbones with similar parameters and slightly more computation. The results show that our MCA method can be used to improve the performance of residual network. It is worth noting that the SRM method achieves the comparable performance with MCA-E on Resnet-50 model but have a poor performance on Reset-18 model, while GCT method has comparable performance with MCA-E on Resnet-18 model. In general, MCA method has better generalization performance and brings improvement to different types of models.

ShufflenetV2 We also verify the effectiveness of our ECA module on lightweight CNN architectures. We utilize ShufflenetV2 (Ma et al. 2018) as backbone model, comparing our MCA module with baseline, SENet, SRM, ECA, and GCT module. All attention modules are placed before channel shuffle. The results in Table 4 show that MCA-E performance better than previous best SRM block by 0.61% in Top-1 accuracy. Compared to vanilla ShufflenetV2, our MCA-E improves by a 2.21% in Top-1 accuracy. These results indicate that MCA can be successfully applied to

Methods	Params	GFLOPs	Top-1	Top-5
ResNet-18	11.69M	1.822	69.76	89.08
+SE	+89.08K	1.823	70.59	89.78
+SRM	+0.77K	1.823	69.89	89.45
+ECA	+0.04K	1.823	70.85	89.75
+GCT	+0.01K	1.823	71.21	90.04
+MCA-E	+0.77K	1.823	**71.23**	**90.10**
+MCA-S	+0.78K	1.823	70.93	89.87
ResNet-50	25.56M	4.122	74.97	92.23
+SE	+2.53M	4.130	75.90	92.75
+SRM	+6.04K	4.122	76.44	93.02
+ECA	+0.09K	4.127	75.40	92.66
+GCT	+0.02K	4.127	74.02	91.75
+MCA-E	+6.06K	4.122	76.36	93.14
+MCA-S	+6.06K	4.133	**76.61**	**93.21**
ResNet-101	44.55M	7.849	76.51	93.10
+SE	+4.78M	7.863	76.72	93.31
+SRM	+130.0K	7.849	77.96	93.92
+ECA	+0.16K	7.858	76.56	92.96
+GCT	+0.03K	7.858	75.66	92.61
+MCA-E	+130.4K	7.849	78.18	**93.97**
+MCA-S	+130.4K	7.858	**78.21**	93.88
ShufflenetV2	2.28M	0.151	65.71	86.74
+SE	+16.8K	0.152	66.95	87.68
+SRM	+16.7K	0.151	67.32	87.41
+ECA	+0.08K	0.151	66.56	87.03
+GCT	+0.02K	0.151	62.24	84.20
+MCA-E	+16.7K	0.151	**67.93**	87.87
+MCA-S	+16.9K	0.151	67.91	**88.00**

Table 4: Image classification results of the state-of-the-art channel attention blocks on ImageNet dataset.

lightweight model.

Conclusion

In this paper, we analyze the excitation module of channel attention mechanism from a statistical perspective and utilize moment statistics to aggregate extensive moment feature, and propose the Moment Channel Attention (MCA) method that achieves higher performance and comparable complexity. Furthermore, we introduce the Cross Moment Convolution method to fuse low-order and other-order information both inside and cross the channel. Additionally, through a series of ablation experiments, we demonstrate the effectiveness of our MCA method. Finally, we evaluate the performance of our method in image classification, object detection, and instance segmentation tasks on large-scale datasets such as the COCO and ImageNet. The results of our comparative analysis show that our method outperforms other state-of-the-art channel attention methods. In addition, there are several aspects that need further improvement and investigation: (1) The M_{triple} method, as well as single moment M_2 and M_3 have not been thoroughly explored yet, leaving potential for further improvement in performance. (2) The fourth moment Kurtosis (Bai and Ng 2005) is widely used in the economics. Incorporate Kurtosis as another moment representation could be a valuable avenue for future investigation. In the future work, we will further investigate incorporation of MCA with spatial attention module.

Methods	AP	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L
ResNet50	34.0	55.2	36.0	16.3	36.9	49.6
+SE	34.2	56.0	36.0	15.9	37.2	49.8
+SRM	34.5	55.4	36.8	16.0	36.9	50.7
+ECA	34.5	56.4	36.4	16.1	37.7	50.3
+GCT	34.8	56.8	37.1	16.2	37.8	50.5
+MCA-E	35.3	**57.6**	37.3	**16.8**	38.4	**51.4**
+MCA-S	**35.4**	57.4	**37.7**	16.7	**38.7**	50.7
ResNet101	35.7	57.2	38.0	16.6	38.7	52.2
+SE	36.0	58.1	38.2	17.7	39.2	52.6
+SRM	35.8	57.2	38.0	16.1	39.0	52.4
+ECA	36.2	58.2	38.5	17.6	39.4	53.3
+GCT	36.7	59.0	**38.9**	17.4	40.1	**53.6**
+MCA-E	36.6	59.0	**38.9**	17.3	39.9	53.5
+MCA-S	**36.8**	**59.5**	38.8	**18.0**	**40.5**	53.3

Table 5: Instance Segmentation of the state-of-the-art channel attention blocks on COCO val2017 set.

Acknowledgments

This work is supported by the National Key R&D Program of China (2020YFB1313500), National Natural Science Foundation of China (T2293723, 61972347), the Key R&D Program of Zhejiang Province (2022C01022, 2022C01119, 2021C03003), and the Fundamental Research Funds for the Central Universities (No. 226-2022-00051).

References

Athreya, K. B.; and Lahiri, S. N. 2006. *Measure theory and probability theory*, volume 19. Springer.

Bai, J.; and Ng, S. 2005. Tests for skewness, kurtosis, and normality for time series data. *Journal of Business & Economic Statistics*, 23(1): 49–60.

Cao, Y.; Xu, J.; Lin, S.; Wei, F.; and Hu, H. 2019. GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 0–0.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Casella, G.; and Berger, R. 2002. *Statistical Inference*. Duxbury Press, 2nd edition.

Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; Zhang, Z.; Cheng, D.; Zhu, C.; Cheng, T.; Zhao, Q.; Li, B.; Lu, X.; Zhu, R.; Wu, Y.; Dai, J.; Wang, J.; Shi, J.; Ouyang, W.; Loy, C. C.; and Lin, D. 2019. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv preprint arXiv:1906.07155*.

Chen, Y.; Kalantidis, Y.; Li, J.; Yan, S.; and Feng, J. 2018. Aˆ 2-nets: Double attention networks. *Advances in neural information processing systems*, 31.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

Egozcue, M.; García, L. F.; Wong, W.-K.; and Zitikis, R. 2012. THE SMALLEST UPPER BOUND FOR THE pTH ABSOLUTE CENTRAL MOMENT OF A CLASS OF RANDOM VARIABLES. *Mathematical Scientist*, 37(2).

Flusser, J. 2000. On the independence of rotation moment invariants. *Pattern recognition*, 33(9): 1405–1410.

Flusser, J.; and Suk, T. 2006. Rotation moment invariants for recognition of symmetric objects. *IEEE Transactions on Image Processing*, 15(12): 3784–3790.

Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; and Lu, H. 2019. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3146–3154.

Gao, Z.; Xie, J.; Wang, Q.; and Li, P. 2019. Global second-order pooling convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3024–3033.

Garthwaite, P. H.; Jolliffe, I. T.; and Jones, B. 2002. *Statistical inference*. Oxford Science Publications.

Gonzalez, R. C.; and Woods, R. E. 2008a. *Digital image processing*. Prentice Hall.

Gonzalez, R. C.; and Woods, R. E. 2008b. *Digital image processing*. Prentice Hall.

Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D.; and Wierstra, D. 2015. Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, 1462–1471. PMLR.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Huawei Technologies Co., L. 2022. Deep Learning Frameworks. In *Artificial Intelligence Technology*, 123–135. Springer.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. PMLR.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

Lee, H.; Kim, H.-E.; and Nam, H. 2019. Srm: A style-based recalibration module for convolutional neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1854–1862.

Li, X.; Wang, W.; Hu, X.; and Yang, J. 2019. Selective kernel networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 510–519.

Li, Z.; Liu, S.; and Sun, J. 2021. Momentumˆ 2 Teacher: Momentum Teacher with Momentum Statistics for Self-Supervised Learning. *arXiv preprint arXiv:2101.07525*.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Liu, C.; Wang, D.; Zhang, H.; Wu, W.; Sun, W.; Zhao, T.; and Zheng, N. 2022. Using simulated training data of voxel-level generative models to improve 3D neuron reconstruction. *IEEE transactions on medical imaging*, 41(12): 3624–3635.

Ma, N.; Zhang, X.; Zheng, H.-T.; and Sun, J. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, 116–131.

Park, J.; Woo, S.; Lee, J.-Y.; and Kweon, I. S. 2018. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*.

Patel, J. K.; and Read, C. B. 1996. *Handbook of the normal distribution*, volume 150. CRC Press.

Qin, Z.; Zhang, P.; Wu, F.; and Li, X. 2021. Fcanet: Frequency channel attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 783–792.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Roy, A. G.; Navab, N.; and Wachinger, C. 2018. Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks. *IEEE transactions on medical imaging*, 38(2): 540–549.

Ruan, D.; Wang, D.; Zheng, Y.; Zheng, N.; and Zheng, M. 2021. Gaussian context transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15129–15138.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3): 211–252.

Smeulders, A. W.; Worring, M.; Santini, S.; Gupta, A.; and Jain, R. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12): 1349–1380.

Su, W.; Zhu, X.; Cao, Y.; Li, B.; Lu, L.; Wei, F.; and Dai, J. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.

Tong, Q.; Liang, G.; and Bi, J. 2022. Calibrating the adaptive learning rate to improve convergence of ADAM. *Neurocomputing*, 481: 333–356.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; and Hu, Q. 2020a. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11531–11539.

Wang, Q.; Wu, T.; Zheng, H.; and Guo, G. 2020b. Hierarchical Pyramid Diverse Attention Networks for Face Recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8326–8335.

Wang, X.; Girshick, R.; Gupta, A.; and He, K. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7794–7803.

Woo, S.; Park, J.; Lee, J.-Y.; and Kweon, I. S. 2018. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, 3–19.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500.

Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; and He, X. 2018. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1316–1324.

Yang, J.; Ren, P.; Zhang, D.; Chen, D.; Wen, F.; Li, H.; and Hua, G. 2017. Neural aggregation network for video face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4362–4371.

Yang, Z.; Zhu, L.; Wu, Y.; and Yang, Y. 2020. Gated channel transformation for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11794–11803.

Yu, H.; Li, M.; Zhang, H.-J.; and Feng, J. 2002. Color texture moments for content-based image retrieval. In *Proceedings. International Conference on Image Processing*, volume 3, 929–932. IEEE.

Yuan, Y.; Huang, L.; Guo, J.; Zhang, C.; Chen, X.; and Wang, J. 2018. Ocnet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*.

Zellinger, W.; Grubinger, T.; Lughofer, E.; Natschläger, T.; and Saminger-Platz, S. 2017. Central moment discrepancy (cmd) for domain-invariant representation learning. *ICLR*.

Zellinger, W.; Moser, B. A.; Grubinger, T.; Lughofer, E.; Natschläger, T.; and Saminger-Platz, S. 2019. Robust unsupervised domain adaptation for neural networks via moment alignment. *Information Sciences*, 483: 174–191.

Zhang, H.; Goodfellow, I.; Metaxas, D.; and Odena, A. 2019. Self-attention generative adversarial networks. In *International conference on machine learning*, 7354–7363. PMLR.

Zou, F.; Shen, L.; Jie, Z.; Zhang, W.; and Liu, W. 2019. A Sufficient Condition for Convergences of Adam and RMSProp. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11119–11127.