Accumulating global channel-wise patterns via deformed-bottleneck recalibration

Thanh Tuan Nguyen[1*], Thanh Phuong Nguyen[2] and Vincent Nguyen[3]

[1]HCMC University of Technology and Education, Faculty of IT, Thu Duc City, Ho Chi Minh City, Vietnam.
[2]Université de Toulon, Aix Marseille Univ, CNRS, LIS, Marseille, France.
[3]Université d'Orléans, LIFO, Orléans, France.

*Corresponding author(s). E-mail(s): tuannt@hcmute.edu.vn;
Contributing authors: tpnguyen@univ-tln.fr;
vincent.nguyen@univ-orleans.fr;

Abstract

Embedding attention modules into deep convolutional neural networks (CNNs) is currently one of the common deliberations to enhance their learning ability of feature representation. In previous works, the global channel-wise patterns of a given tensor are computed and squeezed into CNN-based models through an attention mechanism. Squeezing different kinds of these features can lead to the less fusion of attentive information due to the independent operations of channel-wise recalibration. To deal with this issue, an efficient attention module of accumulated features (**MAF**) is proposed by accumulating these diverse squeezes for a unitary recalibrating perceptron as follows. Firstly, we take advantage of average and deviation calculations to produce correspondingly statistical patterns of a given tensor for aggregating the global channel information. An adaptative perceptron of deformed-bottleneck recalibration (**DBR**) is then presented to cohere the resultant features. Finally, the robust **DBR**-based lightweights will be utilized to weight the concerning tensor. Additionally, to exploit more spatial-wise information, we address **MAF** for an effective alternative of the channel-wise component in two critical attention units to form two corresponding modules that will be then inspected to indicate which integration is good for real applications. We adapt the MAF-based modules to MobileNets for further enhancement investigation. Experiments on benchmark datasets for image classification have proved the efficacy of our proposals. The code of the MAF module is available at https://github.com/nttbdrk25/MAFAttention.

Keywords: Light-weight models, Feature attention, CNNs, Neural networks, Image classification

1 Introduction

Deep CNN-based architecture is one of the various powerful learning techniques to deal with diverse assignments in computer vision such as image classification [1–4], object detection [5–9], semantic segmentation [10–13], semantic segmentation [11, 13–15], real-life issues [16–20], etc. Indeed, inspired by the impressive performance of AlexNet [1], researchers have introduced various productive network architectures in deeper and wider learning layers which are stacked by multi-convolutional blocks for informative representation. For popular instances, VGGNet [21] consists of 16 or 19 layers, twice more layers than AlexNet. The large architecture of ResNet [22] can go with up to 1202 layers that are adapted with residual connections to improve the gradient flow. GoogLeNet [23] is also structured by very deep layers for forceful representation with diverse feature concatenations that are obtained by addressing various sizes of filtering kernels for a convolutional block. DenseNet [24] exploits the concatenations of feature maps computed from different layers. Recently, several miniature networks with small computing layers have been presented to be adaptative for mobile devices, whose computational resources have been restricted, e.g., MobileNet-based models [25–29], factorized CNNs [30], MnasNet [31], EfficientNet [32], Xception [33], and ShuffleNet [34]. etc. For light hyper-parameters, most of them are usually designed by allocating depthwise and pointwise convolutions.

In order to direct the CNN-based networks to concentrate on the important information instead of learning on the useless backgrounds, many feature-attention approaches have been proposed by hooking slight attention models on their architecture to enhance the learning ability. These attention propositions have to ensure that the model complexity of the obtained networks is still at a reasonable level in comparison with the baseline ones. Indeed, Hu *et al.* [35] introduced a Squeeze-and-Excitation (SE) block with a lightweight gating mechanism to exploit the global average-pooled features of a given tensor X for enhancing the representational power of deep CNN-based networks. Motivated by this attention mechanism, several methods have been introduced to take advantage of the spatial/channel-wise patterns of X such as Convolutional Block Attention Module (CBAM) [36], Bottleneck Attention Module (BAM) [37], Style-based Recalibration Module (SRM) [38], Efficient Channel Attention (ECA) [39], etc. Some of them [37, 39] just address one squeezing kind of global channel-wise features, leading to lack of attentive information for image representation. In the meanwhile, some others [36, 38] take into account two kinds of squeezes. However, their squeezing process for weightable values can lead to less attentive information because their interval recalibration is handled independently for an aggregating operation of global channel-wise patterns.

Addressing the above problems, an efficient Module of Accumulated Features (MAF) is proposed in this work to accumulate two diverse squeezes for a unitary recalibrating perceptron. For a given tensor X, two squeezes of its global average and standard-deviation features are taken into account for accumulating the global channel-wise information. An adaptative perceptron of deformed-bottleneck recalibration (DBR) is then introduced to cohere the resultant squeezes in order for producing robust lightweights embedded into the concerning tensor X. Finally, the MAF-based features will be computed and delivered to MobileNets for informative enhancement in image representation. The proposed MAF module will make the model complexity of the MAF-based networks trivially increase in comparison with that of the baseline ones, while the performance is better. It should be noted that our accumulative mechanism is different from that of CBAM [36] and SRM [38]. CBAM [36] just addresses a simple element-wise summation of two global descriptors computed by two discrete SE-based operations, while SRM [38] utilizes a channel-independent fully connected layer for each channel-wise squeeze. Thanks to the proposed DBR perceptron, MAF can make an attentive fusion of the global features at the learning stage of the adaptive recalibration. This allows the MAF-based networks to boost their discriminative representation when learning the nonlinear interaction of their channels. In addition, in order to take advantage of the extra spatial-wise information, we locate MAF as an efficient alternative of the channel-wise component in two particular attention modules (i.e., CBAM [36] and BAM [37]) to form two corresponding modules (named MAF_C and MAF_B respectively). We then investigate them to find out which integration is better for real applications. Experiments for image classification on benchmark datasets have proved that the MAF-based networks obtain better results compared to the baseline ones as well as other attention-based methods. In short, the main contributions of this work can be summarized as

- An efficient accumulation to simultaneously unify two squeezing kinds of global channel-wise patterns.
- A unitary recalibrating perceptron to effectively exploited diverse squeezes for producing robust lightweights.
- Investigating alternatives of MAF for the channel-wise component of two particular attention modules.
- MobileNets adapted with MAF-based features obtain good performances in comparison with the baseline ones as well as other attention methods.

2 Related works

2.1 Squeeze-and-excitation attention mechanism

To enhance the representation power of deep CNN-based networks, one of the main approaches is to hook some sub-models on several layers of their architecture so that they can concentrate on the important properties instead of the fewer ones, while the architecture complexity of the hooked networks reasonably increases in comparison with the baseline ones. To this end, Hu *et al.* [35] introduced a squeeze-and-excitation (SE) attention block to concentrate on the selective channel-wise information of a given tensor while passing over the less useful features. Indeed, let $X \in \mathbb{R}^{H \times W \times C}$ be an input tensor. The SE block is built as a computational block to transform $z_i \in X$ to $\widetilde{z}_i \in \widetilde{X}$ by addressing the global channel-wise information extracted from X. In general, the transformation can be defined subject to a volume of attention weights $\mathbf{W} \in \mathbb{R}^{H \times W \times C}$ as

$$\Psi : X \to \widetilde{X}, \widetilde{z}_i = \Psi(z_i) = \omega_i \odot z_i \tag{1}$$

where $\widetilde{X} \in \mathbb{R}^{H \times W \times C}$; $\odot$ means a binary operator to embed a lightweight value $\omega_i \in \mathbf{W}$ into a tensor element $z_i \in X$.

2.2 SE-based attention modules

Thanks to a lightweight gating mechanism, the SE block [35] has been taken into account global average-pooled features for enhancing the representational power of GoogLeNet [23] and ResNet [22]. Motivated by this attention mechanism, Lee *et al.* [38] introduced SRM to utilize squeezes of average and deviation features with a channel-independent fully connected layer for each squeeze. CBAM [36] exploited both global average-pooled and max-pooled informative features for the separate SE-based operations. Then the channel-refined tensor was spatially averaged and maximized before feeding the computed spatial information into a slight convolutional block for extracting spatial attention features. BAM [37] was proposed to learn the integration of the global average-pooled features and the spatial information that were computed by a SE-based operation and a dilated convolutional function [40], respectively. Wang *et al.* [41] took into account the non-local algorithm (NL) [42] to build an attention module of NL blocks. Motivated by this non-local concept, Chen *et al.* [43] introduced double attention networks (A^2-Nets) by aggregating and propagating the spatio-temporal informative features of input images/videos. Fu *et al.* [44] addressed simultaneously NL-based channel and spatial information for attention mechanism. Cao *et al.* [45] combined the optimal implementation of a simplified NL block and a SE block to form a global context (GC) block for an improvement, while a non-local spatial attention module (NL-SAM) [46] was proposed by a combination of NL-block concept and CBAM [36]. In other attention approaches, Hu *et al.* [47] proposed a Gather-Excite (GE) module to aggregate the contextual information of feature maps. Two spatial pooling stages of rich descriptor extraction and information fusion [48] were formed to take into account more globally informative cues absorbed by a fusion step for a powerful channel characteristic. Wang *et al.* [39] introduced an ECA module for local cross-channel interaction by addressing fast 1D convolution with 1D kernels of neighbors to hold channel dimensionality when learning the global average-pooled information of an input tensor. A Selective Kernel (SK) block [49] brought the feature-map attention across two convolutional branches, while Zhang *et al.* [50] proposed a general model of SE and SK blocks, called Split Attention (SA) that was accelerated by grouped convolutions for cardinal representations. Other noticeable attention modules are inferred as FcaNet [51]: multi-spectral channel attention based on various frequency components; SimAM [52]: a SIMple parameter-free Attention Module for full 3-D weights; Attention Augmented convolution (AA) [53]: an attentive mechanism for jointly serving to spatial and feature subspaces; Pyramid Squeeze Attention (PSA) [54]: splitting the channels of a given tensor into branches for grouped convolutions.

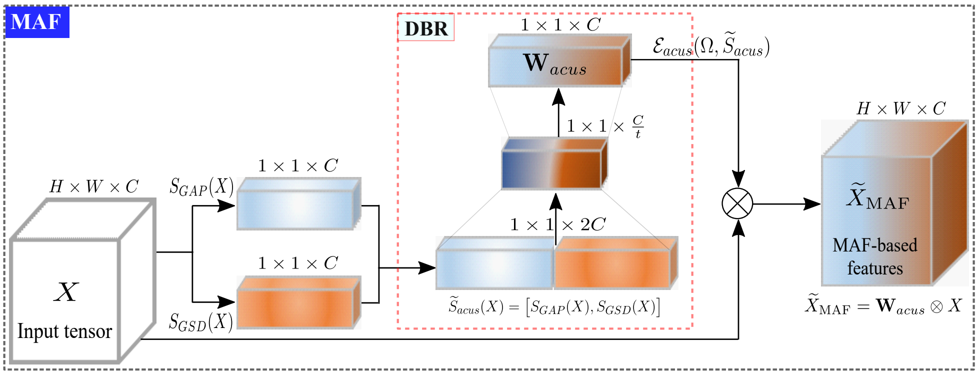

MAF
DBR
$1 \times 1 \times C$
$\mathbf{W}_{acus}$
$\mathcal{E}_{acus}(\Omega, \widetilde{S}_{acus})$
$1 \times 1 \times \frac{C}{t}$
$H \times W \times C$
$S_{GAP}(X)$
$1 \times 1 \times C$
$1 \times 1 \times 2C$
$\widetilde{X}_{\mathrm{MAF}}$
MAF-based features
X
Input tensor
$1 \times 1 \times C$
$S_{GSD}(X)$
$\widetilde{S}_{acus}(X) = [S_{GAP}(X), S_{GSD}(X)]$
$\widetilde{X}_{\mathrm{MAF}} = \mathbf{W}_{acus} \otimes X$

Fig. 1 An intuitive illustration of the proposed module of accumulated channel-wise features (MAF).

3 A productive accumulation of global features

The channel-wise lightweights of SE-based attention modules [36, 38] can lack adhesive information representations due to the independent recalibration of squeezes. To mitigate this deficiency, we propose a module of accumulated features (MAF) with an efficient accumulating operation to fuse two kinds of channel-wise statistical information. Fig. 1 graphically illustrates the attentive mechanism of MAF. It includes three main stages: *i)* Squeezing the global average (S_{GAP}) and standard deviation (S_{GSD}) features of a given tensor; *ii)* A perceptron of deformed-bottleneck recalibration (DBR) for the squeezed output ($\widetilde{S}_{acus}$) to produce a robust lightweight volume; and *iii)* Feature attention with DBR-based lightweights. Hereafter, these stages will be presented in detail.

3.1 Squeezing global channel-wise patterns

In the previous attention modules [36, 38], light-weights are extracted from learning two squeezes of global channel features in two normal ways: *i)* a separate recalibration for each squeeze and *ii)* a simple element-wise summation of two squeezes. These operations can lead to the less fusion of attentive information. Addressing this issue, we propose an efficient accumulation of two global channel squeezes for a unitary perceptron of recalibration. In this section, we present the corporation of average and deviation squeezes, while the unitary recalibration will be mentioned in Section 3.2.

Average squeeze: As mentioned in the SE block [35], the statistic feature of global average pooling (named $S_{GAP}(X) \in \mathbb{R}^{1\times1\times C}$) is computed by considering the corresponding spatial planes of tensor X as

$$S_{GAP}(\xi_c \in X) = \frac{1}{\mathcal{N}} \sum_{x \in \xi_c} f(x) \tag{2}$$

where ξ_c denotes the c^{th} channel of X; $f(.)$ returns a specific feature map value of a 2D point $x \in \xi_c$; $\mathcal{N} = H \times W$ means the number of 2D points of ξ_c.

Deviation squeeze: Motivated by [38, 55], subject to $S_{GAP}(X)$, we calculate another type of channel-wise statistic pattern in consideration of the global standard deviation (named $S_{GSD}(X) \in \mathbb{R}^{1\times1\times C}$) of each spatial plane ξ_c of tensor X as

$$S_{GSD}(\xi_c \in X) = \sqrt{\frac{1}{\mathcal{N}} \sum_{x\in\xi_c} \Big(f(x) - S_{GAP}(\xi_c)\Big)^2} \tag{3}$$

In order to capture diversity of contextual information, these two squeezes will be concatenated to obtain an accumulated squeeze (ACUS) of the given tensor X as

$$\widetilde{S}_{acus}(X) = \big[S_{GAP}(X), S_{GSD}(X)\big] \tag{4}$$

It can be seen that $\widetilde{S}_{acus} \in \mathbb{R}^{1\times1\times2C}$ has a double size in comparison with the traditional squeezes of global channel features (i.e., S_{GAP} or S_{GSD}) addressed for the previous SE-based attention modules [35–39]. So it is necessary to take a favorable recalibration for the informative fusion of $\widetilde{S}_{acus}$. A such adaptative perceptron will be proposed and presented hereafter.

3.2 A perceptron of deformed-bottleneck recalibration

As well known, the previous SE-based modules [35–37] utilized a bottleneck excitation for recalibrating a $1\times1\times C$ block of squeezed information (e.g., S_{GAP}, S_{GSD}, etc.), where a reduction ratio is allocated to control the sharp increase of #parameters. Definitely, this initial bottleneck shape is not suitable for recalibrating our accumulated squeeze $\widetilde{S}_{acus}$ due to the double dimension i.e., $1\times1\times2C$. Like the conventional squeezes, $\widetilde{S}_{acus}$ is purely a collection of discrete channel-wise patterns. To make them fused-dependent, it needs a learning process to perceive the nonlinear interactions between channels. To this end, we propose a deformed bottleneck recalibration (named DBR) to capture the fused dependencies of diverse channel features gathered in $\widetilde{S}_{acus}$. The DBR perceptron consists of two fully connected layers with a non-linear function between them. The first layer is for a dimensional reduction with learnable parameters $\Omega_1 \in \mathbb{R}^{\frac{C}{r}\times2C}$, where r denotes a reduction ratio. In the meanwhile, the second layer with $\Omega_2 \in \mathbb{R}^{C\times\frac{C}{r}}$ is for an expansion so that the output would be formed in $1\times1\times C$ dimension to be agreed with a scale calculation with the input tensor X (see Fig. 1). In short, the DBR perceptron can be rewritten as

$$\mathbf{W}_{acus} = \mathcal{E}_{acus}(\Omega, \widetilde{S}_{acus}) = \eta\big(\Omega_2\gamma(\Omega_1\widetilde{S}_{acus} + b_1) + b_2\big) \tag{5}$$

where $\eta(.)$ is a sigmoid function; $b_1 \in \mathbb{R}^{\frac{C}{r}}$ and $b_2 \in \mathbb{R}^{C}$ are biases. The non-linear function $\gamma(.)$ denotes activation ReLU [56]. As defined in Eq. (5), it can be deduced that the total number of the learnable parameters for DBR is $\frac{3C^2}{r}$ in the case of ignoring the biases. It is a negligible quantity compared to other channel-wise components, e.g., $\frac{2C^2}{r}$ of SE [35] and BAM [37]; or $\frac{4C^2}{r}$ of CBAM [36] (refer to Table 3 for specific estimations). The aforementioned ratio r plays the same role of dimensional reduction in the SE-based modules. It can be referred to [35] for further information about various investigations of the influence of r. Accordingly, the reduction ratio should be set to $r = 16$ for a good trade-off between the accuracy and the increase of the model complexity. For objective evaluations, we utilize this ratio value for the proposed DBR perceptron and other SE-based ones [36, 37] in the below experiments.

3.3 Embedded features with DBR-based lightweights

As mentioned above, the proposed DBR perceptron can take advantage of the accumulated squeeze ($\widetilde{S}_{acus}$) for robust lightweights ($\mathbf{W}_{acus}$). According to Eq. (1), MAF-based feature maps $\widetilde{X}_{\text{MAF}}$ can be pointed out by embedding $\mathbf{W}_{acus}$ into the given tensor X, where each channel $\xi_c \in X$ will be correspondingly weighted with each $\omega_c \in \mathbf{W}_{acus}$ as

$$\widetilde{X}_{\text{MAF}} = \mathbf{W}_{acus} \otimes X : \text{each channel } \widetilde{\xi}_c \in \widetilde{X}_{\text{MAF}}, \widetilde{\xi}_c = \omega_c \otimes \xi_c \tag{6}$$

Therein, "$\otimes$" denotes an element-wise multiplication operator (see Fig. 1 for an intuitively illustration of this calculation).

4 Beneficial properties of MAF module

4.1 Extracting robust lightweights for attention mechanism

It can be deduced that the proposed MAF module can produce robust lightweights for attention mechanism in comparison to other modules. It is thanks to three main properties as

- Two kinds of global channel features (S_{GAP} and S_{GSD}) are extracted in MAF instead of only one done in [35, 37, 39].
- All these global features will be joined to form an accumulated squeeze $\widetilde{S}_{acus}$, instead of being considered as separate squeezes in CBAM [36] and SRM [38].
- An efficient DBR perceptron is proposed to recalibrate the accumulated squeeze $\widetilde{S}_{acus}$ for robust lightweights.

Furthermore, it should be noted that the proposed DBR-based recalibration is different from SRM [38]. Indeed, both S_{GAP} and S_{GSD} are also addressed for attention-weight calculation in SRM [38]. However, the separate recalibration of SRM can lead to the less fused coherence of global channel features because a channel-independent fully connected layer was applied for each channel-wise squeeze. Empirical evaluations in Section 6.2.1 have validated the effectiveness of the unified information $\mathbf{W}_{acus}$ in comparison with SRM and other channel-wise-based modules. Additionally, our accumulative mechanism will make a solid fusion of channel-wise statistics, while the CBAM block [36] is designed with two discrete SE-based operations and then a pure element-wise summation is applied for two corresponding outputs (see Section 6.2.2 for evaluations in detail).

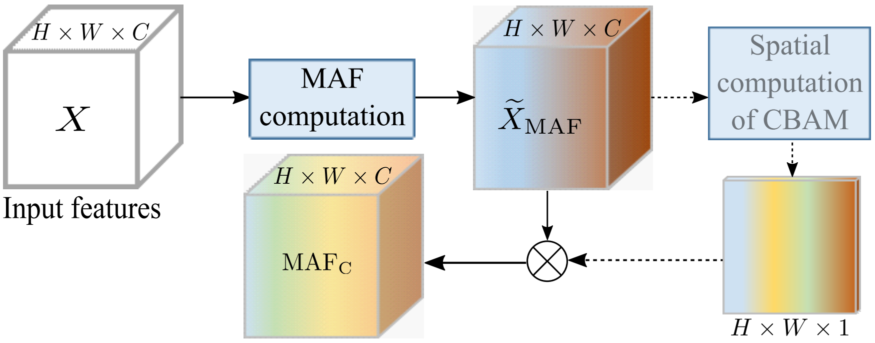


Fig. 2 A $\mathrm{MAF_C}$ attention, where the dash arrows indicate the spatial computation of CBAM [36].

4.2 An efficient alternative for channel attention

As presented in Section 3, it can be seen that the proposed MAF block is intended for exploiting the accumulated channel-wise information of a given tensor. Meantime, several recent approaches [36, 37, 57] indicated that the spatial-wise characteristics of X also have an important contribution in boosting the performance. So we would like to inherit their spatial computation to enrich more attentive features. To this end, we propose an efficient replacement for the channel-wise component of two particular CNN-based attentions, i.e., CBAM [36] and BAM [37]. Concretely, we will replace their channel-wise computation with our MAF to form efficient attention modules, whose weightable features for X are simply conducted by making an element-wise product between the MAF-based values and the spatial ones of CBAM and BAM. As a result, we obtain two spatial-channel feature blocks, named $\mathrm{MAF_C}$ and $\mathrm{MAF_B}$ correspondingly. Figs. 2 and 3 illustrate the adaptative substitution of MAF for channel attention, where the blurred shapes denote the inherited spatial computation. It should be noted that when the S_{GSD} operation of MAF is addressed for the channel-wise computation of $\mathrm{MAF_C}$, the corresponding S_{GSD}-based spatial features would be taken into account, instead of the maximum-spatial ones of CBAM [36]. Experiments have verified that $\mathrm{MAF_C}$ and $\mathrm{MAF_B}$ can obtain better performance in comparison with the original modules [36, 37], while controlling the computational complexity of the corresponding MAF-based networks to be reasonable (see Section 6.2 for more evaluations).

5 Adapting MAF-based attentions to MobileNets

As well known, depthwise and pointwise convolutions are two main operators that are used to design slight layers with a trivial number of trainable parameters for small neural networks [27, 33, 34]. In MobileNets, these light-weight operators are gathered into CNN-based groups: blocks of depthwise separable convolutions (DSC) in MobileNetV1 [25] or linear bottleneck blocks (LBN) in MobileNetV2 [26] and MobileNetV3 [27]. These tiny networks are potential solutions for applications of mobile devices. So we would like to integrate our MAF-based features into their architecture to investigate the efficiency of the proposed MAF-based modules. Accordingly, the output

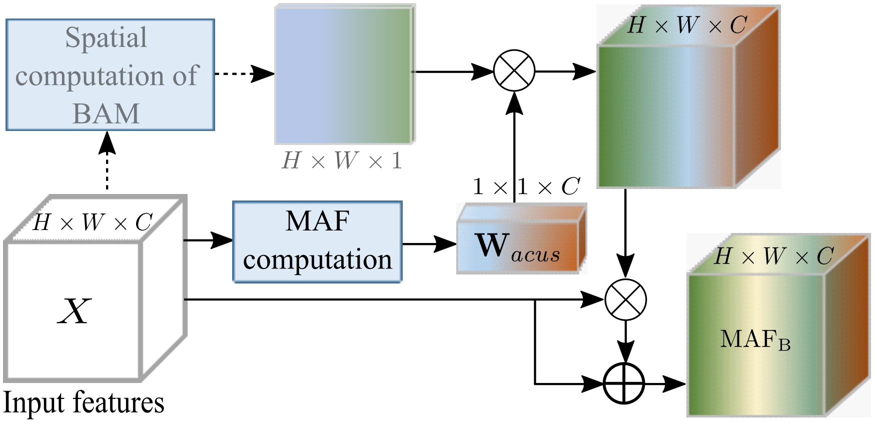


Fig. 3 A $\mathrm{MAF_B}$ attention module, where the dash arrows indicate the spatial computation of BAM [37]. $\bigoplus$ denotes an element-wise summation.

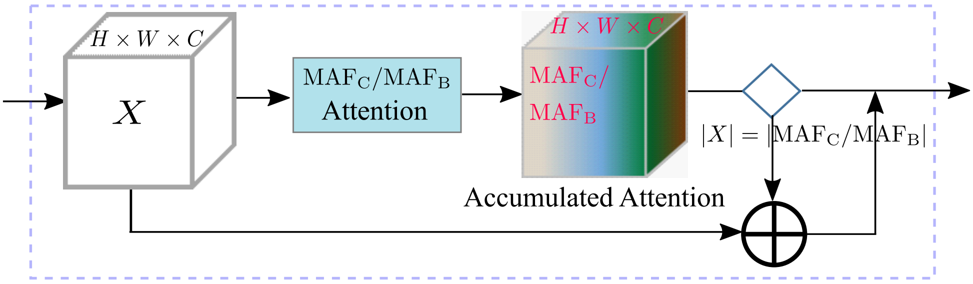

$H \times W \times C$
X
$\mathrm{MAF_C}/\mathrm{MAF_B}$ Attention
$\mathrm{MAF_C}/$ $\mathrm{MAF_B}$
$|X| = |\mathrm{MAF_C}/\mathrm{MAF_B}|$
Accumulated Attention

Fig. 4 Illustration of weighting the accumulated MAF-based features for a given tensor X, which is an output of a DSC-based block of MobileNetV1 [25] or a LBN-based block of MobileNetV2 [26] and MobileNetV3 [27]. $\bigoplus$ denotes an element-wise summation.

of the DSC/LBN blocks will be weighted by either the MAF information or the $\mathrm{MAF_C}/\mathrm{MAF_B}$ features to find out which attentive integration is better and potential for real applications. Fig. 4 intuitively illustrates the weighting process of the accumulated MAF-based features for a given DSC/LBN-based tensor X in MobileNets. It should be noted that because Howard *et al.* [27] embedded the SE block [35] in the initial architecture of MobileNetV3, we will remove it and take the location for the proposed MAF-based ones. The implementation code of the MAF-based networks is available at https://github.com/nttbdrk25/MAFAttention.

6 Experiments and evaluations

6.1 Parameter settings and datasets

To thoroughly investigate the efficiency of MAF-based attention mechanisms, we address them for different architectures of MobileNets, i.e., *width-multiplier* $= \{\mathrm{x}0.25, \mathrm{x}0.5, \mathrm{x}0.75, \mathrm{x}1.0\}$. The aforementioned MAF-based networks will be trained and evaluated for image classification tasks on the following datasets with general learning arguments: a SGD optimizer with *momentum* $= 0.9$; initial learning-rate $lr = 0.1$; weight-decay $= 10^{-4}$. Other learning settings will be allocated subject to the particular properties of each dataset for objective comparisons.

CIFAR-10/100 [58] consists of two subsets of 60k 32×32 color images: CIFAR-10 with 10 categories; and CIFAR-100 with 100 categories. Each subset includes 50k images for the training stage and 10k images for the testing one. Following the settings in [22, 29, 59] for both subsets, we train the proposed MAF-based networks for 200 epochs with 128 training instances in each batch. The initial learning rate ($lr = 0.1$) will be decayed by a factor of 0.1 at epochs: 100, 150, and 180. To avoid the overfitting problem, the augmentations in [22, 59] will be addressed so that the training instances are randomly augmented by horizontal flips and shifts by up to 4px.

ImageNet-1k [60] is a challenging large database. It includes 1000 categories of scene images. For the image classification task, 1.28 million images are taken for the training stage and 50k for the validating one. Following [29, 36, 37], we train the proposed MAF-based networks on a set of cropped 224×224 images for 100 epochs with 256 training instances in each batch. The initial learning rate ($lr = 0.1$) will be decayed by a factor of 0.1 at epochs: 30, 60, and 90. The augmenting transforms will be applied as follows: scaling images into dimension of 256×256 and randomly cropping the output images for 224×224; horizontal flips.

ImageNet-100 was introduced in 2020 by Tian *et al.* [61]. It is constructed by randomly taking out 100 classes of ImageNet [60] for evaluating the image classification ability of contrastive learning models. In this work, we would like to take into account for validating light-weight models. We address the same learning settings as done on ImageNet-1k to train the MAF-based networks.

Stanford Dogs [62] is composed by gathering dog images and their corresponding annotations from those of ImageNet [60]. It has over 20k images that are categorized into 120 breeds of dogs with 12000 images for the training stage and 8580 for the testing one. Subject to the experimental scenario of Haase and Amthor [29], we train the MAF-based networks for 200 epochs with 64 training images in each batch. The learning rate ($lr = 0.1$) will be decayed by a factor of 0.1 at epochs: 100, 150, and 180. The augmenting transforms will be applied as the same on ImageNet-1k with color jitter [1].

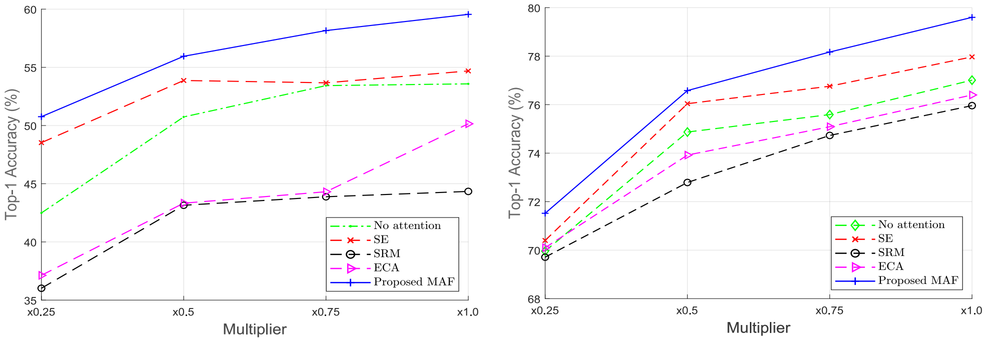
6.2 Efficiency analysis of MAF-based attentions

To objectively compare the efficiency of the proposed MAF-based attentions with others, we would like to deploy several recent spatial-channel attentions on MobileNets with the same parameter settings, as mentioned in Section 6.1. Without feature attention, we implement and report the results of the original MobileNets. For modules with only channel-wise patterns, we address SE [35], SRM [38], and ECA [39]. For both spatial-wise and channel-wise, CBAM [36] and BAM [37] will be considered. It should be noted that there is an SE module [35] with the reduction ratio $r = 4$, which is embedded in the initial model of MobileNetV3. So we will simply replace it with the compared modules with $r = 16$ for objective evaluations. For convenient presentation, we would like to refer to the MobileNets' versions by short names: V1 for MobileNetV1

Table 1 Top-1 accuracy (%) with different attention modules on CIFAR-10/100.

			MobileNetV1				MobileNetV2				MobileNetV3			
Dataset	Attention		(x0.25)	(x0.50)	(x0.75)	(x1.0)	(x0.25)	(x0.50)	(x0.75)	(x1.0)	(x0.25)	(x0.50)	(x0.75)	(x1.0)
		No Attention	90.96	92.46	93.21	93.76	89.60	92.64	93.40	94.06	90.70	92.81	94.10	94.08
CIFAR-10	Channel	SE [35]	91.29	92.93	93.61	94.22	**90.06**	92.48	**93.75**	94.06	90.63	92.94	93.70	94.08
		SRM [38]	91.49	93.66	94.14	94.26	89.35	92.54	93.60	**94.46**	90.53	92.43	93.57	94.13
		ECA [39]	**91.82**	93.25	93.40	94.04	89.05	92.59	93.65	94.11	89.87	93.01	93.74	94.21
		MAF	91.40	**93.70**	**93.96**	**94.53**	89.83	**92.78**	93.61	93.98	**90.79**	**93.33**	**93.87**	**94.30**
	Channel+Spatial	CBAM [36]	91.22	92.84	93.73	93.98	90.09	93.00	93.62	93.63	90.56	92.55	93.75	93.97
		$\mathbf{MAF_C^{avg\_max}}$	91.55	**93.56**	93.98	94.29	**90.35**	92.76	93.83	93.76	**90.75**	**92.69**	93.66	94.13
		$\mathbf{MAF_C^{avg\_std}}$	**91.65**	93.50	**94.24**	**94.35**	**90.35**	**93.02**	**93.88**	**93.86**	89.26	92.61	**93.80**	**94.21**
	Channel	BAM [37]	91.18	**93.08**	93.34	**93.68**	**90.64**	92.75	93.56	93.94	90.68	93.29	93.89	94.22
		$\mathbf{MAF_B}$	**91.23**	92.73	**93.57**	93.63	90.09	**92.83**	**93.69**	**94.13**	**90.77**	**93.31**	**93.95**	**94.38**
		No Attention	68.08	71.85	73.97	74.27	69.12	72.50	73.30	74.38	67.50	71.52	72.20	73.40
CIFAR-100	Channel	SE [35]	69.69	73.28	74.63	75.37	69.31	72.37	73.52	74.47	67.01	71.63	73.71	74.36
		SRM [38]	68.81	72.86	73.83	75.08	65.79	71.98	74.07	74.66	67.24	72.05	73.03	73.61
		ECA [39]	68.39	71.72	73.33	73.96	66.97	70.93	72.82	73.41	67.49	71.74	72.51	73.05
		MAF	**70.29**	**73.56**	**75.87**	**75.95**	**70.29**	**72.66**	**74.74**	**74.82**	**67.88**	**72.28**	**74.07**	**74.65**
	Channel+Spatial	CBAM [36]	67.86	72.50	74.11	74.98	**68.95**	71.54	73.51	73.39	67.99	**72.51**	73.89	73.70
		$\mathbf{MAF_C^{avg\_max}}$	69.62	72.76	75.06	75.72	68.12	72.36	74.42	75.31	67.89	72.50	73.89	74.64
		$\mathbf{MAF_C^{avg\_std}}$	**70.15**	**74.07**	**75.45**	**76.17**	67.32	**72.63**	**74.46**	**75.95**	**68.07**	72.28	**74.09**	**75.14**
	Channel	BAM [37]	68.96	72.74	74.86	75.07	68.00	73.10	74.49	75.59	67.69	72.84	74.05	75.06
		$\mathbf{MAF_B}$	**69.13**	**73.65**	**75.81**	**75.94**	**69.15**	**73.12**	**74.69**	**75.67**	**67.86**	**73.12**	**74.66**	**75.44**

Note: Numbers in parentheses indicate different width multipliers for model reduction of MobileNets.



(a) Performances on Stanford Dogs

(b) Performances on ImageNet-100

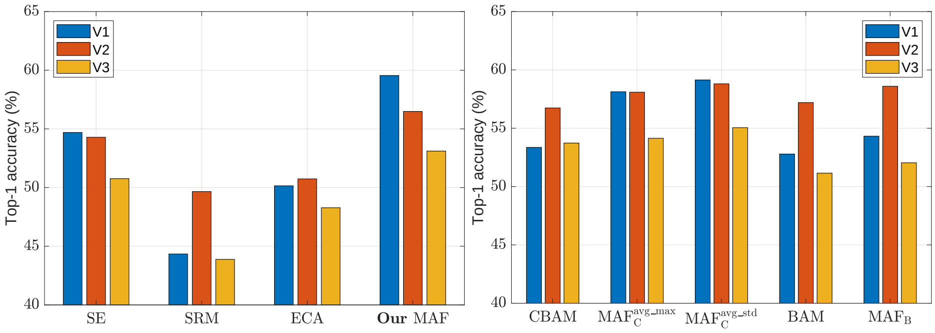
Fig. 5 Performances of MobileNetV1 addressing the proposed MAF module with its different width multipliers on Stanford Dogs (a) and ImageNet-100 (b) compared to MobileNetV1 addressing other channel-wise attentions.

[25]; V2 for MobileNetV2 [26]; and V3 for MobileNetV3 [27] in the below investigations of the efficiency of those attention modules. Furthermore, in case no multiplier is explicitly indicated for MobileNets, *width-multiplier* = 1.0 will be referred to. Due to the enormousness of ImageNet-1k, we would like to train the MAF-based networks with parameters generating the best results as discussed in Section 6.2.2.

Table 2 Top-1 accuracy (%) with different attention modules on Stanford Dogs and ImageNet-100.

			MobileNetV1				MobileNetV2				MobileNetV3			
Dataset	Attention		(x0.25)	(x0.50)	(x0.75)	(x1.0)	(x0.25)	(x0.50)	(x0.75)	(x1.0)	(x0.25)	(x0.50)	(x0.75)	(x1.0)
Stanford Dogs		No Attention	42.49	50.74	53.43	53.58	46.91	51.56	53.50	54.10	41.85	43.51	46.44	50.52
	Channel	SE [35]	48.53	53.87	53.67	54.69	46.27	49.65	53.00	54.29	40.64	48.68	48.36	50.76
		SRM [38]	36.02	43.15	43.88	44.34	34.62	46.17	49.14	49.66	37.03	42.63	42.58	43.88
		ECA [39]	37.13	43.33	44.31	50.15	40.73	43.74	49.55	50.74	39.93	42.18	46.94	48.28
		MAF	**50.76**	**55.95**	**58.16**	**59.55**	**47.88**	**52.34**	**55.79**	**56.49**	**42.14**	**49.66**	**49.76**	**53.11**
	Channel+Spatial	CBAM [36]	47.79	51.35	52.09	53.36	43.50	51.66	52.94	56.74	43.65	**50.91**	51.04	53.73
		$\mathbf{MAF_C^{avg\_max}}$	49.25	55.21	57.34	58.13	45.51	53.59	**56.46**	58.09	42.55	49.11	52.63	54.14
		$\mathbf{MAF_C^{avg\_std}}$	**52.16**	**57.59**	**59.06**	**59.14**	**48.19**	**54.12**	56.22	**58.81**	**44.90**	49.19	**54.04**	**55.05**
	Channel	BAM [37]	44.68	46.92	50.21	52.79	44.25	51.70	54.37	57.20	42.37	48.53	49.51	51.15
		$\mathbf{MAF_B}$	**48.12**	**53.23**	**54.22**	**54.32**	**49.22**	**52.69**	**54.39**	**58.60**	**44.27**	**49.83**	**51.45**	**52.04**
ImageNet-100		No Attention	69.96	74.87	75.59	77.01	69.73	74.33	75.68	77.87	**69.02**	74.32	76.39	77.60
	Channel	SE [35]	70.40	76.04	76.76	77.97	68.74	**74.77**	76.47	77.58	68.81	73.64	**76.67**	77.09
		SRM [38]	69.71	72.79	74.73	75.96	64.63	74.04	76.11	76.38	67.95	72.88	75.70	76.32
		ECA [39]	70.10	73.92	75.09	76.40	69.66	74.28	75.73	77.70	67.31	73.67	75.68	76.37
		MAF	**71.52**	**76.58**	**78.17**	**79.60**	**70.37**	74.56	**77.31**	**78.28**	68.57	**74.62**	76.43	**78.07**
	Channel+Spatial	CBAM [36]	70.74	74.47	78.03	78.12	68.37	73.85	76.68	77.62	68.59	73.52	76.32	77.26
		$\mathbf{MAF_C^{avg\_max}}$	71.64	75.26	78.38	79.09	69.14	73.75	77.18	78.31	69.35	73.07	75.81	77.68
		$\mathbf{MAF_C^{avg\_std}}$	**71.86**	**76.37**	**78.77**	**79.34**	**70.15**	**75.33**	**76.89**	**79.32**	**69.63**	**73.66**	**76.42**	**77.79**
	Channel	BAM [37]	70.57	74.22	**77.05**	**77.49**	**70.00**	**75.17**	77.15	78.29	68.35	**74.61**	76.45	78.24
		$\mathbf{MAF_B}$	**71.29**	**74.99**	76.03	76.79	69.72	74.66	**77.84**	**78.83**	**68.55**	73.86	**76.60**	**78.30**

Note: Numbers in parentheses indicate different width multipliers for model reduction of MobileNets.



(a) Channel-wise attentions (b) Spatial-channel-wise attentions

Fig. 6 Performances of the MAF-based modules on Stanford Dogs in comparison with other attentions embedded into MobileNets (i.e., *multiplier* = 1.0).

It can be observed in Tables 1 and 2 that the MAF-based networks obtained better performances for most multipliers of MobileNets in comparison with the original MobileNets and the other attention modules as well. This advance would be thanks to the efficient accumulation mechanism of the proposed DBR perceptron, which generated a fused attention map by taking advantage of the inter-channel feature relationships of S_{GAP} and S_{GSD}. Indeed, Table 1 shows that 0.5∼1% on CIFAR-10 and 1%∼1.5% on CIFAR-100 are higher results of MAF compared to the initial MobileNets

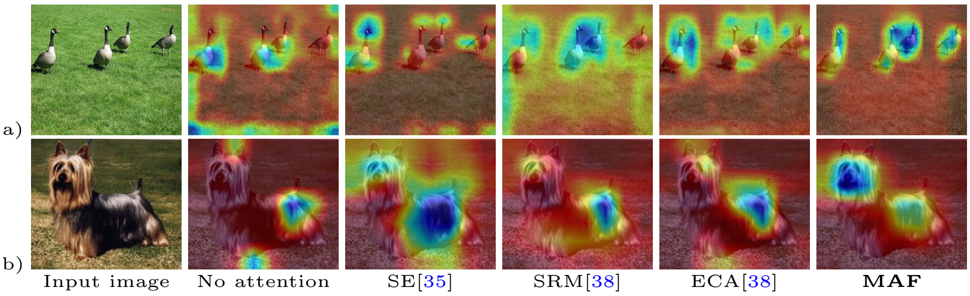


Fig. 7 An efficient attention of V1+MAF on ducks (a) and an Australian_terrier dog (b) in comparison with that of other modules. These visualizations are drawn out by Grad-CAM [63].

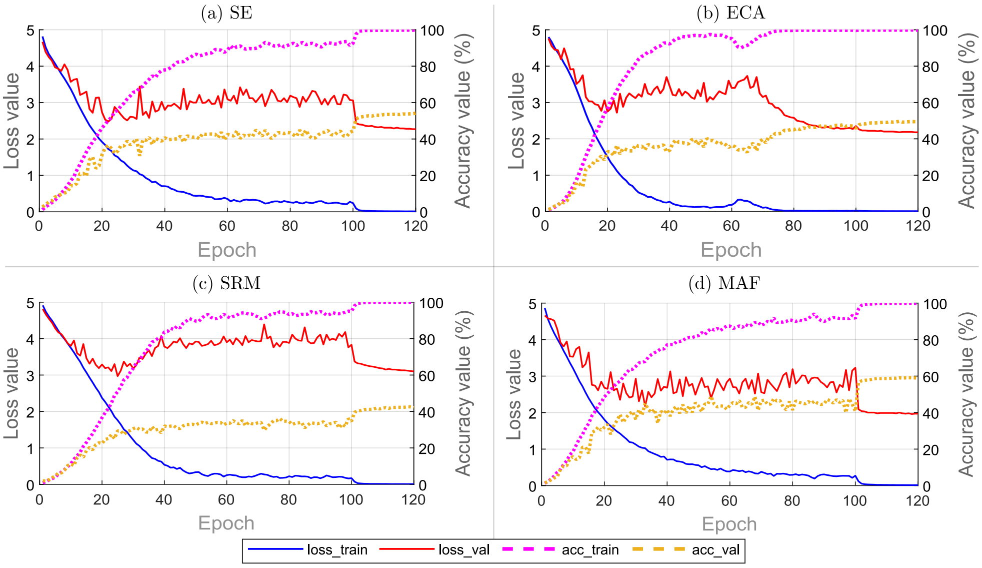


Fig. 8 Training and validating plots of the proposed V1+MAF (d) on Stanford Dogs compared to other channel-wise modules: (a) V1+SE [35], (b) V1+ECA [39], (c) V1+SRM [38].

(i.e., without attention for V1 and V2, while SE [35] with $r = 4$ for V3). On the challenging schemes, our MAF also achieved better rates: 1%-1.5% on ImageNet-100; while 1.5%-6% on Stanford Dogs (see Table 2), particularly, up to over 8% (i.e., 50.76% contrast to 42.49%) when embedding MAF into V1(x0.25) (see Fig. 5). Hereinafter, we will present evaluations of MAF's performances in detail.

6.2.1 Performance analyses of channel-wise patterns

It would be verified that the proposed accumulation of channel-wise patterns with the DBR perceptron produced accumulated-attentive information which enhances the efficiency of the training process of MobileNets [25–27]. Based on the experimental results in Tables 1 and 2, we can point out the following thorough assessments compared to the particular channel-wise modules: SE [35], SRM [38], and ECA [39].

Table 3 Complexity of MobileNetV2(x1.0) with different attention modules.

	CIFAR-100		Stanford Dogs		ImageNet-1k	
Attention	FLOPs	#Params	FLOPs	#Params	FLOPs	#Params
SE [35]	47.04M	2.38M	166.78M	2.41M	167.34M	3.54M
SRM [38]	47.03M	2.36M	166.77M	2.38M	167.33M	3.51M
ECA [39]	47.10M	2.35M	167.05M	2.38M	167.62M	3.51M
MAF	47.05M	2.40M	166.79M	2.42M	167.35M	3.55M
CBAM [36]	47.27M	2.38M	167.93M	2.41M	168.49M	3.54M
$\mathbf{MAF_C^{avg\_std}}$	47.26M	2.40M	167.92M	2.42M	168.49M	3.55M
BAM [37]	47.58M	2.41M	168.66M	2.44M	169.23M	3.57M
$\mathbf{MAF_B}$	47.59M	2.43M	168.67M	2.45M	169.23M	3.58M

Note: FLOPs and #Params are computed by using *ptflops* [64].

- Thanks to the accumulated channel-wise information, MAF can make the learning processes concentrate on the important information in more effectiveness than SE [35], SRM [38], and ECA [39]. Indeed, Tables 1 and 2 indicate the eminent results of MAF in most of the testing cases for MobileNets, particularly, those results with *width-multiplier* = 1 that are often addressed for real applications (see Fig. 6(a) for a visual view of the performances on Stanford Dogs). For instance, V1+MAF gained the best rates with 79.6% and 59.55% on ImageNet-100 and Stanford Dogs. Those can be because the attentive operation of MAF notoriously focused on ducks in an image of ImageNet-100 (see Fig. 7(a)), while the attention of MAF on the face of an Australian_terrier dog in Stanford Dogs rather on the other external anatomies of the dog (see Fig. 7(b)). Additionally, Fig. 8 indicates well-behaved curves of V1+MAF (d) against V1+SE [35] (a), V1+ECA [39] (b), and V1+SRM [38] (c). This leads to the prominent performances of MAF versus the others'.
- The experimental results have also verified that the attentive mechanism of SRM [38] and ECA [39] is suitable for ImageNet-100 rather than Stanford Dogs. For instance, about 76.3% is for both V3+SRM and V3+ECA on ImageNet-100, while just 43.88% and 48.28% are on Stanford Dogs, respectively (see Table 2 for more circumstances). It might be that they concentrated on allocating features of the dog's thigh instead of doing on those of its face, which are some of the distinguished anatomies for dog recognition (see Fig. 7(b)).
- It can be observed from Table 3 that the computational complexity of MAF is nearly the same as other channel-wise attentions. For instance of training on Stanford Dogs, V2+MAF takes into account 2.42M learnable parameters, a little higher than V2+SE [35] (2.41M), V2+SRM [38] (2.38M), and V2+ECA [39] (2.38M). Also, it is worth mentioning that V2+MAF (56.49%) obtained a significant image classification rate in comparison with V2+SE [35] (54.29%), V2+SRM (49.66%), and V2+ECA (50.74%). Refer to Tables 2 and 3 for further circumstances.

6.2.2 Performance analyses of spatial-channel-wise patterns

The replacement and integration of MAF into CBAM [36] and BAM [37] (i.e., $\mathrm{MAF_C}$ and $\mathrm{MAF_B}$ respectively) boosted the learning ability of image representation. This

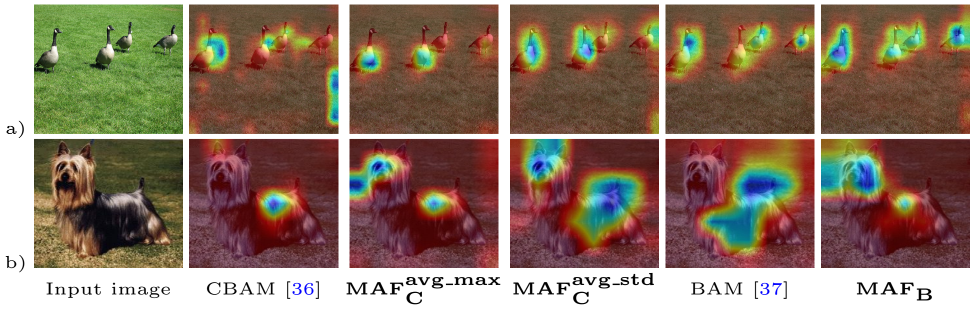


Fig. 9 An efficient attention of $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$ and MAF_B on ducks (a) and an Australian_terrier dog (b) in comparison with that of others. These visualizations are drawn out by Grad-CAM [63].

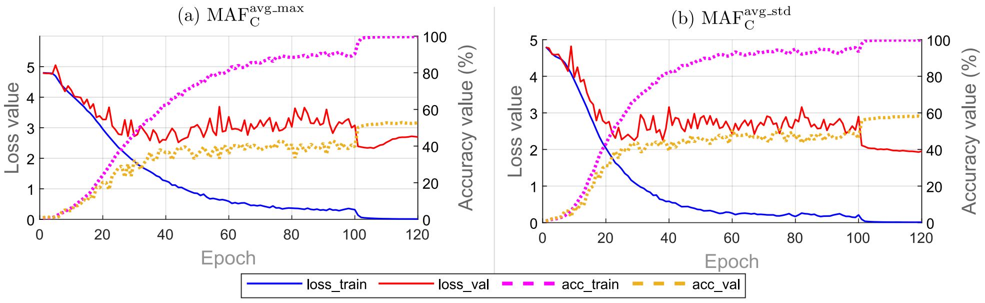


Fig. 10 Training and validating plots of V1+$\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$ (a) and V1+$\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_std}$ (b) on Stanford Dogs.

advance is thanks to the DBR-based accumulation of two channel-wise pattern blocks. To evaluate the accumulation of different channel-wise patterns, we investigate MAF_C with the DBR recalibration of S_{GAP} and the Global Max Pooling features (S_{GMP}) (called $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$); while the recalibration of S_{GAP} and S_{GSD} is called $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_std}$. Based on the experimental results in Tables 1 and 2, we can point out several crucial statements as

- The proposed DBR perceptron learns more efficiently than the multi-layer bottleneck perceptron (MLP) used in CBAM [36]. In other words, the proposed accumulation by learning contemporaneously channel-wise patterns can fuse more discriminative information than that by doing separately with an element-wise summation for the outputs. Indeed, $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$ pointed out higher performances thanks to replacing our DBR for MLP in CBAM [36] to accumulate S_{GAP} and S_{GMP}. Fig. 9 intuitively shows a convergent learning process of $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$ on images of the ducks and the Australian_terrier dog compared to CBAM, while Fig. 6(b) indicates its better classification rates. Also, refer to Tables 1 and 2 for more comparative results of CBAM and $\mathrm{MAF}_\mathrm{C}^\mathrm{avg\_max}$.
- The DBR-based fusion of S_{GAP} and S_{GSD} produced more solid discrimination than fusing S_{GAP} and S_{GMP}. It means that S_{GSD} can provide superior power of representative information for the attention mechanism. Fig. 10 indicates that the loss

validation of $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_max}}$ tends to increase from the 100^{th} epoch, while that of $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ decreases steadily. Furthermore, Fig. 9(a) $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ concentrated on all ducks, while Fig. 9(b) intuitively verifies a substantial attention of $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ with a wider region of the face and tail of the Australian_terrier dog in comparison with $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$.

- Addressing the DBR recalibration for BAM [37] also improves its performance. However, it is unstable for image classification on various datasets. $\mathrm{MAF}_{\mathrm{B}}$ is absolutely better on CIFAR-100 and Stanford Dogs, but not on the others (see Tables 1 and 2). This might be due to the loss of information caused by the spatial compression of DSC-based tensors across the channel dimension.
- In respect of evaluating the computational complexity, $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ and $\mathrm{MAF}_{\mathrm{B}}$ have nearly the same as the original ones, i.e., CBAM [36] and BAM [37] respectively. For instance, FLOPs and #Params of V2+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ on CIFAR-100 are 47.26M and 2.40M against 47.27M and 2.38M of CBAM [36]. In the meanwhile, those parameters of V2+$\mathrm{MAF}_{\mathrm{B}}$ are 47.59M and 2.43M versus 47.58M and 2.41M of BAM [37] (see more instances in Table 3).

Consequently, it can be conducted that $\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ achieved the best stability for most versions of MobileNets (see Tables 1 and 2). Therein, the performances of V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ will be taken into account for the below evaluations in comparison with the state of the art. V2+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ gave slightly lower results with smaller computational complexity. So there is a trade-off when considering them for implementation in practice. It means that V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ should be recommended in the case of ordering an absolutely high accuracy.

6.3 MAF-based performances compared to state of the art

It can be observed from Table 4 that MobileNetV1 [25], adapted by our MAF module, obtained 94.53% (V1+MAF) on CIFAR10 and 76.17% (V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$) on CIFAR-100, the highest rates compared to other V1-based ones. On CIFAR-100, it improved up to 2% while its trainable parameters are $\#Params$ = 4.04M, a little higher than the others. In addition, V2/V3+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ also obtained competitive results with a smaller number of learnable parameters. Several non-MobileNet-based models achieved better performances but took a large level of computational complexity, i.e., ResNet18 [22]+SE [35] (76.44% with 11.40M) and DenseNet-121 [24] (76.21% with 7.06M).

In terms of image classification on the challenging datasets, 59.55% of V1+MAF is the best rate on Stanford Dogs (see Table 5). Meantime, 73.13% of V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ on ImageNet-1k is the highest result in comparison with MobileNet-based networks. V1-BSConv [29] (59.10%) has nearly the same performance as ours on Stanford Dogs, but it is approximately 2% inferior on ImageNet-1k. Just taking into account a half smaller number of learnable parameters, V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ is up to about 3% better than several attention modules integrated into ResNet18 [22], i.e., (73.13%, 4.96M) of V1+$\mathrm{MAF}_{\mathrm{C}}^{\mathrm{avg\_std}}$ compared to (70.59%, 11.78M) of ResNet18+SE [35], (70.73%, 11.78M) of ResNet18+CBAM [36], and (71.12%, 11.71M) of ResNet18+BAM [37].

Table 4 Top-1 performance (%) on CIFAR-10/100.

	CIFAR-10		CIFAR-100	
Architecture	Top-1	#Params	Top-1	#Params
V1 Baseline [25]	93.76	3.22M	74.27	3.31M
V1-BSConv [29]	94.30	3.22M	75.70	3.31M
V1-GDF [28]	93.87	3.28M	74.48	3.37M
V1 + SE [35]	94.22	3.70M	75.37	3.80M
V1 + SRM [38]	94.26	3.24M	75.08	3.33M
V1 + ECA [39]	94.04	3.22M	73.96	3.31M
V1 + CBAM [36]	93.98	3.70M	74.98	3.80M
V1 + BAM [37]	93.63	4.22M	75.07	4.31M
V1+MAF	**94.53**	3.94M	75.95	4.04M
V1+MAF$_{\mathbf{C}}^{\mathbf{avg\_std}}$	94.35	3.94M	76.17	4.04M
V2+MAF$_{\mathbf{C}}^{\mathbf{avg\_std}}$	93.86	2.28M	74.82	2.40M
V3+MAF$_{\mathbf{C}}^{\mathbf{avg\_std}}$	94.21	2.72M	74.65	2.84M
ResNet18 Baseline [22]	93.02	11.17M	75.10	11.20M
ResNet18 [22] + SE [35]	-	-	**76.44**	11.40M
ShuffleNetV1 [34]	92.29	0.91M	70.06	1.00M
ShuffleNetV2(x1.5) [65]	93.93	2.49M	74.53	2.58M
AugShuffleNet [66]	93.63	1.21M	74.07	1.30M
DenseNet-121 [24]	94.23	6.96M	76.21	7.06M
SqueezeNet [67]	-	-	69.41	0.78M

Note: “-” means “not to be reported”. Comparative attentions on V1 are referred due to their good results, as discussed in Section 6.2.2.

Table 5 Top-1 performance (%) on Stanford Dogs and ImageNet-1k.

	Stanford Dogs		ImageNet-1k	
Architecture	Top-1	#Params	Top-1	#Params
V1 Baseline [25]	51.60	3.33M	66.90	4.23M
V1-BSConv [29]	59.10	3.33M	71.50	4.23M
V1-GDF [28]	54.90	3.39M	67.55	4.30M
V1 [25] + SE [35]	54.69	3.82M	70.03	4.72M
V1 [25] + CBAM [36]	53.36	3.82M	70.99	4.72M
V1 [25] + BAM [37]	52.79	4.33M	69.42	5.23M
V2 Baseline [26]	56.01	2.37M	67.05	3.51M
V3-small [27]	49.40	1.16M	64.40	2.90M
V3-large [27]	54.90	3.09M	71.50	5.40M
V1+MAF	**59.55**	4.04M	72.71	4.96M
V1+MAF$_{\mathbf{C}}^{\mathbf{avg\_std}}$	59.14	4.06M	73.13	4.96M
ResNet18 Baseline [22]	-	-	70.40	11.69M
ResNet18 [22] + SE [35]	-	-	70.59	11.78M
ResNet18[22] + CBAM[36]	-	-	70.73	11.78M
ResNet18 [22] + BAM [37]	-	-	71.12	11.71M
ShuffleNetV2 [65]	-	-	69.36	2.30M
DenseNet-121 [24]	56.90	7.08M	74.43	7.98M
EfficientNet-B0 [32]	54.70	4.16M	**77.10**	5.30M
MnasNet [31]	54.80	3.26M	75.20	4.38M
SqueezeNet [67]	-	-	41.90	1.25M

Note: “-” means “not available” or “not to be reported”.

Those have validated the prominence of the proposed MAF-based feature attention on different datasets. Furthermore, it should be noted that DenseNet-121 [24], EfficientNet-B0 [32], and MnasNet [31] have higher performance on ImageNet-1k, but they often own two disadvantages: *i)* requiring a larger number of trainable parameters [24, 32], *ii)* their learning ability is sharply weakened on Stanford Dogs (see Table 5).

7 Conclusion

We have presented an efficient channel-wise attention module (MAF) utilizing the proposed DBR perceptron for a fusion-excitation operation. The global average and the standard deviation information of an input tensor were accumulated to produce a volume of robust lightweights. To take more spatial-wise information, MAF was addressed as a potent channel-wise component replacing the original one of two critical attentions (CBAM [36] and BAM [37]) to form two corresponding modules, i.e., MAF_C and MAF_B. The experimental results on image classification have verified the appreciable efficiency of MAF-based MobileNets in comparison with the other spatial/channel-wise attention modules, particularly the adaption of MAF for image classification on the challenging datasets (i.e., Stanford Dogs and ImageNet). According to the comprehensive evaluations, $\text{V1+MAF}_\text{C}^\text{avg\_std}$ should be recommended for real applications requiring strictly high accuracy.

For perspectives, it can investigate an extra-fused attentive mechanism by taking advantage of our DBR perceptron to simultaneously accumulate S_{GAP}, S_{GSD}, S_{GMP}, and other channel-wise patterns (e.g., power-average pooling information), while maintaining the model complexity at a reasonable level.

References

[1] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS, pp. 1106–1114 (2012)

[2] Rayhan, F., Galata, A., Cootes, T.F.: Choicenet: CNN learning through choice of multiple feature map representations. Pattern Analysis and Applications **24**(4), 1757–1767 (2021)

[3] Yao, X., Song, T.: Rotation invariant gabor convolutional neural network for image classification. Pattern Recognition Letters **162**, 22–30 (2022)

[4] Hörhan, M., Eidenberger, H.: Gestalt descriptions for deep image understanding. Pattern Analysis and Applications **24**(1), 89–107 (2021)

[5] Park, S., Yang, S., Lee, H.: Mvdet: multi-view multi-class object detection without ground plane assumption. Pattern Analysis and Applications **26**(3), 1059–1070 (2023)

[6] Ma, Y., Wang, C.: Sdcnet for object recognition. Comput. Vis. Image Underst. **215**, 103332 (2022)

[7] Al-Faris, M., Chiverton, J.P., Yang, Y., Ndzi, D.: Multi-view region-adaptive multi-temporal DMM and RGB action recognition. Pattern Analysis and Applications **23**(4), 1587–1602 (2020)

[8] Wu, Z., Yan, H.: Adaptive dynamic networks for object detection in aerial images. Pattern Recognition Letters **166**, 8–15 (2023)

[9] Chen, S., Zhang, Y., Yin, B., Wang, B.: TRFH: towards real-time face detection and head pose estimation. Pattern Analysis and Applications **24**(4), 1745–1755 (2021)

[10] Maglietta, R., Amoroso, N., Boccardi, M., Bruno, S., Chincarini, A., Frisoni, G.B., Inglese, P., Redolfi, A., Tangaro, S., Tateo, A., Bellotti, R.: Automated hippocampal

segmentation in 3d MRI using random undersampling with boosting algorithm. Pattern Analysis and Applications **19**(2), 579–591 (2016)

[11] Jia, D., Cao, J., Pan, J., Pang, Y.: Multi-stream densely connected network for semantic segmentation. IET Comput. Vis. **16**(2), 180–191 (2022)

[12] Hu, X., Feng, J., Gong, J.: Lffnet: lightweight feature-enhanced fusion network for real-time semantic segmentation of road scenes. Pattern Analysis and Applications **27**(1), 27 (2024)

[13] Zhou, Q., Feng, Z., Gu, Q., Cheng, G., Lu, X., Shi, J., Ma, L.: Uncertainty-aware consistency regularization for cross-domain semantic segmentation. Comput. Vis. Image Underst. **221**, 103448 (2022)

[14] Sun, Y., Su, L., Luo, Y., Meng, H., Li, W., Zhang, Z., Wang, P., Zhang, W.: Global mask R-CNN for marine ship instance segmentation. Neurocomputing **480**, 257–270 (2022)

[15] Mecheter, I., Abbod, M., Zaidi, H., Amira, A.: Brain MR images segmentation using 3d CNN with features recalibration mechanism for segmented CT generation. Neurocomputing **491**, 232–243 (2022)

[16] Lei, B., Yang, M., Yang, P., Zhou, F., Hou, W., Zou, W., Li, X., Wang, T., Xiao, X., Wang, S.: Deep and joint learning of longitudinal data for alzheimer's disease prediction. Pattern Recognition **102**, 107247 (2020)

[17] Geng, C., Tao, L., Chen, S.: Guided CNN for generalized zero-shot and open-set recognition using visual and semantic prototypes. Pattern Recognition **102**, 107263 (2020)

[18] Gomes, M.E.N., Macêdo, D., Zanchettin, C., Mattos Neto, P.S.G., Oliveira, A.L.I.: Multi-human fall detection and localization in videos. Comput. Vis. Image Underst. **220**, 103442 (2022)

[19] Frejlichowski, D., Gosciewska, K., Forczmanski, P., Hofman, R.: Application of foreground object patterns analysis for event detection in an innovative video surveillance system. Pattern Analysis and Applications **18**(3), 473–484 (2015)

[20] Chao, L., Wang, Z., Zhang, H., Xu, W., Zhang, P., Li, Q.: Sparse-view cone beam CT reconstruction using dual cnns in projection domain and image domain. Neurocomputing **493**, 536–547 (2022)

[21] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)

[22] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)

[23] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR, pp. 1–9 (2015)

[24] Huang, G., Liu, Z., Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR, pp. 2261–2269 (2017)

[25] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. CoRR **abs/1704.04861** (2017)

[26] Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., Chen, L.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: CVPR, pp. 4510–4520 (2018)

[27] Howard, A., Pang, R., Adam, H., Le, Q.V., Sandler, M., Chen, B., Wang, W., Chen, L., Tan, M., Chu, G., Vasudevan, V., Zhu, Y.: Searching for mobilenetv3. In: ICCV, pp. 1314–1324 (2019)

[28] Nguyen, T.T., Nguyen, T.P.: Assembling extra features with grouped pointwise convolutions for mobilenets. In: DICTA, pp. 265–272 (2023)

[29] Haase, D., Amthor, M.: Rethinking depthwise separable convolutions: How intra-kernel correlations lead to improved mobilenets. In: CVPR, pp. 14588–14597 (2020)

[30] Wang, M., Liu, B., Foroosh, H.: Factorized convolutional neural networks. In: ICCV Workshops, pp. 545–553 (2017)

[31] Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V.: Mnasnet: Platform-aware neural architecture search for mobile. In: CVPR, pp. 2820–2828 (2019)

[32] Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML, vol. 97, pp. 6105–6114 (2019)

[33] Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: CVPR, pp. 1800–1807 (2017)

[34] Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: CVPR, pp. 6848–6856 (2018)

[35] Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR, pp. 7132–7141 (2018)

[36] Woo, S., Park, J., Lee, J., Kweon, I.S.: CBAM: convolutional block attention module. In: ECCV, vol. 11211, pp. 3–19 (2018)

[37] Park, J., Woo, S., Lee, J., Kweon, I.S.: BAM: bottleneck attention module. In: BMVC, p. 147 (2018)

[38] Lee, H., Kim, H., Nam, H.: SRM: A style-based recalibration module for convolutional neural networks. In: ICCV, pp. 1854–1862 (2019)

[39] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: Efficient channel attention for deep convolutional neural networks. In: CVPR, pp. 11531–11539 (2020)

[40] Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR (2016)

[41] Wang, X., Girshick, R.B., Gupta, A., He, K.: Non-local neural networks. In: CVPR, pp. 7794–7803 (2018)

[42] Buades, A., Coll, B., Morel, J.: A non-local algorithm for image denoising. In: CVPR, pp. 60–65 (2005)

[43] Chen, Y., Kalantidis, Y., Li, J., Yan, S., Feng, J.: Aˆ2-nets: Double attention networks. In: NeurIPS, pp. 350–359 (2018)

[44] Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H.: Dual attention network for scene segmentation. In: CVPR, pp. 3146–3154 (2019)

[45] Cao, Y., Xu, J., Lin, S., Wei, F., Hu, H.: Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In: ICCV Workshops, pp. 1971–1980 (2019)

[46] Chen, B., Huang, Y., Xia, Q., Zhang, Q.: Nonlocal spatial attention module for image classification. International Journal of Advanced Robotic Systems **17**(5) (2020)

[47] Hu, J., Shen, L., Albanie, S., Sun, G., Vedaldi, A.: Gather-excite: Exploiting feature context in convolutional neural networks. In: NeurIPS, pp. 9423–9433 (2018)

[48] Jin, X., Xie, Y., Wei, X., Zhao, B., Chen, Z., Tan, X.: Delving deep into spatial pooling for squeeze-and-excitation networks. Pattern Recognition **121**, 108159 (2022)

[49] Li, X., Wang, W., Hu, X., Yang, J.: Selective kernel networks. In: CVPR, pp. 510–519 (2019)

[50] Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., Li, M., Smola, A.J.: Resnest: Split-attention networks. In: CVPR Workshops, pp. 2735–2745 (2022)

[51] Qin, Z., Zhang, P., Wu, F., Li, X.: Fcanet: Frequency channel attention networks. In: ICCV, pp. 763–772 (2021)

[52] Yang, L., Zhang, R., Li, L., Xie, X.: Simam: A simple, parameter-free attention module for convolutional neural networks. In: ICML, vol. 139, pp. 11863–11874 (2021)

[53] Bello, I., Zoph, B., Le, Q., Vaswani, A., Shlens, J.: Attention augmented convolutional networks. In: ICCV, pp. 3285–3294 (2019)

[54] Zhang, H., Zu, K., Lu, J., Zou, Y., Meng, D.: Epsanet: An efficient pyramid squeeze attention block on convolutional neural network. In: ACCV, vol. 13843, pp. 541–557 (2022)

[55] Huang, X., Belongie, S.J.: Arbitrary style transfer in real-time with adaptive instance normalization. In: ICCV, pp. 1510–1519 (2017)

[56] Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: ICML, pp. 807–814 (2010)

[57] Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., Chua, T.: SCA-CNN: spatial and channel-wise attention in convolutional networks for image captioning. In: CVPR, pp. 6298–6306 (2017)

[58] Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. In: Tech Report (2009)

[59] Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)

[60] Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR, pp. 248–255 (2009)

[61] Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. In: ECCV, vol. 12356, pp. 776–794 (2020)

[62] Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: CVPR Workshop (2011)

[63] Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. Int. J. Comput. Vis. **128**(2), 336–359 (2020)

[64] Sovrasov, V.: Ptflops: a Flops Counting Tool for Neural Networks in Pytorch Framework. https://github.com/sovrasov/flops-counter.pytorch

[65] Ma, N., Zhang, X., Zheng, H., Sun, J.: Shufflenet V2: practical guidelines for efficient CNN architecture design. In: ECCV, vol. 11218, pp. 122–138 (2018)

[66] Ye, L.: AugShuffleNet: Communicate More, Compute Less. CoRR **abs/2203.06589** (2022)

[67] Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size. CoRR **abs/1602.07360** (2016)