LAM: Lightweight Attention Module

Qiwei Ji[1,3], Bo Yu[1,3](✉), Zhiwei Yang[2,3](✉), and Hechang Chen[1,3](✉)

[1] School of Artificial Intelligence, Jilin University, Changchun, China
{jiqw20,byu20}@mails.jlu.edu.cn, chenhc@jlu.edu.cn

[2] College of Computer Science and Technology, Jilin University, Changchun, China
yangzw18@mails.jlu.edu.cn

[3] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Changchun, China

Abstract. The attention mechanisms have been widely used in existing methods due to their effectiveness. In the field of computer vision, these mechanisms can be grouped as 1) channel attention mechanisms, which highlight the important channels for images, and 2) spatial attention mechanisms, which focus on the location features for all channels of images. These two groups of mechanisms, which have various strategies for capturing features, actually play complementary roles in image classification. Existing lightweight models based on one group of attention mechanisms have fewer parameters than convolutional networks. However, few works consider their integration and maintain their merits for lightweight neural networks. In this paper, we propose a new Lightweight Attention Module (LAM) for lightweight convolutional neural networks to efficiently integrate these attention mechanisms. Specifically, we use element-wise addition and smaller convolutional kernels in the spatial module, avoiding the vanishing gradient problem. Besides, we replace the multi-layer perceptron (MLP) layer with squeeze-and-excitation layers in the channel module, alleviating the problem of channel dependencies. Finally, we adopt a parallel mechanism to coordinate these two attention modules with low computational complexity. Experimental results on benchmark datasets demonstrate the effectiveness of LAM in terms of image classification tasks, ablation study and robustness analysis.

Keywords: Attention mechanism · Lightweight model · Convolutional neural networks

1 Introduction

The attention mechanisms [1] can well improve models' accuracy by capturing key information of pictures, e.g., find 'where' and 'what' to focus on. As an effective component of neural networks [2], attention modules have shown good

We truly thank the reviewers for their great effort and pertinent comments on our submission. This work is supported in part by the National Natural Science Foundation of China (No. 61902145), the National Key R&D Program of China (2021ZD0112501, 2021ZD0112502).

G. Memmi et al. (Eds.): KSEM 2022, LNAI 13369, pp. 485–497, 2022.
https://doi.org/10.1007/978-3-031-10986-7_39

performances in various visual tasks, including image classification [3], object detection [4], semantic segmentation [5] and object tracking [6].

Existing studies introduce two kinds of fundamental attention modules widely used in computer vision: channel and spatial attention modules [7]. These two modules strengthen the representations by combining the feature maps from all the positions with different strategies. There have been many useful implemental architectures for these years. For channel attention modules, Jie Hu et al. [8] automatically recalibrate channel-wise feature reflections by explicitly modeling interdependencies between channels. Xiang Li et al. [9] employ a dynamic selection mechanism that enables every cell to automatically adjust its receptive field size on the basis of multiple scales of input representations. For spatial attention modules, Jun Fu et al. [10] encode a wider range of contextual information into local features, which improves their representative capability. Moreover, researchers try to aggregate both of the attention mechanisms, Sanghyun et al. [7] sequentially infer attention maps along the channel and spatial dimensions, then the attention maps are multiplied to the input feature maps for adaptive feature refinement. All these methods introduce attention modules for the neural networks to learn feature representations of images.

However, the above attention modules are designed mainly for normal networks. When adapted to lightweight models [11], they usually have various kinds of problems. First, the neural networks with single spatial or channel attention modules, like SENet [8], may ignore the other dimensions' information. They don't make full use of other dimensions' representations of images, and the lightweight models with rare parameters can't absorb the information well, which leads to poor performances. Second, the complex mixed architecture like CBAM [7] violates the principle of lightweight models, which will result in poor efficiency. Specifically, it concats mean and max spatial feature maps. After convolution operations with a large kernel, the concated feature maps will return to the original size, and multiply with the initial feature map. The complex concat and convolution functions respond to the vanishing gradient problem. Therefore, it will be beneficial to visual tasks by incorporating the information of the two dimensions in a simple and effective architecture.

In view of this, we propose a novel attention module called Lightweight Attention Module (LAM). For the spatial part, we use element-wise addition to process the average and max pooled feature maps, and use a smaller convolutional kernel to extract features. For the channel part, we also add the max-pooling and average-pooling feature maps first, then use the squeeze-and-excitation layers [8] to extract features. At last, we add the two output feature maps in a parallel arrangement. Overall, our model simplifies extensive convolution operations, which may cause vanishing gradient problems in previous modules. Meanwhile, we use the parallel instead of the traditional sequential arrangement [7]. As a result, our model efficiently helps the information flow into the next layer within the lightweight neural networks by learning which points to emphasize.

The key contributions can be summarized as follows:

- A novel lightweight attention module called LAM is proposed, which is capable of capturing information by incorporating the features of the channel and spatial dimensions with a parallel arrangement.
- The superiority of the LAM is demonstrated compared with the previous methods using image classification datasets, and in-depth analysis gives the rationality and robustness of the proposed method.

2 Related Work

In this section, we introduce the related works in the area of lightweight neural networks and attention mechanisms separately.

2.1 Lightweight Model

Since AlexNet [12] had excellent performances on the ImageNet competition in 2012 [13], deep neural networks started to explode researchers' interest again. The 2014 ImageNet champion GoogleNet [14] got 74.8 top-1 accuracy. Afterwards, In The 2017 ImageNet match, SENet [8] won the game with 82.7 top-1 accuracy. However, these models are too big to be applied in our real life. These models have reached the hardware limitations. So experts started to reduce the size of model by gaining efficiency in place of accuracy. Since smart phones get popular, there comes various efficient lightweight models like ShuffleNet [15,16], MobileNet [17–19], and EfficientNet [20]. Later, neural architecture search (NAS) [21] cut a striking figure in designing lightweight models. They perform better than the hand-crafted neural networks by adapting the models' width, channels, kernels and sizes. While most of the neural network designing ways focus on the aspects of depth, width and cardinality, we care about the other influence factor, 'attention', which draws lessons from human visual system.

2.2 Attention Module

Attention is one of the most important concepts in the deep learning field [22,23], inspired by human visual system that cannot manipulate all the information of the same image immediately [24]. As a replacement, people use a series of partial scans and conditionally pay attention to the obvious part for more information.

Recently, there have been many experts trying to combine the channel and spatial attention modules with models for real-world tasks. RAN (Residual Attention Network) [25] makes use of an encoder and decoder to make up attention module. Through purifying the feature maps, the model gets high accuracy even faced up with noisy datasets. Instead of processing the whole 3D attention feature maps, we resolve the procedure that comprehends channel and spatial representations respectively. The single attention-generating part for 3D feature maps has fewer parameters, and the end-to-end design enables it to be a plug-and-play module, which is very suitable for existing lightweight deep neural networks.

Close to our work, CBAM illustrates a channel and spatial mixed module to find the inner relationships of various feature maps. In CBAM's channel part, it uses MLP layers to get global average features for channel-wise attention. But we find that the linear layers for inferring attention maps may affect the feature extraction process in the lightweight models, so we replace them and use squeeze-and-excitation module, which has better performances both in speed and feature capturing ability. Similar to the channel part, we also delete the 7×7 convolutional kernel, which multiplies with the concat map, and that may cause vanishing gradient problems. Instead, we use a smaller kernel to multiply with the overlying map. In our LAM, we employ both channel and spatial attention in a simplified way intended for lightweight networks. The experiments verify that LAM not only improves the accuracy but also considers the handiness.

3 Methodology

In this section, we propose an attention module intended for lightweight neural networks called LAM. To understand the module, we first introduce the overall framework of the algorithm, then the channel attention module, the spatial attention module and the arrangement of attention modules part, respectively.

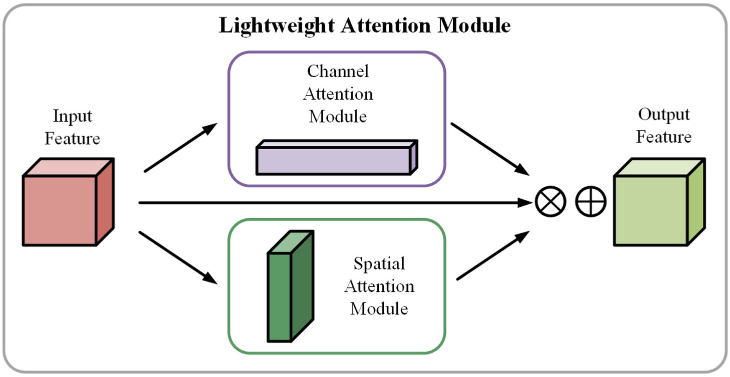


Fig. 1. Overview of LAM. The module has two parallel parts: channel and spatial. The feature map is refined by our module at every convolutional step of neural networks.

3.1 Attention Module

LAM is built on a transformation, which separately uses a one dimension channel attention map $\mathbf{M}_c \in \mathbf{R}^{C\times1\times1}$ and a two dimension spatial attention map $\mathbf{M}_s \in \mathbf{R}^{1\times H\times W}$ to map an input $\mathbf{X} \in \mathbf{R}^{C\times H\times W}$ to feature maps $\mathbf{U} \in \mathbf{R}^{C\times H\times W}$ as shown in Fig. 1. The computational procedure can be written as:

$$\mathbf{X}_1 = \mathbf{M}_c(\mathbf{X})\otimes\mathbf{X} \tag{1}$$

$$\mathbf{X}_2 = \mathbf{M}_s(\mathbf{X}) \otimes \mathbf{X} \tag{2}$$

$$\mathbf{U} = \mathbf{X}_1 \oplus \mathbf{X}_2 \tag{3}$$

where $\otimes$ means element-wise multiplication, $\oplus$ means element-wise addition. In the computational process, the attention maps are dispersed to all the dimensions: spatial attention maps are broadcasted along the channel dimension, and vice versa. $\mathbf{U}$ is the final output result. The following part will detailedly talk about the core of each attention module and computation process.

3.2 Channel Attention Module

In the LAM module, we use a channel attention computational unit to dig the inner channel information of feature representations. Because every channel is seen as a detector, channel attention pays attention to what is the important part of the images. To get the channel attention maps efficiently, we use the squeeze-and-excitation part to process the input feature representations. In the squeeze-and-excitation block, they use global squeeze information in a channel descriptor [26] to solve the problem of channel dependencies. And the squeeze part uses average-pooling in their module to get spatial statistics. To obtain the feature representations in the squeeze part, they use a second operation for fully capturing channel-wise dependencies. They employ a simple gating mechanism [27], which consists of a dimensionality reduction and increasing layer returning to the channel dimension of the transformation, with sigmoid activation.

In CBAM, they propose that max-pooling can help collect another key clue about obvious objects to obtain better channel-wise attention. So they use both average and max pooling to process feature maps simultaneously, which highly improves the effectiveness of models.

Different from these works, we argue that although the two pooling ways do improve the ability to capture key features, the linear layers in MLP deeply affect the simple architecture of lightweight neural networks and bring extra computation. So we remove the linear and activation layer and use the convolutional layers to multiply with feature maps. We describe the detailed operation below (Fig. 2).

We first aggregate spatial features by using both max and average pooling to generate two kinds of spatial descriptors: $\mathbf{X}^c_{avg}$ and $\mathbf{X}^c_{max}$, which represent max and average features along the spatial dimensions. These two descriptors are then forwarded by a squeeze-and-excitation module to generate the final attention map $\mathbf{M}_c \in \mathbf{R}^{C\times1\times1}$. The squeeze-and-excitation module consist of a dimensionality reduction and increasing convolutional layer to decrease parameters overhead. After the squeeze-and-excitation layer is applied to both descriptors, we add the output vectors with element-wise addition. The channel attention module can be summarized as follows:

$$\begin{aligned}\mathbf{M}_c(\mathbf{X}) &= \sigma(SE(AvgPool(\mathbf{X})) + SE(MaxPool(\mathbf{X}))) \\ &= \sigma(\mathbf{W}_1(\mathbf{W}_0(\mathbf{X}^c_{avg})) + \mathbf{W}_1(\mathbf{W}_0(\mathbf{X}^c_{max})))\end{aligned} \tag{4}$$

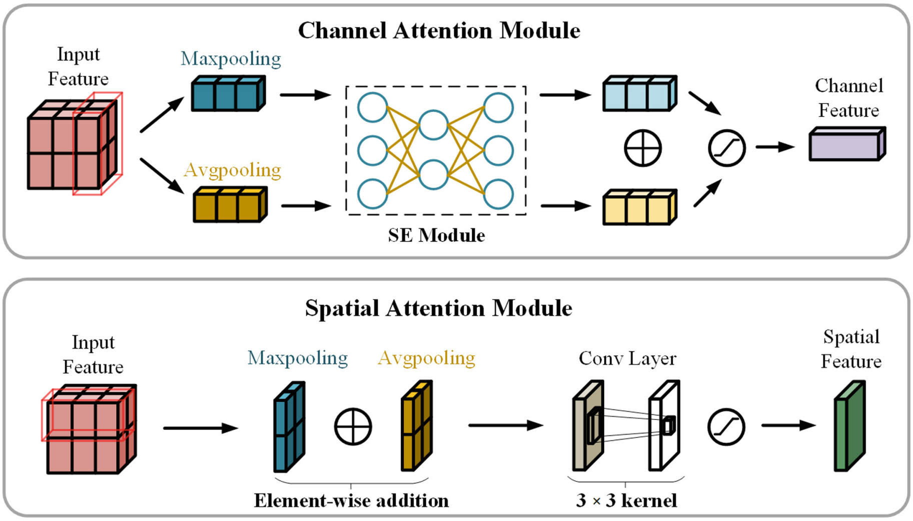


Fig. 2. Diagram of each attention sub-module. The figure shows that the channel sub-module emphasizes both max-pooling outputs and average-pooling outputs with a squeeze-and-excitation architecture. The spatial sub-module emphasizes the channel and spatial output features that are pooled along the channel axis and forward them to a convolutional layer.

where σ means the sigmoid function, $\mathbf{W}_0 \in \mathbf{R}^{C/r \times C}$, $\mathbf{W}_1 \in \mathbf{R}^{C \times C/r}$. And the squeeze-and-excitation weight $\mathbf{W}_1$ and $\mathbf{W}_1$ are shared for both input feature maps. Pay attention that the ReLU activation function is followed by $\mathbf{W}_0$.

3.3 Spatial Attention Module

In the LAM module, we get a spatial attention map by exploiting the inner spatial relationships of feature vectors. Unlike the channel attention module, the spatial attention module pays more attention to the location of images, which is seen as an important part complementary to the channel attention module. We also use average-pooling and max-pooling computational operations across the channel axis. And then sum them to get an efficient feature descriptor. On the summed feature descriptor, we feed it into a convolutional layer to get a spatial attention map $\mathbf{M}_s(\mathbf{F}) \in \mathbf{R}^{H \times W}$, which contains the information where to emphasize. We describe the detailed operation below.

We first aggregate channel features by using both max-pooling and average-pooling to generate two kinds of 2D feature maps: $\mathbf{M}^s_{avg}(\mathbf{X}) \in \mathbf{R}^{1 \times H \times W}$ and $\mathbf{M}^s_{max}(\mathbf{X}) \in \mathbf{R}^{1 \times H \times W}$, which represent average and max features across the channel dimensions respectively. The vectors are then added with element-wise addition and convolved by a standard convolutional layer to get the final 2D feature maps. The spatial attention module can be summarized as follows:

$$\begin{aligned}\mathbf{M}_c(\mathbf{X}) &= \sigma(f^{3\times 3}(AvgPool(\mathbf{X})) + MaxPool(\mathbf{X})) \\ &= \sigma(f^{3\times 3}(\mathbf{X}^c_{avg})) + (\mathbf{X}^c_{max})))\end{aligned} \tag{5}$$

where σ means the sigmoid function. $f^{3\times 3}$ means a convolutional layer with the kernel size of 3×3.

3.4 Arrangement of Attention Modules

The two attention modules pay attention to the channel and spatial dimension separately with complementary computing attention. Unlike CBAM, we adopt a parallel arrangement. The sequential arrangement will affect the lightweight in a bad way, whether it is channel-first order or spatial first-order. We will discuss experimental results in the next section (Fig. 3).

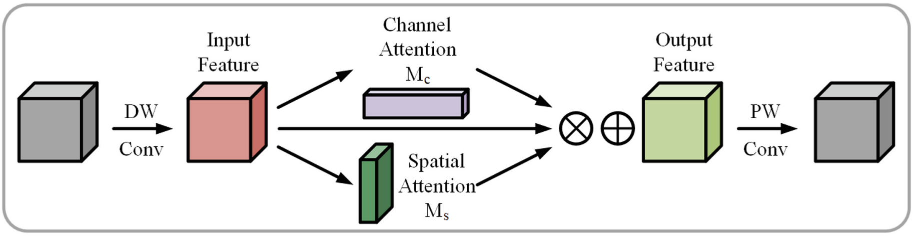


Fig. 3. LAM is integrated with a Basicblock in lightweight models. This figure shows the exact position of our module when integrated within a depthwise separable block [17]. We apply LAM to the convolution outputs in each block.

4 Experiment

In this section, we first introduce the experiment settings, including datasets, baselines and evaluation methods. Then we evaluate the effectiveness of the LAM in image classification tasks and ablation studies. In general, we seek to answer the following questions:

- **Q1:** Does the LAM have better performances than other attention modules when adapted to lightweight neural networks?
- **Q2:** Is the current arrangement most suitable for the LAM?
- **Q3:** Is the proposed LAM model sensitive to the main parameters, e.g., the learning ratio and kernel size?

4.1 Datasets and Experimental Setting

We conduct experiments on two standard image classification datasets [28]: Cifar10 and Cifar100 [29]. Both of the two standard datasets comprise a collection of 50k training and 10k test 32 × 32 pixel RGB real-world images, labelled with 10 and 100 classes respectively.

Besides, experiments use the same data augmentations [30,31] and parameter settings. The input images are randomly horizontally flipped and zero-padded on each side with 4 pixels before taking a random 32 × 32 cropping operation. We also adopt mean and standard deviation normalization. We use the Top-1 accuracy to compare our model with other baselines for image classification tasks. We use the authors' released code for baseline models.

Baselines and Metrics Here we present some existing lightweight neural networks as baselines, which proves that the LAM fits into lightweight models well.

- `MobileNetV1` [17]: This model uses depthwise separable convolutions to decrease computational complexity.
- `MobileNetV2` [18]: It builds inverted residuals and linear bottlenecks to filter features as a source of non-linearity.
- `MobileNetV3` [19]: It uses a combination of complementary search techniques as well as a novel architecture ang.
- `ShuffleNetV1` [15]: It utilizes pointwise group convolution and channel shuffle operations to greatly reduce computation cost.
- `ShuffleNetV2` [16]: It uses an additional convolutional layer right before global averaged pooling to mix up features.
- `EfficientNet` [20]: It uses neural architecture search to design a new baseline network and scale it up.

Here we present some attention modules as baselines, which can prove that the LAM has better performances than other attention modules when applied to the lightweight models.

- `Squeeze-and-excitation module` [8]: It adaptively recalibrates channel-wise feature responses by explicitly modelling interdependencies between channels.
- `Convolutional block attention module` [7]: It emphasizes meaningful features along those two principal dimensions: channel and spatial axes.

4.2 Results and Analysis (Q1)

To address the first question (Q1), we conduct experiments to measure the LAM's quality and compare it with other baseline methods. We train the models on the training dataset and test them on the validation dataset. PyTorch and Adam optimizer are used in our model (Learning Rate = 0.05, Weight Decay = 0.0001, Batch Size = 64). We report the average results of Top-1 accuracy by

Table 1. Accuracy of image classification in lightweight neural networks

Architecture	Params	MFlops	Cifar10 Top-1 acc	Cifar100 Top-1 acc
MobileNetV1	13.12	356.16	88.33	66.48
MobileNetV1 + CBAM	15.56	359.42	–[a]	–
MobileNetV1 + SE	18.49	360.05	89.12	67.30
MobileNetV1 + LAM	18.18	360.23	**89.37**	**68.09**
MobileNetV2	2.91	32.14	90.54	64.15
MobileNetV2 + CBAM	3.09	33.34	90.69	64.61
MobileNetV2 + SE	3.20	32.54	90.17	62.38
MobileNetV2 + LAM	3.17	32.63	**91.08**	**64.97**
MobileNetV3small	4.98	35.55	78.01	66.64
MobileNetV3small + CBAM	4.90	35.78	–	–
MobileNetV3small + LAM	4.95	35.55	**79.46**	**67.75**
MobileNetV3large	10.69	158.85	84.22	64.23
MobileNetV3large + CBAM	10.52	158.98	–	–
MobileNetV3large + LAM	10.67	158.85	**84.66**	**64.35**
ShuffleNetV1G3	4.07	91.79	85.12	**68.91**
ShuffleNetV1G3 + CBAM	4.16	92.33	–	–
ShuffleNetV1G3 + SE	4.40	92.09	83.06	65.83
ShuffleNetV1G3 + LAM	4.34	92.01	**85.24**	68.89
ShuffleNetV2	5.50	90.46	83.28	65.32
ShuffleNetV2 + CBAM	5.61	92.03	–	–
ShuffleNetV2 + SE	5.99	91.84	83.26	64.53
ShuffleNetV2 + LAM	5.62	90.83	**83.87**	**66.07**
EfficientNetB0	20.95	20.16	88.51	72.48
EfficientNetB0 + CBAM	21.38	22.03	79.32	62.53
EfficientNetB0 + LAM	21.54	21.54	**88.75**	**72.79**

[a] The empty result means vanishing problem happens in the models.

running the model 100 epochs [32]. The learning rate drops at the 40_{th}, 60_{th} and 80_{th} epoch by a factor of 10.

Table 1 summarizes the experimental results. The networks with LAM outperform all the baselines significantly, showing that the LAM can generalize well on lightweight models in the image classification datasets. Moreover, the models with LAM improve the accuracy compared to other attention modules. Firstly, although the CBAM improves the accuracy when applied in the MobileNetV2, it results in vanishing gradient problems in other models due to the complicated convolutional layers. So we use both channel and spatial attention modules but simplify convolutional parts. Secondly, the MobileNetV3 and EfficientNet use the squeeze-and-excitation layer to extract features, which provides a significant improvement. But it's not robust to other models, specifically, improvements on the MobileNetV1, deteriorations on the MobileNetV2, ShuffleNetV1 and ShuffleNetV2. Although it may help models find channel features, it pays too much attention to the channel dimension and ignores spatial information, thus being sensitive when faced with different tasks.

Finally, our LAM absorbs the strengths of these two modules to pay attention to both channel and spatial dimensions. In addition, we use a parallel arrangement to process the output feature maps, which operation gives them equal weights. The experiment results imply that our proposed module is powerful, showing the efficacy of a new method that generates a richer descriptor that complements the two attention effectively. The LAM also obeys the rule of lightweight, which means small amounts of parameters and fast forward speed.

4.3 Ablation Study (Q2)

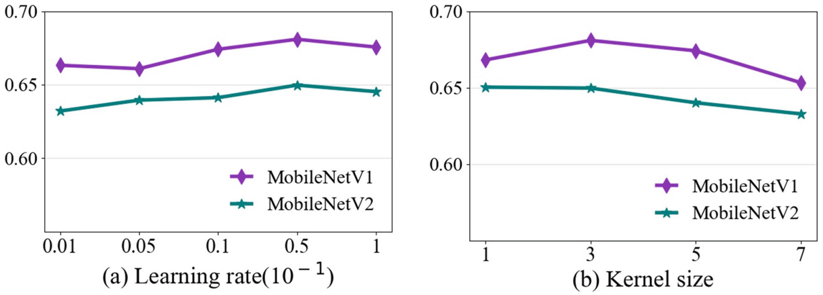
To answer the second question (Q2), we will verify whether the current arrangement are the most beneficial to the effectiveness of the model. In the experiment, we design three variants of the proposed model:

- **LAM(I):** using the LAM before depthwise convolution;
- **LAM(II):** using the LAM after pointwise convolution;
- **LAM(III):** using sequential arrangement.

We measure the accuracy of these three variants. As shown in Table 2, our model performs the best when all latent variables are introduced. The performances of the LAM(I) and LAM(II) are worse than the current module. This shows that the current arrangement, which places the module after the depthwise filters, helps the attention to be applied to the largest representations. The results of LAM(III) show that vanishing gradient problems happen in MobileNetV1, and the sequential arrangement has more parameters and lower speed than the parallel arrangement.

Table 2. Ablation study of the LAM

Architecture	Params	MFlops	Cifar10 Top-1 acc	Cifar100 Top-1 acc
MobileNetV1 + LAM	18.18	360.23	**89.37**	**68.09**
MobileNetV1 + LAM(I)	18.18	360.63	88.64	67.53
MobileNetV1 + LAM(II)	20.22	361.59	89.02	67.92
MobileNetV1 + LAM(III)	18.18	360.23	–	–
MobileNetV2 + LAM	3.17	32.63	**91.08**	**64.97**
MobileNetV2 + LAM(I)	3.17	33.15	89.96	64.10
MobileNetV2 + LAM(II)	3.59	34.82	90.07	64.58
MobileNetV2 + LAM(III)	3.17	32.63	85.31	60.19



(a) Learning rate(10^{-1})

(b) Kernel size

Fig. 4. Parameter sensitivity analysis on Cifar100 dataset.

4.4 Sensitivity Analysis (Q3)

In this subsection, we test the robustness of the model and verify whether the settings of super parameters have an impact on the model. We conduct two groups of experiments, i.e., the learning rates (0.001, 0.005, 0.01, 0.05, 0.1) and the kernel sizes of convolutional layers in the spatial attention part (1, 3, 5, 7). As shown in Fig. 4, our model still keeps a high accuracy between a small range under the change of learning rates and kernel sizes. It implies that the proposed LAM model is not sensitive to these main parameters, and thus has good robustness.

5 Conclusion

In this paper, we propose a novel attention module called LAM for lightweight neural networks, which uses two attention mechanisms but simplifies the components effectively. Specifically, in the spatial attention module, we use element-wise addition and smaller convolutional kernels to avoid the previous vanishing gradient problem. In the channel module, we use the squeeze-and-excitation layers in place of the MLP layers. At last, we take a parallel architecture to integrate the two parts efficiently. The experimental results on the two image classification datasets verify the effectiveness of the proposed attention module for lightweight models. The LAM is ready to be applied to other tasks related to lightweight neural networks, e.g., object tracking in the field of computer vision.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. Computer Science, pp. 1–15 (2014)
2. Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J.: A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans. Neural Netw. Learn. Syst., 1–13 (2021)

3. Machado, G.R., Silva, E., Goldschmidt, R.R.: Adversarial machine learning in image classification: a survey toward the defender's perspective. ACM Comput. Surv. (CSUR) **55**(1), 1–38 (2021)
4. Liu, Y., Sun, P., Wergeles, N., Shang, Y.: A survey and performance evaluation of deep learning methods for small object detection. Exp. Syst. Appl. **172**, 114602 (2021)
5. Yuan, X., Shi, J., Gu, L.: A review of deep learning methods for semantic segmentation of remote sensing imagery. Exp. Syst. Appl. **169**, 114417 (2021)
6. Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Kim, T.K.: Multiple object tracking: a literature review. Artif. Intell. **293**, 103448 (2021)
7. Woo, S., Park, J., Lee, J.-Y., Kweon, I.S.: CBAM: convolutional block attention module. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_1
8. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
9. Li, X., Wang, W., Hu, X., Yang, J.: Selective kernel networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–519 (2019)
10. Fu, J., et al.: Dual attention network for scene segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3146–3154 (2019)
11. Roesch, M.: Snort: lightweight intrusion detection for networks. In: LISA, pp. 229–238 (1999)
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Commun. ACM **60**, 84–90 (2012)
13. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009)
14. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
15. Zhang, X., Zhou, X., Lin, M., Sun, J.: ShuffleNet: an extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6848–6856 (2018)
16. Ma, N., Zhang, X., Zheng, H.-T., Sun, J.: ShuffleNet V2: practical guidelines for efficient CNN architecture design. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 122–138. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_8
17. Howard, A.G., et al.: MobileNets: efficient convolutional neural networks for mobile vision applications. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2017)
18. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNet V2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
19. Howard, A., et al.: Searching for MobileNetV3. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1314–1324 (2019)
20. Tan, M., Le, Q.: EfficientNet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114 (2019)
21. Zoph, B., Le, Q.V.: Neural architecture search with reinforcement learning. In: International Conference on Learning Representations, pp. 1–16 (2016)

22. Marvasti-Zadeh, S.M., Cheng, L., Ghanei-Yakhdan, H., Kasaei, S.: Deep learning for visual tracking: a comprehensive survey. IEEE Trans. Intell. Transp. Syst. **23**, 3943–3968 (2021)
23. Yang, Z., Ma, J., Chen, H., Zhang, Y., Chang, Y.: HiTRANS: a hierarchical transformer network for nested named entity recognition. In: Findings of the Association for Computational Linguistics, EMNLP 2021, pp. 124–132 (2021)
24. Salin, P.A., Bullier, J.: Corticocortical connections in the visual system: structure and function. Physiol. Rev. **75**(1), 107–154 (1995)
25. Wang, F., et al.: Residual attention network for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164 (2017)
26. Qian, K., Wu, C., Yang, Z., Liu, Y., Zhou, Z.: PADS: passive detection of moving targets with dynamic speed using PHY layer information. In: 2014 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS), pp. 1–8 (2014)
27. Miyazawa, A., Fujiyoshi, Y., Unwin, N.: Structure and gating mechanism of the acetylcholine receptor pore. nature **423**(6943), 949–955 (2003)
28. Brigato, L., Barz, B., Iocchi, L., Denzler, J.: Image classification with small datasets: overview and benchmark. IEEE Access **10**, 49233–49250 (2022)
29. Krizhevsky, A., Hinton, G., et al.: Learning Multiple Layers of Features from Tiny Images, pp. 1–60 (2009)
30. Lin, M., Chen, Q., Yan, S.: Network in network. In: International Conference on Learning Representations, pp. 1–10 (2013)
31. Huang, G., Sun, Yu., Liu, Z., Sedra, D., Weinberger, K.Q.: Deep networks with stochastic depth. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 646–661. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_39
32. Shen, L., Lin, Z., Huang, Q.: Relay backpropagation for effective learning of deep convolutional neural networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 467–482. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_29