Journal Pre-proof

AdvMask: A sparse adversarial attack-based data augmentation method for image classification

Suorong Yang, Jinqiao Li, Tianyue Zhang, Jian Zhao, Furao Shen

PII: S0031-3203(23)00545-9
DOI: https://doi.org/10.1016/j.patcog.2023.109847
Reference: PR 109847

To appear in: *Pattern Recognition*

Received date : 4 May 2023
Revised date : 6 July 2023
Accepted date : 26 July 2023

Please cite this article as: S. Yang, J. Li, T. Zhang et al., AdvMask: A sparse adversarial attack-based data augmentation method for image classification, *Pattern Recognition* (2023), doi: https://doi.org/10.1016/j.patcog.2023.109847.

Title Page

AdvMask: A Sparse Adversarial Attack-Based Data Augmentation Method for Image Classification

Suorong Yang[a,b], Jinqiao Li[a,b], Tianyue Zhang[a,b], Jian Zhao[d], Furao Shen [a,c,*]

[a]*State Key Laboratory for Novel Software Technology, Nanjing University, China*
[b]*Department of Computer Science and Technology, Nanjing University, China*
[c]*School of Artificial Intelligence, Nanjing University, China*
[d]*School of Electronic Science and Engineering, Nanjing University, China*

Abstract

Data augmentation has been an essential technique for improving the generalization ability of deep neural networks in image classification tasks. However, intensive changes in appearance and different degrees of occlusion in images are the key factors that severely affect the generalization ability of image classification models. Therefore, in order to enhance the generalization performance and robustness of deep models, data augmentation approaches by providing models with more diverse training data in various scenarios are widely applied. Although many existing data augmentation methods simulate occlusion in the augmented images to enhance the generalization of models, these methods randomly delete some areas in images without considering the semantic information of images. In this work, we propose a novel data augmentation method named AdvMask for image classification based on sparse adversarial attack techniques. AdvMask first identifies the key points that have the greatest influence on the classification results via a proposed end-to-end sparse adversarial attack module. During the data augmentation process, AdvMask efficiently generates diverse augmented data with structured occlusions based on the key points. By doing so, AdvMask can force deep models to seek other relevant content while the most discriminative con-

*Corresponding author. E-mail address: frshen@nju.edu.cn (F. Shen).

Email addresses: sryang@smail.nju.edu.cn (S. Yang), lijq@smail.nju.edu.cn (J. Li), DZ1733025@smail.nju.edu.cn (T. Zhang), jianzhao@nju.edu.cn (J. Zhao), frshen@nju.edu.cn (F. Shen)

tent is hidden. Extensive experimental results on various benchmark datasets and deep models demonstrate that our proposed method can effectively improve the generalization performance of deep models and significantly outperforms previous data augmentation methods. Code for reproducing our results is available at https://github.com/Jackbrocp/AdvMask.

Keywords: Data augmentation, image classification, sparse adversarial attack, generalization

Highlights

AdvMask: A Sparse Adversarial Attack-Based Data Augmentation Method for Image Classification

Suorong Yang, Jinqiao Li, Tianyue Zhang, Jian Zhao,
Furao Shen

- We propose AdvMask, a novel data augmentation method that utilizes sparse adversarial attack techniques to identify critical regions in images for downstream data augmentation, surpassing traditional salient portions.
- We develop an end-to-end sparse adversarial attack module that automates the selection of pixels for data augmentation, eliminating the need for hand-crafted rules.
- Experimental results have demonstrated that AdvMask outperforms information deletion-based augmentation approaches on various datasets and deep models. This highlights the effectiveness and superiority of our proposed method.
- AdvMask can be seamlessly combined with other data augmentation methods, such as automated augmentation, to achieve further improvements in performance. This flexibility enhances the versatility and potential applications of our method.

AdvMask: A Sparse Adversarial Attack-Based Data Augmentation Method for Image Classification

Suorong Yang[a,b], Jinqiao Li[a,b], Tianyue Zhang[a,b], Jian Zhao[d], Furao Shen [a,c,*]

[a]*State Key Laboratory for Novel Software Technology, Nanjing University, China*
[b]*Department of Computer Science and Technology, Nanjing University, China*
[c]*School of Artificial Intelligence, Nanjing University, China*
[d]*School of Electronic Science and Engineering, Nanjing University, China*

Abstract

Data augmentation has been an essential technique for improving the generalization ability of deep neural networks in image classification tasks. However, intensive changes in appearance and different degrees of occlusion in images are the key factors that severely affect the generalization ability of image classification models. Therefore, in order to enhance the generalization performance and robustness of deep models, data augmentation approaches by providing models with more diverse training data in various scenarios are widely applied. Although many existing data augmentation methods simulate occlusion in the augmented images to enhance the generalization of models, these methods randomly delete some areas in images without considering the semantic information of images. In this work, we propose a novel data augmentation method named AdvMask for image classification based on sparse adversarial attack techniques. AdvMask first identifies the key points that have the greatest influence on the classification results via a proposed end-to-end sparse adversarial attack module. During the data augmentation process, AdvMask efficiently generates diverse augmented data with structured occlusions based on the key points. By doing so, AdvMask can force deep models to seek other relevant content while the most discriminative con-

*Corresponding author. E-mail address: frshen@nju.edu.cn (F. Shen).

Email addresses: sryang@smail.nju.edu.cn (S. Yang), lijq@smail.nju.edu.cn (J. Li), DZ1733025@smail.nju.edu.cn (T. Zhang), jianzhao@nju.edu.cn (J. Zhao), frshen@nju.edu.cn (F. Shen)

tent is hidden. Extensive experimental results on various benchmark datasets and deep models demonstrate that our proposed method can effectively improve the generalization performance of deep models and significantly outperforms previous data augmentation methods. Code for reproducing our results is available at https://github.com/Jackbrocp/AdvMask.

Keywords: Data augmentation, image classification, sparse adversarial attack, generalization

1. Introduction

Convolutional neural networks (CNNs) have made tremendous progress in the field of computer vision, such as image classification [1]. Despite the tremendous progress, training large CNN models to achieve satisfactory generalization performance typically requires a considerable amount of labeled data, which is often unavailable. Meanwhile, the generalization ability of CNNs is based on a fundamental assumption in machine learning algorithms that the training and test data share the same distribution, which is not always the case. In practical application, because training data in most image datasets typically exhibit limited variance in appearance and occlusion, the generalization ability of CNNs can be severely impacted in these scenarios [2, 3]. For instance, if a classification model is trained with images that are all clearly visible, due to the limited generalization performance of the model, it may fail to recognize partly occluded objects in practice. Considering the extreme difficulty in collecting and labeling data in diverse scenes, in recent years, data augmentation approaches based on information deletion have been a research focus of data augmentation, aiming to enhance the generalization performance and robustness of CNN models. Data augmentation techniques are applied to training data to generate more diverse augmented data during the training phase and thus enhance the models' generalization performance. These information deletion-based approaches typically generate additional data by deleting some sub-regions of the original images to simulate the situation in the real world where objects may be partially covered, or the whole structure is incomplete, such as Cutout [4], Random Erasing [2], Hide and Seek (HaS) [5], and GridMask [3]. It is known that these approaches can reduce the sensitivity of models, increase the perception field, and enhance the generalization ability of deep models [3]. However, instead of customizing the occlusion mask for each image based on its structural

characteristics, these approaches randomly block some areas in the images. The completely random regional deletion is uninterpretable and cannot be customized according to the images' characteristics. Moreover, the random deletion may inevitably introduce noise and ambiguity into the training process, which can result in augmented images with false or fuzzy labels in some situations [6, 7]. In contrast, tailoring the removal of sub-regions based on the images' structural and semantic information can mitigate these concerns, thus bolstering model robustness.

To address these issues, this paper proposes AdvMask, an innovative sparse adversarial attack-based data augmentation approach. It aims to generate augmented samples by simulating situations where the critical information used for classification is partially lost and forces models to identify other relevant information in the images to improve the generalization capability and robustness of CNN models. Inspired partly by two-stage automated augmentation approaches such as AutoAugment [8], and Fast-AutoAugment [9], AdvMask is primarily composed of two modules: an end-to-end sparse adversarial attack module and a data augmentation module. In the first stage, the sparse adversarial attack module automatically identifies the critical pixels that have the most significant impact on the classification decisions in an offline manner. In the second stage, the data augmentation module generates augmented data by deleting some sub-regions that contain the adversary key points. Through the integration of the automatic identification of critical pixels and the subsequent customization of regional deletion, AdvMask presents a more targeted and refined approach compared to alternative methods. At the same time, these adversary key points are closely related to the classification results because applying invisible perturbations to them could cause misclassification. Therefore, instead of randomly deleting some regions in images, the rationale of AdvMask lies in the fact that it can mimic real-scene occlusions and force models to search for other relevant information while the most discriminative content is hidden, which enhances the models' robustness. Meanwhile, using the customized occlusion on adversary key points is also effective in helping the model jump out of local optimum [10], which ultimately reduces over-fitting risks and enhances the model's generalization and robustness. In order to validate the efficacy of our proposed approach, we perform extensive evaluations on CIFAR-10, CIFAR-100 [11] and Tiny-ImageNet [12] datasets for coarse-grained classification, and Oxford Flower Dataset [13] for fine-grained classification and effect visualization. Utilizing various neural architectures, we demonstrate that our approach outperforms

other information deletion-based data augmentation approaches and yields significant improvements when combined with other augmentations, including Mixup [14], CutMix [15], AutoAugment [8], Fast-AutoAugment [9] and TrivialAugment [16]. Moreover, through effect visualization, we demonstrate that our approach enables deep models to focus on the most discriminative and salient areas of images. Finally, the training time costs of AdvMask are still competitive with other augmentation approaches, indicating its practical value.

We highlight our contributions as follows:

- We propose AdvMask, a novel data augmentation method that utilizes sparse adversarial attack methods for data augmentation. To the best of our knowledge, we are the first to utilize sparse adversarial attack techniques to locate the critical regions for data augmentation. We also prove with experimental results that the proposed method effectively identifies critical regions in images that are more significant for image classification than traditional salient portions.

- We propose an end-to-end sparse adversarial attack module, which uses the adversarial attack technique to guide the automated selection of pixels for downstream data augmentation without designing handcrafted rules. Compared with other sparse adversarial attack methods, it is competitive in attack performance.

- Experimental results show that AdvMask significantly outperforms other information deletion-based augmentation approaches on various datasets and deep models. At the same time, our method can also be combined with other data augmentation methods (e.g., automated augmentation) to yield further improvements.

2. Related Work

2.1. Data Augmentation

Recently, many data augmentation methods have been proposed to improve the generalization of deep models.

Image mixing-based data augmentation approaches, such as Mixup [14] and CutMix [15], utilize multi-image information to synthesize new training samples during training. By modifying the input images and labels, these methods can fuse information from multiple images and achieve good results.

KeepAugment [6] uses the saliency map to detect important regions on the original images and preserve these informative regions during augmentation.

Two-stage automated augmentations like AutoAugment [8] and Fast-AutoAugment [9] leverage reinforcement learning to find the optimal combination of data augmentation operations for each dataset. TrivialAugment [16] employs the same augmentation space obtained by these automated augmentations and applies a single augmentation to each image during training. When employing augmentations, there is no requirement for reinforcement learning's search for augmentation policies. Our proposed method also employs a two-stage mechanism, yet it can be used both independently and in conjunction with other image mixing and two-stage automated augmentation methods for improved performance.

Information deletion-based methods simulate occlusion by selectively deleting some information from an input image. Random Erasing (RE) [2] reduces the risk of over-fitting and makes the model robust to occlusion by randomly selecting a rectangular region in an image and erasing its pixels with random values. Similarly, Hide-and-Seek (HaS) [5] randomly hides patches in a training image, improving object localization accuracy and the generalization ability of deep models. Cutout [4] randomly masks out square regions of input images, which serves as a regularization technique and enhances the overall performance of deep models. GridMask [3] adopts structured dropping regions in input images, emphasizing the balance between information deletion and reservation and significantly improving the performance of deep models. However, these methods randomly select regions for deletion, which is uninterpretable and may introduce noise and ambiguity into the augmented data. Different from the previous works, AdvMask first identifies the classification-critical regions of images and selectively drops some structured sub-regions with critical points during augmentation, which is more reasonable than random deletion.

Similar to our work, [10] proposes a data augmentation method for person re-identification tasks, which generates occluded samples adversarial to the existing models. Meanwhile, the work [17] proposes an inverse attack scheme for data augmentation in image segmentation by modifying images while keeping misclassification rates as low as possible. However, these approaches identify the discriminative regions of training images with the help of network visualization techniques (e.g., classification activation maps), or directly use adversarial examples as augmented data. In contrast, we devise an end-to-end sparse adversarial attack module to guide the automated selec-

tion of pixels without hand-crafted rules or any other network visualization techniques and generate augmented images based on the clean data during training, indicating enhanced flexibility and broader application scenarios.

2.2. Adversarial Attack

Adversarial attack is a technique that carefully crafts images to deceive machine learning models by introducing small perturbations to the original clean images. These perturbations usually can not be detected by human beings. However, determining which pixels to perturb is a challenging task. Existing methods adopt a two-stage pipeline to perturb the most critical pixels until the attack is successful. Firstly, these methods artificially define a measure of pixel importance, such as gradients. Then, based on the importance of each pixel, an iterative strategy is applied, and the most critical pixels are selected to be attacked until the attack succeeds. For example, based on the saliency map, JSMA [18] uses a heuristic strategy to select the pixels to be perturbed in each iteration iteratively. C&W-ℓ_0 [19] first uses the attack under the constraint of ℓ_2-norm and then fixes several least important pixels according to the perturbation magnitudes and gradients. PGD-$\ell_0 + \ell_\infty$ [20] projects the perturbations generated by PGD [21] onto the ℓ_0-ball to achieve the ℓ_0 version of PGD. The specific projection method is to fix some pixels that do not need to be perturbed according to the perturbation magnitudes and projection loss. SparseFool [22] converts the problem into a ℓ_1-norm constrained problem and selects some pixels to perturb in each iteration according to the geometric relationship. Based on the gradient and the distortion map, GreedyFool [23] selects some pixels to be added to the modifiable pixel set in each iteration and then uses the greedy method to drop as many less critical pixels as possible to obtain better sparsity. However, these methods rely on hand-crafted rules to evaluate the importance of pixels and use greedy methods to remove as many unimportant pixels as possible to achieve better sparsity, which may not well serve the downstream data augmentation process because suitable rules for evaluating the importance of pixels should be carefully designed. Instead, our method eliminates the need for a predefined metric for pixel importance and employs an end-to-end sparse adversarial attack module to select the most classification-sensitive pixels, thereby facilitating the data augmentation process. Thus, our approach has broader application scenarios compared to previous methods that rely on hand-crafted rules or heuristic strategies. The technique of factorizing the perturbation into a product of magnitude and the binary mask has

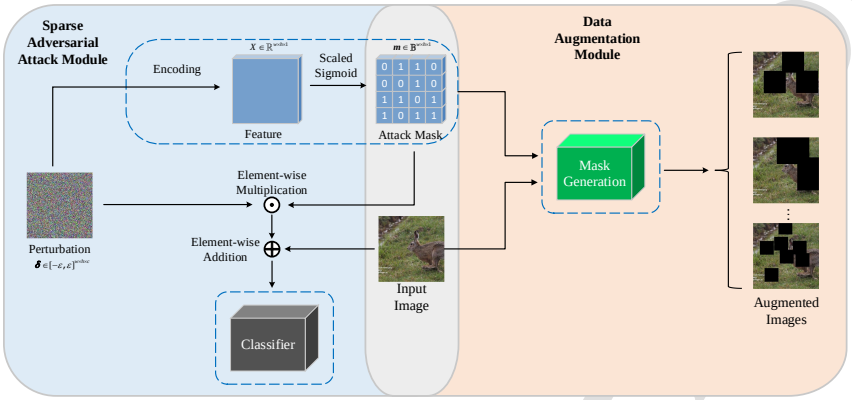


Figure 1: The AdvMask framework is composed of two key modules: the sparse adversarial attack module and the data augmentation module. In the first module, a perturbation is encoded using a trainable neural network to generate a feature map, which is then used to produce an approximately binary attack mask through the scaled sigmoid function. This attack mask is subsequently used to inform the generation of augmentation masks for the augmented images. It should be noted that the resulting shapes of the generated augmentation masks exhibit a high degree of variability.

been utilized by SAPF [24], which jointly optimizes the perturbation components using ℓ_p-box ADMM [25], a popular optimization algorithm. Our proposed method shares similarities with SAPF, in that we also employ the binary mask to generate adversarial perturbations. However, unlike SAPF, our method employs a trainable neural network to generate the binary mask, which can be conveniently applied in practice.

3. AdvMask

As depicted in Figure 1, the AdvMask framework comprises two key modules: the sparse adversarial attack and the data augmentation module. Before delving into the details of these modules, we first introduce two types of masks employed in AdvMask: the attack mask and the augmentation mask. The attack mask is generated by the sparse adversarial attack module and provides information about the spatial distribution of the adversarial attack points. On the other hand, the augmentation mask is obtained from the data augmentation module and is guided by the attack mask to locate the occluded regions of the augmented data. In the sparse adversarial attack

module, AdvMask takes an image as input and applies a random adversarial perturbation. The perturbation is encoded by a trainable neural network to automatically generate a binary attack mask of the same size. The binary attack mask is then utilized to obtain the Points of Interest (POIs) for the data augmentation module. In the data augmentation module, AdvMask performs data augmentation based on the POIs derived from the attack mask.

3.1. Sparse Adversarial Attack

Sparse adversarial attack aims to find the most sensitive pixels in images and perturb these pixels to induce misclassification from the CNN models [20]. Let $f : [0,1]^{w\times h\times c} \rightarrow \mathbb{R}^K$ be the classification model, where w, h and c denote the width, height, and the number of channels of the images, respectively. K represents the total number of classes. We formulate the sparse adversarial attack as follows:

$$\begin{aligned} \min_{\boldsymbol{\delta}} \quad & \|\boldsymbol{\delta}\|_0 \\ \text{s.t.} \quad & \boldsymbol{\delta} \in [-\epsilon, \epsilon]^{w\times h\times c} \\ & \arg\max_{i=1,\ldots,K} f_i(\boldsymbol{x}+\boldsymbol{\delta}) \neq y_{true} \end{aligned} \tag{1}$$

where $\boldsymbol{\delta} \in \mathbb{R}^{w\times h\times c}$ is the adversarial perturbation, $\|\cdot\|_0$ denotes the l_0-norm that is the number of non-zero elements, $f_i(\cdot)$ denotes the probability that the model classifies the input $\boldsymbol{x}+\boldsymbol{\delta}$ as class i, and ϵ denotes the maximum perturbation magnitude allowed, which is the ℓ_∞-norm of the perturbation. $\boldsymbol{x}$ and y_{true} denote the input image and its actual label, respectively. Therefore, the optimization finds $\boldsymbol{\delta}$, which is added into the original image $\boldsymbol{x}$, so that models classify $\boldsymbol{x}+\boldsymbol{\delta}$ as any class except the actual one.

In practice, to obtain sparse key pixels, the magnitude of the adversarial perturbations is restricted by limiting ϵ. The sparsity of adversarial attack points is advantageous to the data augmentation module because these points are more likely to cover the foreground and background. Our work utilizes l_0-norm as a distance function to learn sparse adversarial attacks. However, because the l_0-norm is not differentiable, the difficulty in optimizing the problem makes the sparse adversarial attack an NP-hard problem [23]. Inspired by the research of neural network pruning [26, 27], pruning neural network can be regarded as an optimization problem of l_0-norm that indicates the sparsity of convolution kernels. Therefore, a branch is added that includes encoding and binarization in the usual attack process, which outputs a mask to select pixels automatically, avoiding the use of artificially

defined importance indicators and greedy strategies. The branch takes the adversarial perturbation as the input and generates an approximate binary mask of the same size. The adversarial perturbation and the encoder are jointly optimized according to the sparsity and attack effect. By gradually forcing the scaled sigmoid function to output binary values, the perturbation of some pixels will finally be 0, thereby ensuring sparsity.

Encoding. Let $\mathcal{H} : \mathbb{R}^{w\times h\times c} \to \mathbb{R}^{w\times h\times 1}$ denote the mapping of the neural network. $\mathcal{H}(\boldsymbol{\delta}) \in \mathbb{R}^{w\times h\times 1}$ represents the tensor obtained by encoding the perturbation $\boldsymbol{\delta}$ with $\mathcal{H}$. In cases where the image size is small, a fully-connected layer is utilized as the encoder with weight parameters $\mathcal{W} \in \mathbb{R}^{(w\cdot h\cdot c)\times(w\cdot h)}$. However, large image sizes lead to a high number of parameters to be trained, which poses challenges for optimization. To address the issue, we propose the use of U-net [28], an image segmentation network, as the encoder. U-net ensures that the input and output dimensions are the same and thus provides a feasible solution for encoding perturbations in larger images.

Binarization. The elements of the encoded tensor are real numbers, while the attack mask consists of binary elements where 1 denotes the key points, and 0 denotes others. To maintain differentiability and continuity, we employ the scaled sigmoid function to generate an approximate binary mask:

$$\boldsymbol{m} = \text{sigmoid}(\alpha \cdot \mathcal{H}(\boldsymbol{\delta})) \tag{2}$$

where $\boldsymbol{m} \in [0, 1]^{w\times h\times 1}$ denotes the approximate binary mask, and α is the scaling factor that controls the degree of binarization. During training, α is gradually increased to facilitate the convergence of mask elements to binary values. If α is set too large initially, the selection of pixels is determined before the training, degenerating our method into randomly selecting pixels. As such, we mitigate this issue by gradually increasing α from α_{start} to α_{end} during training to ensure that the elements of the mask can converge to binary values and prevent the method from degenerating into random pixel selection.

Loss Function. Our objective is to identify the most sensitive and sparse pixels in images. Therefore, the design of the loss function is to effectively deceive the target model with the pruned perturbations while keeping the key pixels as sparse as possible. To this end, we utilize adversarial attacks, such as PGD [21] and MI-FGSM [29], to calculate the loss of cross entropy

with a successful attack, denoted as $\mathcal{L}_{init}$. We formulate our loss function as follows:

$$\begin{aligned} \mathcal{L} &= \max(\mathcal{L}_{\text{classify}}, \eta \cdot \mathcal{L}_{\text{classify}}) + \lambda \frac{\|\boldsymbol{m}\|_1}{N}, \\ \mathcal{L}_{\text{classify}} &= -\frac{\mathcal{L}_{\boldsymbol{CE}}(f(\boldsymbol{x} + \boldsymbol{\delta} \odot \boldsymbol{m}), y_{true})}{\mathcal{L}_{init}} + 1 \end{aligned} \tag{3}$$

where $\odot$ denotes the element-wise multiplication, $\mathcal{L}_{\text{classify}}$ is the adversarial classification loss, usually the cross-entropy loss. Since the elements of $\boldsymbol{m}$ are approximately binary, the second term can represent the mask's sparsity, where $N = w \times h \times 1$ represents the size of the mask. λ is a dynamic parameter to balance these two terms and is calculated by the following formula:

$$\lambda = C + \frac{\gamma}{N} \sum_{i=1}^{N} \mathbb{I}\left(m_i > 0.5\right) \tag{4}$$

where $C > 0$, is a hyperparameter that denotes the minimum value of λ, m_i is the i-th element of the mask, and $\gamma > 0$ is a hyperparameter. In cases where the current mask is not sparse enough, our approach prioritizes enhancing the sparsity of the mask. On the other hand, if the current mask is already sufficiently sparse, our approach focuses on improving the effectiveness of the adversarial attack.

Update Perturbation and Encoder. Since the attack mask is to identify the most classification-sensitive points in an image, the sparsity of the attack mask need not be carefully considered. Instead of updating $\boldsymbol{\delta}$ with greedy strategies, adversarial attack methods such as PGD and MI-FGSM can be used to quickly update $\boldsymbol{\delta}$, while constraining its ℓ_0-norm and ℓ_∞-norm. Specifically, the update formula of $\boldsymbol{\delta}$ is as follows:

$$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{\delta}} \mathcal{L}}{\|\nabla_{\boldsymbol{\delta}} \mathcal{L}\|_1} \tag{5}$$

$$\boldsymbol{\delta}_{t+1} = \text{Clip}_{\boldsymbol{\epsilon}}\left\{\boldsymbol{\delta}_t - \beta \cdot \text{sign}\left(\boldsymbol{g}_{t+1}\right)\right\} \tag{6}$$

where $\mathcal{L}$ is the loss defined by Equation(3), μ denotes the momentum decay factor, β represents the update step, and $\text{Clip}_{\boldsymbol{\epsilon}}\{\cdot\}$ is used to project adversarial perturbation into the ℓ_∞-ball of radius ϵ. The perturbation is trained concurrently with the encoder, and the attack mask is generated upon the completion of training.

Algorithm 1 Sparse Adversarial Attack module

Input: an image $\boldsymbol{x}$, target class label y^*, maximum number of iterations T, classification model f, maximum perturbation magnitude ϵ, hyperparameter γ and C, scaling factor α_{start} and α_{end}, update step β

Output: adversarial perturbation $\boldsymbol{\delta}$

$\boldsymbol{\delta}_0 \leftarrow \mathbf{0}$
$\boldsymbol{g}_0 \leftarrow \mathbf{0}$
$\alpha_0 \leftarrow \alpha_{start}$
Randomly initialize the encoder $\mathcal{H}_0$
for $t = 0{:}T-1$ **do**
  $\boldsymbol{m}_t \leftarrow \text{sigmoid}(\alpha_t \cdot \mathcal{H}_t(\boldsymbol{\delta}_t))$ according to Eq. (2)
  Calculate the dynamic parameter λ_t based on $\boldsymbol{m}_t$ and Eq. (4)
  Calculate the loss $\mathcal{L}$ according to $\boldsymbol{m}_t$, $\boldsymbol{\delta}_t$, λ_t and Eq. (3)
  Update $\boldsymbol{g}_{t+1}$ according to Eq. (5)
  Update $\boldsymbol{\delta}_{t+1}$ according to Eq. (6)
  Update encoder $\mathcal{H}_{t+1}$ based on the loss $\mathcal{L}$ and SGD with momentum
  Update the scaling factor α_{t+1} using the algorithm in [26]
end for
$\boldsymbol{m}_T \leftarrow \text{sigmoid}(\alpha_T \cdot \mathcal{H}_T(\boldsymbol{\delta}_T))$
$\boldsymbol{\delta} \leftarrow \boldsymbol{\delta_T} \odot \boldsymbol{m}_T$
return $\boldsymbol{\delta}$

Following the commonly adopted two-stage mechanism of automated augmentation approaches [8, 9], the sparse adversarial attack module is carefully devised in an offline manner. The module is dependent solely on the dataset and requires only one training run to obtain the POIs for each dataset. No inference or training is necessary from the sparse adversarial attack module on the downstream data augmentation module. Although the sparse adversarial attack module is not directly integrated into the classification model training, it is designed to optimize efficiency while maintaining satisfactory performance. Several examples of attack masks are presented in Figure 2. As shown in Figure 2, we can observe that the critical points nearly cover the main object and some areas in the background, revealing that most of the information used for classification is concentrated on the main object of the image with some information in the background. Consequently, AdvMask can provide more diverse augmented data by hiding discriminative information in both the foreground and background. Finally, to ensure reproducibil-

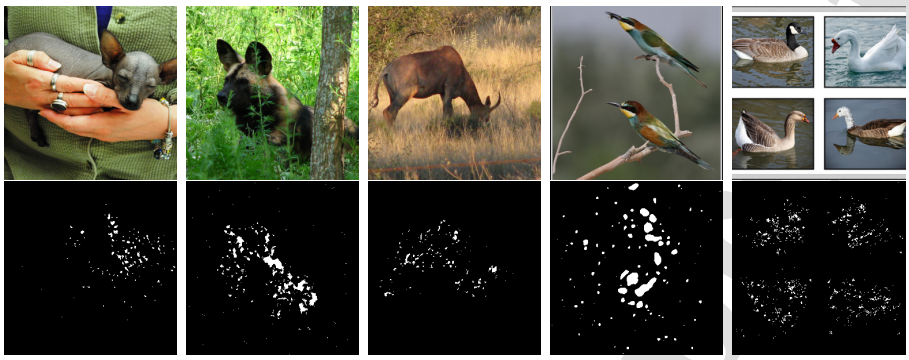


Figure 2: Illustration of attack mask generated by our sparse adversarial attack module. The first row: original images from the ImageNet dataset [30]. The second row: attack masks for images in the first row.

ity and facilitate future research endeavors, we will release the attack masks employed on various benchmark datasets to the public in the near future.

3.2. Data Augmentation

To strike a balance between the removal and preservation of crucial information in an image, the data augmentation module is appropriately devised. Excessive deletion of regions may lead to loss of contextual information and the complete removal of an object [3], while too much preservation increases overfitting risks. Meanwhile, considering that removing structural regions is beneficial to model training [4, 5, 3], our proposed data augmentation module utilizes structural dropping regions to customize the augmentation masks for each image. Our setting can be expressed as $\mathbf{x}' = \mathbf{x} \odot M$, where $\mathbf{x} \in \mathbb{R}^{w \times h \times c}$ denotes the input image, $M \in [0,1]^{w \times h}$ is the binary augmentation mask indicating the pixels to be removed, and $\odot$ is the element-wise multiplication. The augmentation mask M is determined by three parameters: l, p, and o.

Square Length l. In our proposed approach, we introduce a square region, denoted as s, that is centered on each perturbation point in the augmentation mask. The length of s is controlled by the parameter l, which is randomly sampled from the interval $[l_{min}, l_{max}]$.

Mask Ratio p. As some recent works [31, 3] have shown that dropping a small region is useless for the convolution operation, parameter mask ratio p

is used to control the total proportion of masked pixels. Considering the balance between information deletion and reservation, the scale range of masked regions p should be limited. Once the parameter l is determined, the area of total masked regions can be calculated by $\text{sum}(M) = |S| * l^2$, where S is the set of squares in the augmentation mask and $|S|$ denotes the number of squares. Then, we get $p = \frac{\text{sum}(M)}{h \times w}$, where h and w are the height and width of the images, respectively. We ensure that the number of masked pixels is neither too high nor too low by limiting the scale range of p to $[p_{min}, p_{max}]$.

Overlapping Ratio *o.* Because each square s has different lengths, the squares centered on these points may easily overlap when the perturbation points are densely positioned. This situation could deteriorate sharply when the lengths of the squares are large, and the perturbation points are dense, which causes the deleted areas to overlap and finally concentrate on subareas continuously. In the worst case, no matter what l is, the whole object of interest will be deleted. To mitigate this issue, we propose the control of overlapping ratio o by randomly sampling from the range $[0, o_{max}]$. By restricting the overlapping area of every pair of squares to be no larger than $\lceil l^2 \times o \rceil$, the structural characteristics of the mask are preserved.

During the training process, the randomness introduced by these parameters increases the diversity of the augmentation masks, thereby enhancing the diversity of augmented data. Further details regarding the specific parameter settings of the augmentation masks and ablation study on the effect of parameters can be found in Section 5.4.

4. Experiment

In this section, we test the performance of AdvMask on various benchmark datasets, including CIFAR-10, CIFAR-100, Tiny-ImageNet, and Oxford Flower Dataset.

4.1. Implementation Details

In the sparse adversarial attack module, a binary attack mask is obtained by gradually increasing the scaling factor α from α_{min} to α_{max} during the optimization process, where α_{max} is set to 100 on CIFAR-10 and CIFAR-100, and 5 on Tiny-ImageNet and Oxford Flower Dataset. The parameter α_{min} is fixed at 0.1 as its variation has little effect on the results. In the data augmentation module, the datasets are normalized using per-channel mean and

Dataset	Model	Baseline	Cutout [4]	HaS [5]	GM [3]	RE [2]	Ours
CIFAR-10	ResNet-18	95.28 *	96.01 *	96.10 *	96.38	95.69 *	**96.44**
	ResNet-44	94.10	94.78	94.97	95.02	94.87 *	**95.49**
	ResNet-50	95.66	95.81	95.60	96.15	95.82	**96.69**
	WRN-28-10	95.52	96.92	96.94	**97.23**	96.92	97.02
	Shake-26-32	94.90	96.96 *	96.89 *	96.91	96.46 *	**97.03**
CIFAR-100	ResNet-18	77.54 *	78.04 *	78.19	75.23	75.97 *	**78.43**
	ResNet-44	74.80	74.84	75.82	76.07	75.71 *	**76.44**
	ResNet-50	77.41	78.62	78.76	78.38	77.79	**78.99**
	WRN-28-10	78.96	79.84	80.22	80.40	80.57	**80.70**
	Shake-26-32	76.65	77.37	76.89	77.28	77.30	**79.96**

Table 1: Image classification accuracy of information deletion-based data augmentation methods on CIFAR-10 and CIFAR-100 are summarized in this table. * means results reported in the original paper. **GM**: GridMask. **RE**: random erasing.

standard deviation before applying the AdvMask algorithm. Inspired by the easy-to-hard learning strategy [32, 33], we devise an incremental generative strategy. Specifically, instead of applying AdvMask to every image in each epoch, the number of augmented samples gradually increases during training until it reaches a constant upper bound, which will make the network more robust to occlusion by learning more and more occluded samples. In practice, we set the upper bound to 80% of the total sample size. When training classification models using various data augmentation approaches, we closely follow the experimental settings described in the paper [4, 3]. Specifically, all images are preprocessed by dividing each pixel value by 255 and normalizing by the dataset statistics. We train 1800 epochs with cosine learning rate decay for Shake-Shake [34] using SGD with Nesterov Momentum and a learning rate of 0.01, a batch size of 256, $1e-3$ weight decay and cosine learning rate decay. We train 300 epochs for all other networks using SGD with Nesterov Momentum and a learning rate of 0.1, a batch size of 128, a $5e-4$ weight decay, and cosine learning rate decay. For a fair comparison, all methods are implemented with the same training configurations.

4.2. Results on CIFAR-10 and CIFAR-100

In this section, we present an evaluation of AdvMask's performance on CIFAR-10 and CIFAR-100 datasets, using WideResNet-28-10 (WRN-28-10) [35], ShakeShake-26-32 (Shake-26-32)[34], and ResNet [36] architectures of varying sizes, including ResNet-18, ResNet-44, and ResNet-50. We compare AdvM-

sak with other data augmentation approaches based on information deletion, including Cutout, HaS, GridMask, and Random Erasing, where GridMask is a previous SOTA data augmentation method based on information deletion. These augmentation methods are applied after the baseline augmentation, which involves padding the image to 36×36, randomly cropping it into 32×32, and horizontally flipping it with a probability of 50%. We find that the parameters of AdvMask are related to the complexity of the dataset, with larger masked regions preferred for more complex datasets. Therefore, for CIFAR-10, we train using the range of square length l of $[2, 15]$, the range of mask ratio p of $[0.2, 0.4]$, and overlapping ratio o of 0.1. For CIFAR-100, the range of square length l of $[5, 20]$, the range of mask ratio p of $[0.06, 0.5]$, and overlapping ratio o of 0.2.

Table 1 summarizes the test accuracy of various information deletion-based data augmentation approaches as well as the baseline model on these two datasets using a number of neural architectures. We can observe that AdvMask outperforms other data augmentation approaches as well as the baseline model consistently. For example, on CIFAR-10, AdvMask achieves improvements on the baseline model's performance of ResNet-18, ResNet-44, ResNet-50, WRN-28-10 and ShakeShake-26-32 by 1.16%, 1.39%, 1.03%, 1.50%, and 2.12%, respectively. Similarly, on CIFAR-100, we have improved the classification accuracy of ResNet44, WideResNet-28-10, and ShakeShake-26-32 by 1.64%, 1.74%, and 3.31%, respectively. AdvMask also performs better than the widely-used Cutout augmentation on ResNet-18, ResNet-44, ResNet-50, and WRN-28-10, with the test accuracy improvements of 0.43%, 0.71%, 0.88%, and 0.1%, respectively. The superiority of AdvMask can be attributed to its ability to hide classification-critical information, forcing models to learn other discriminative information, which is different from Cutout's random area removal approach. Therefore, AdvMask can achieve substantial performance compared to these information deletion-based approaches.

Additional Comparisons on CIFAR-10 and CIFAR-100 We also compare AdvMask with other data augmentation methods, including Mixup, CutMix, AutoAugment, Fast AutoAugment, KeepAugment, and TrivialAugment. To ensure a fair comparison of different approaches, we adopted the parameter settings suggested by the original works and train all models using a consistent setup [4, 3]. Inspired by [3, 37, 38], we conduct experiments on CIFAR-10 and CIFAR-100 using various neural architectures by applying our method after the operations of these methods.

Table 2 and Table 3 summarize the test accuracy for various approaches

Method	ResNet-18	ResNet-44	WRN-28-10
Mixup [14]	96.10	94.85	96.92
AdvMask-Mixup	96.79 +0.69	95.08 +0.23	97.21 +0.29
CutMix [15]	96.64	93.96	96.93
AdvMask-CutMix	**96.81** +0.17	95.08 +1.12	97.14 +0.21
AutoAugment [8]	96.07 *	95.01	97.01 *
AdvMask-AA	96.69 +0.62	**96.30** +1.29	**97.59** +0.58
Fast-AutoAugment [9]	95.99	93.80	96.81
AdvMask-FAA	96.52 +0.53	95.42 +1.62	97.52 +0.71
TrivialAugment [16]	96.28	95.00	97.18
AdvMask-TA	96.58 +0.30	95.30 +0.30	97.55 +0.37
KeepCutout [6]	96.10 *	95.35	97.30 *
KeepAutoAugment [6]	96.37	95.46	97.30 *

Table 2: Test accuracy(%) on CIFAR-10 using various models architectures. * means results reported in other papers. **AA**: AutoAugment. **FAA**: Fast-AutoAugment. **TA**: TrivialAugment.

on CIFAR-10 and CIFAR-100, respectively. As shown in Table 2, AdvMask consistently achieves improvements in test accuracy. For instance, when integrated with AutoAugment, AdvMask leads to a 1.29% and 0.58% increase in accuracy for ResNet-44 and WRN-28-10, respectively, compared to AutoAugment alone. This is because deletion-based data augmentation can enhance the generalization capability of deep models by increasing the perception field. AdvMask can yield further improvements when combined with other non-deletion-based augmentation methods. Notably, the test accuracy of AdvMask is still superior to that of KeepCutout, which preserves salient information from being removed after the Cutout operation. This superiority is attributable to the fine-grained control of AdvMask's three parameters, which facilitate a balance between the removal and preservation of areas. In practice, the critical areas are seldom completely removed after augmentation, with critical information typically being partially removed. Therefore, AdvMask can potentially preserve some critical areas from being influenced. Table 3 presents the results on CIFAR-100. We find that AdvMask can also improve test accuracy on CIFAR-100 in conjunction with other approaches. For instance, when combined with AutoAugment, AdvMask can improve the test accuracy of AutoAugment by 1.05%, 1.28%, and 0.8% on ResNet-44, ResNet-50, and WRN-28-10, respectively. When combined with Mixup and

Method	ResNet-44	ResNet-50	WRN-28-10
Mixup [14]	73.25	82.46	83.00
AdvMask-Mixup	74.02 +0.77	**83.93** +1.47	**83.46** +0.46
CutMix [15]	73.97	81.34	82.67
AdvMask-CutMix	74.08 +0.11	81.60 +0.26	81.31 −1.36
AutoAugment [8]	76.36	81.34	82.21
AdvMask-AA	77.41 +1.05	82.62 +1.28	83.01 +0.80
Fast-AutoAugment [9]	76.04	79.08	79.95
AdvMask-FAA	76.27 +0.23	81.18 +2.10	81.45 +0.50
TrivialAugment [16]	76.80	81.34	82.75
AdvMask-TA	**78.96** +2.16	81.29 −0.05	83.19 +0.44
KeepCutout [6]	76.29	78.90	77.52
KeepAutoAugment [6]	77.62	81.25	79.81

Table 3: Test accuracy(%) on CIFAR-100 using various models architectures. **AA**: AutoAugment. **FAA**:Fast-AutoAugment. **TA**: TrivialAugment.

TrivialAugment, AdvMask can achieve the best accuracy on various neural architectures. Lastly, these results demonstrate that AdvMask is a flexible and effective method that can be integrated with other data augmentation techniques to further improve their performance.

4.3. Results on Tiny ImageNet

On the Tiny-ImageNet dataset, we perform experiments to evaluate the effectiveness and stability of various data augmentation approaches based on the accuracy of ResNet-18, ResNet-50, and Wide-ResNet-50-2 [35] models. We resize the images to 64×64, initialize the models with ImageNet pre-trained weight, and fine-tune them on Tiny-ImageNet for some epochs. In order to reflect the improvements of the classification accuracy brought by different augmentation approaches over the baseline, in Figure 3, we present the relative accuracy improvements of various data augmentation methods, as well as the error interval of each method. All experiments are conducted across five independent random trials. The baseline accuracies for ResNet-18, ResNet-50 and Wide-ResNet-50-2 [35] are 61.38%, 73.61% and 81.55%, respectively. As illustrated in Figure 3, the average accuracy of all data augmentation methods is better than that of the baseline. Notably, among all data augmentation methods, AdvMask achieves the highest improvements

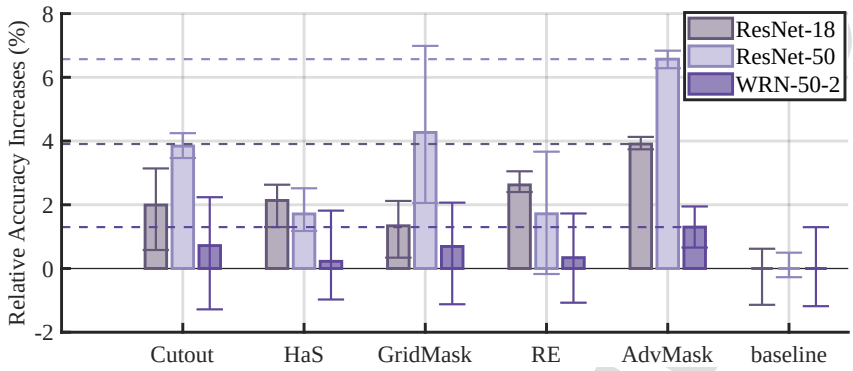


Figure 3: Relative classification accuracy compared with baseline on ResNet-18, ResNet-50, and WideResNet-50-2 for Tiny-ImageNet.

in accuracy using these deep models. Specifically, on ResNet-18, ResNet-50, and WideResNet-50-2, AdvMask improves the accuracy by 3.91%, 6.57%, and 1.30%, respectively. Notably, AdvMask also obtains the smallest error interval among all approaches, indicating its stability and effectiveness. Furthermore, different from others, the worst-case performance of AdvMask is still much higher than that of the baseline, highlighting the effectiveness of AdvMask. Especially on WRN-50-2, the worst-case accuracy of other methods is lower than the baseline, while the performance of AdvMask is stable at a relatively high level. In conclusion, AdvMask is a reliable and efficient data augmentation method for enhancing model performance.

Additional Comparisons on Tiny-ImageNet We also evaluate the efficacy of AdvMask in conjunction with other data augmentation techniques and present the improvements in the test accuracy in Figure 4. Meanwhile, the test accuracy of the original methods without AdvMask is presented in Table 4. It can be observed that AdvMask can consistently enhance the performance of various neural architectures on Tiny-ImageNet when used in combination with other augmentation methods. Notably, AdvMask achieves the highest test accuracy among all tested methods when combined with Fast-AutoAugment and TrivialAugment. While AdvMask only provides slight improvements in several cases, it consistently performs well in most cases and surpasses all other deletion-based methods. In conclusion, these findings underscore the practical and effective nature of AdvMask as a promising

method for improving the performance of deep models in image classification tasks.

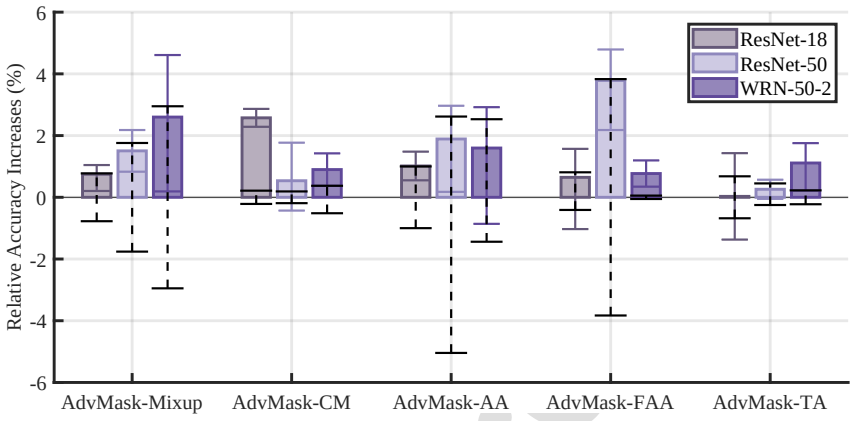


Figure 4: Relative classification accuracy of ResNet-18, ResNet-50, and WideResNet-50-2 models on Tiny-ImageNet when compared to original data augmentation methods. The black dashed plot represents the accuracy gap of the original methods (e.g., Mixup, etc.) with the average set at zero, while the colored plot is the accuracy gap of AdvMask-*. **CM**: CutMix.

Method	ResNet-18	ResNet-50	WRN-50-2
Mixup [14]	64.87	75.33	78.58
CutMix [15]	64.30	76.22	81.18
AutoAugment [8]	67.28	75.29	79.99
Fast-AutoAugment [9]	68.15	75.11	82.90
TrivialAugment [16]	69.97	79.23	82.16

Table 4: Test accuracy(%) on Tiny-ImageNet using various models architectures. **AA**: AutoAugment. **FAA**:Fast-AutoAugment. **TA**:Fast-AutoAugment.

4.4. Results on Oxford Flower Classification Dataset

This section aims to visually demonstrate the efficacy of AdvMask on the Oxford Flower Classification Dataset by utilizing class activation mapping (CAM) visualization. CAM visualization can identify the discriminative regions used by a classification model and provide visual explanations for

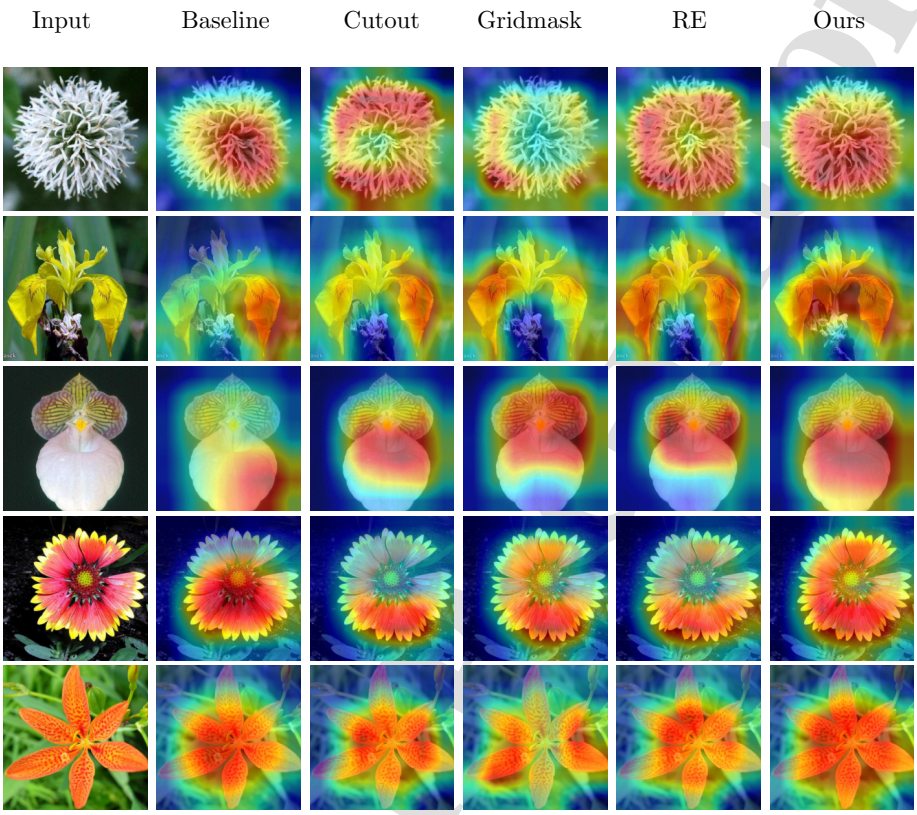


Figure 5: Class activation mapping (CAM) for ResNet-50 model trained on, with baseline augmentation, Cutout, GridMask, Random Erasing (RE), or our AdvMask. The models trained with AdvMask are inclined to focus on large important regions and cover a larger area of the object of interest.

models' performance [39]. Therefore, we visualize the CAM of the Oxford Flower dataset generated by the ResNet-50 model trained with various information deletion-based data augmentation approaches to further compare the performance of these models.

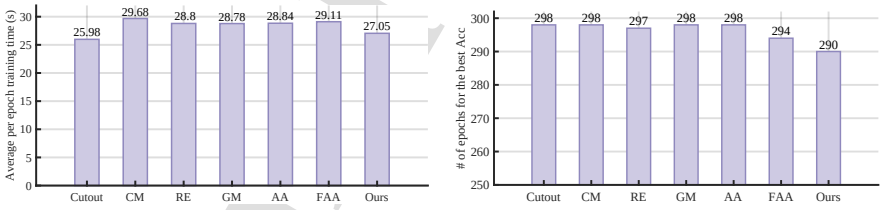
Firstly, Table 5 summarizes the test accuracy of these methods, and we observe that AdvMask outperforms other approaches by a significant margin consistently. Specifically, AdvMask outperforms the performance of Cutout by 3.04%, HaS by 7.45%, RandomErasing by 3.14%, and GridMask by 1.38%. At the same time, Figure 5 illustrates the CAM results of models trained with different methods. We can observe that the model trained with AdvMask

Method	baseline	Cutout [4]	HaS [5]	RE [2]	GM [3]	Ours
Accuracy (%)	80.20	88.53	84.12	88.43	90.19	**91.57**

Table 5: Image classification accuracy on Oxford Flower Classification Dataset.

is more inclined to locate and highlight the most relevant parts of the main objects, while the background is almost ignored. Both statistical and visual results suggest that successful data augmentation helps models focus on the most discriminative and salient areas in the images, thus can improve their generalization ability. For instance, in the second column of Figure 5, the baseline model's regions of interest cover only part of the main object (e.g., flower) and contain some irrelevant background information, indicating an overfitting problem. However, in the last column, the AdvMask's region of interest covers almost the entire flower while ignoring much of the background, indicating improved generalization ability. To summarize, statistical and visual experiment results demonstrate that AdvMask can improve deep models' generalization ability by forcing them to focus on the discriminative and salient areas of images.

4.5. Comparison of Training Efficiency



(a) Average per epoch training time.

(b) Number of epochs for convergence.

Figure 6: Training efficiency on CIFAR-10 using ResNet-18. **CM**: CutMix. **RE**: Random Erasing. **AA**: AutoAugment. **FAA**: Fast-AutoAugment.

Since data augmentation is solely applied to training data during the training phase, using the same deep model, different data augmentation approaches obtain the same test consumption time. Therefore, in this section, we provide additional training cost comparisons on the CIFAR-10 dataset using ResNet-18 architecture to compare the efficiency of various data augmentation methods.

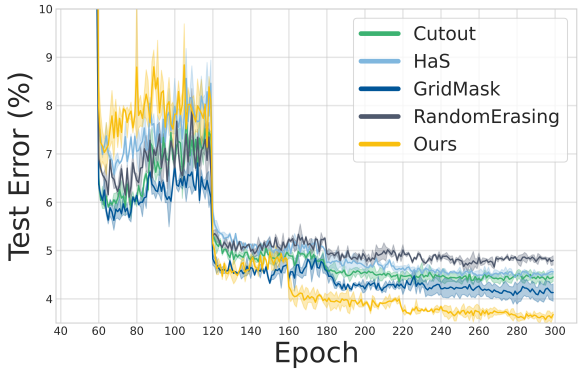


Figure 7: Curves of test errors on CIFAR-10 with ResNet-18.

All experiments are conducted on 2 NVIDIA RTX2080TI GPUs with batch size 128 and 8 parallel workers. The experiments are repeated across three independent random trials. The average per epoch time costs and the number of epochs for convergence are reported in Figure 6(a) and Figure 6(b), respectively. To ensure fairness, in Figure 6(a), we apply AdvMask augmentation to all images in every epoch during training instead of using the incremental generative strategy. It can be observed that although the training time computation of AdvMask is not the lowest, it achieves competitive time costs compared to other data augmentation approaches. Furthermore, the training epochs used by AdvMask to achieve the best are the lowest among various data augmentation approaches, demonstrating that AdvMask enhances the learning process and facilitates the model's ability to generalize better.

To more clearly present the dynamic evolution of test errors throughout the training process, we train ResNet-18 models on CIFAR-10 using a multi-step learning rate decay schedule. The learning rate is initialized as 0.1 and multiplied by 0.2 at epochs 60, 120, 160, 220, and 280. It is worth noting that all other experimental settings remained unchanged. Figure 7 illustrates the test error curves obtained from employing various information deletion-based methods. It is clear that AdvMask achieves a significant improvement after the third learning rate drop and shows even better performance after the fourth drop. Therefore, AdvMask enables models to learn robust and discriminative features more efficiently.

5. Ablation Studies

5.1. Attack Success Rate of Sparse Adversarial Attack Module

Threshold	Method	ASR(%)	l_0	l_2	l_∞
$\epsilon = 8/255$	JSMA	78.9	440.8	0.611	0.031
	PGD-$l_0 + l_\infty$	73.9	1199.7	1.078	0.031
	GreedyFool	100.0	468.2	0.547	0.031
	C&W-l_0	100.0	326.6	0.542	0.068
	SAPF	100.0	321.8	**0.523**	0.085
	Ours	**100.0**	**320.1**	0.532	**0.031**
$\epsilon = 16/255$	JSMA	97.3	247.7	0.896	0.063
	PGD-$l_0 + l_\infty$	72.8	498.0	1.390	0.063
	GreedyFool	100.0	238.3	0.707	0.063
	C&W-l_0	100.0	136.7	0.691	0.118
	SAPF	100.0	133.7	0.718	0.159
	Ours	**100.0**	**131.3**	**0.689**	**0.063**

Table 6: Results of targeted sparse adversarial attack on CIFAR-10.

We present an evaluation of our adversarial attack module using attack success rate (ASR) and different l_p-norms ($p = 0, 2, \infty$) to assess its overall performance. The ASR measures the proportion of misclassified adversarial samples in all samples. ℓ_0-norm indicates the number of non-zero elements, with a lower value indicating fewer perturbed pixels. ℓ_2 measures the distance between the adversarial image and the original clean image. ℓ_∞ represents the maximum value of the perturbation magnitude, with a higher value indicating a greater pixel change. To demonstrate the effectiveness of our proposed sparse adversarial attack module, we compare it with several SOTA sparse adversarial attack methods, including JSMA, C&W-ℓ_0, PGD-$\ell_0 + \ell_\infty$, GreedyFool, and SAPF. The average ℓ_p norm and ASR under two ϵ settings are shown in Table 6, and ϵ is the maximum perturbation magnitude. Under both ϵ settings, we can achieve 100% ASR, while the ℓ_0 and ℓ_∞ are the lowest, indicating that the number of perturbed pixels and the maximum value of the perturbation magnitude are both the least. Therefore, we have proved the effectiveness of our proposed sparse adversarial attack module for identifying the most classification-critical points.

5.2. Effect of the Adversarial Attack Points

To examine the significance of adversarial attack points, we employ three distinct point selection strategies: 1) random points (RP), where the points

Method	ResNet-18 Acc (%)	ResNet-50 Acc (%)
Baseline	95.28	94.12
Random Points	95.66	95.61
Corner Points	95.11	95.41
Salient Points	95.78	95.77
AdvMask	**96.23**	**96.69**

Table 7: Accuracy (%) with different points selection strategies on CIFAR-10 using both ResNet-18 and ResNet-50.

are selected randomly; 2) corner points (CP), where the key points are determined using conventional corner detection; and 3) salient points (SP), where key points are selected from the saliency map [40]. The data augmentation module remains unaltered.

As indicated in Table 7, all point selection strategies can enhance the classification accuracy over the baseline. However, our proposed approach outperforms the others significantly on both ResNet-18 and ResNet-50. In particular, on ResNet-50, the accuracy of AdvMask is 0.92%, 1.08%, and 1.28% higher than that of SP, RP, and CP, respectively. The results show that conventional key point selection algorithms are inadequate for pinpointing classification-critical areas in images, thus, are not helpful for downstream data augmentation operations. Therefore, we experimentally demonstrate the efficacy of the proposed adversarial attack points. Although obtaining adversarial attack points for images inevitably introduces additional training costs, utilizing our carefully devised sparse adversarial attack module can maximize efficiency while obtaining satisfactory performance. Lastly, we will open-source experimental results of adversarial attack points for several datasets soon so that researchers and practitioners can use and build upon our work to achieve further improvements.

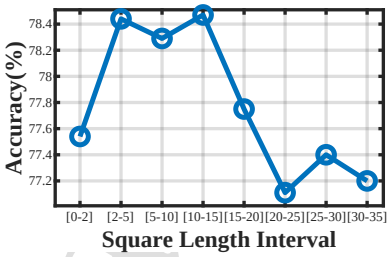
5.3. Effect of Data Augmentation Module

To validate the efficacy of our data augmentation module, we conducted experiments using attack masks as augmentation masks, where the masked points are all adversarial attack points with 1×1 square. Table 8 presents the accuracy, and we can see that the accuracy of using attack masks is much lower than that of our method. The accuracy of using attack masks is even worse than the baseline on ResNet-18. This is because the masked regions are

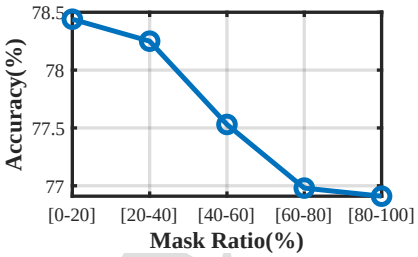
Method	ResNet-18 Acc (%)	ResNet-50 Acc (%)
AAPM	92.96	94.52
Baseline	95.28	94.12
AdvMask	**96.23**	**96.69**

Table 8: Test accuracy on CIFAR-10 on ResNet-18 and ResNet-50. AAPM: adversarial attack point mask.

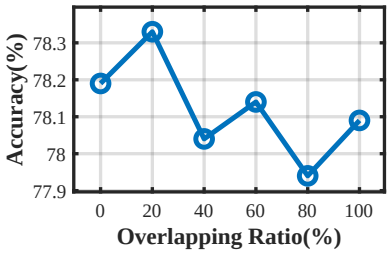
so small and thus bring noises into images. Influenced by the noise, the classifier misclassifies much more samples, leading to this side effect. Differently, our data augmentation module effectively leverages structured deletion on these adversarial attack points to simulate situations where the key information used for classification is partly lost. Therefore, we demonstrate that the data augmentation module can enhance the generalization and robustness of deep models.



(a) Effect of l.



(b) Effect of p.



(c) Effect of o.

Figure 8: The effect of parameters of AdvMask on the test accuracy.

5.4. Effect of Parameters

To investigate the effect of AdvMask's parameters on the model's performance, we conduct experiments on CIFAR-100 with varying parameter settings.

Parameter l. We set up eight disjoint intervals for the length of the mask square, denoted as l. As shown in Figure 8(a), there is a trend of initially increasing and then decreasing accuracy with increasing l. When l is too small or too large, the classification accuracy tends to decrease, especially when l is too large. This is because when the mask is too large, it is more likely to cover the entire object of interest, leading to undesirable side effects on the classifier. Therefore, the maximum value of l should be limited, and the variety of l is beneficial to improve the overall performance of AdvMask.

Parameter p. By adjusting the value of p, AdvMask controls the total masked areas to avoid being too small or too large. In particular, we set five intervals for p with length 20% from 0 to 100%. As revealed in Figure 8(b), when p is relatively small, the difference in accuracy is not noticeable. However, when p is further increased, accuracy drops drastically, with the lowest accuracy obtained when p is at its maximum value. The classification accuracy varies by up to 1.53%. This is because a too-large masked area would damage the classifier's performance by masking most of the image.

Parameter o. To maintain the structured information of images, the deletion of contiguous regions in a single image is avoided by controlling the parameter o. As illustrated in Figure 8(c), under the control of both l and p, the influence of o is insignificant. It is observed that the accuracy is relatively higher when the overlapping ratio is below 20%. However, increasing the value of o leads to a decrease in accuracy, with a maximum drop of 0.39%. Smaller values of o are preferred to preserve structured information and avoid the complete removal or reservation of the main object.

Parameter Settings. In general, parameters l, p, and o can be easily set based on the statistics of the dataset. Typically, the maximal value of p is set no larger than 50% to ensure that the deleted regions are not too large. The interval for l is determined by l_{min} and l_{max}. Because dropping a tiny region is useless for the convolution operation, l_{min} should not be lower than 2, usually in $[2, 5]$. Under the constraint of $p \leq 50\%$, the maximum value of l should not exceed 65% of the image length; thus, the area of a single maximal dropping

square is less than 42.25% of the image area. As illustrated in Figure 8(c), the influence of o is not significant when p and l are constrained. Therefore, to facilitate experimentation, o is typically set to 20%.

6. Conclusion

In this paper, we propose AdvMask, a novel data augmentation method for image classification. AdvMask first employs a sparse adversarial attack module to identify critical points in images and randomly occludes structured regions based on these key points in each iteration. In addition, AdvMask can be used alone or in conjunction with other data augmentation methods. AdvMask reduces the sensitivity of deep models to occlusion and effectively enhances the generalization ability of models by forcing models to focus more on less sensitive areas with important features. Extensive experiments have proven that AdvMask improves deep model performance significantly and consistently outperforms other data augmentation approaches.

Besides strengths, the limitations of our method should also be mentioned. Firstly, while adversarial attack points effectively capture critical information in images, obtaining the adversarial attack points inevitably introduces additional training costs. Secondly, although we have provided parameter-setting suggestions for AdvMask, determining the optimal values remains a challenge.

In the future, several extensions and improvements can be explored. Firstly, to mitigate the computational costs associated with the sparse adversarial attack module, the adoption of our carefully devised sparse adversarial attack module can minimize the costs while obtaining satisfactory performance. Moreover, Moreover, future research may focus on developing a fast algorithm for estimating adversarial attack points, further enhancing the efficiency of our method. Secondly, given the involvement of multiple parameters in AdvMask, it is worthwhile to investigate whether the optimal parameter values can be learned. Similar to the approach employed in Fast-AutoAugment [9], we can employ a search strategy to determine the AdvMask policies with optimal parameter values. Lastly, the application of AdvMask to other computer vision tasks, such as object detection and semantic segmentation, holds promise for future research.

In conclusion, the proposed AdvMask method demonstrates significant improvements in deep model performance for image classification. We hope

that our work will inspire further research into the development of data augmentation methods.

Acknowledgments

This work was supported in part by the STI 2030-Major Projects of China under Grant 2021ZD0201300, and by the National Science Foundation of China under Grant 62276127.

References

[1] A. M. Obeso, J. Benois-Pineau, M. S. G. Vázquez, A. Á. R. Acosta, Visual vs internal attention mechanisms in deep neural networks for image classification and object detection, Pattern Recognition 123 (2022) 108411.

[2] Z. Zhong, L. Zheng, G. Kang, S. Li, Y. Yang, Random erasing data augmentation, in: Proc. AAAI, Vol. 34, 2020, pp. 13001–13008.

[3] P. Chen, S. Liu, H. Zhao, J. Jia, Gridmask data augmentation, arXiv preprint arXiv:2001.04086 (Jan 2020).

[4] T. DeVries, G. W. Taylor, Improved regularization of convolutional neural networks with cutout, arXiv preprint arXiv:1708.04552 (Nov 2017).

[5] K. K. Singh, Y. J. Lee, Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization, in: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), IEEE, 2017, pp. 3544–3553.

[6] C. Gong, D. Wang, M. Li, V. Chandra, Q. Liu, Keepaugment: A simple information-preserving data augmentation approach, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 1055–1064.

[7] F. Dornaika, D. Sun, K. Hammoudi, J. Charafeddine, A. Cabani, C. Zhang, Object-centric contour-aware data augmentation using superpixels of varying granularity, Pattern Recognition 139 (2023) 109481.

[8] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, Q. V. Le, Autoaugment: Learning augmentation strategies from data, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 113–123. doi:10.1109/CVPR.2019.00020.

[9] S. Lim, I. Kim, T. Kim, C. Kim, S. Kim, Fast autoaugment, in: Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019.
URL https://proceedings.neurips.cc/paper/2019/file/6add07cf50424b14fdf649da87843d01-Paper.pdf

[10] H. Huang, D. Li, Z. Zhang, X. Chen, K. Huang, Adversarially occluded samples for person re-identification, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018.

[11] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images (2009).

[12] P. Chrabaszcz, I. Loshchilov, F. Hutter, A downsampled variant of imagenet as an alternative to the cifar datasets, arXiv preprint arXiv:1707.08819 (Aug 2017).

[13] M.-E. Nilsback, A. Zisserman, Automated flower classification over a large number of classes, in: Proceeding of the Sixth Indian Conference on Computer Vision, Graphics and Image Processing, 2008, pp. 722–729. doi:10.1109/ICVGIP.2008.47.

[14] H. Zhang, M. Cisse, Y. N. Dauphin, D. Lopez-Paz, mixup: Beyond empirical risk minimization, in: International Conference on Learning Representations, 2018.
URL https://openreview.net/forum?id=r1Ddp1-Rb

[15] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, Y. Yoo, Cutmix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6023–6032.

[16] S. G. Müller, F. Hutter, Trivialaugment: Tuning-free yet state-of-the-art data augmentation, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 774–782.

[17] T. V. Maliamanis, K. D. Apostolidis, G. A. Papakostas, How resilient are deep learning models in medical image analysis? the case of the moment-based adversarial attack (mb-ada), Biomedicines 10 (10) (2022). doi:10.3390/biomedicines10102545.
URL https://www.mdpi.com/2227-9059/10/10/2545

[18] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, A. Swami, The limitations of deep learning in adversarial settings, in: Proc. IEEE European Symposium on Security and Privacy (EuroS&P), IEEE, 2016, pp. 372–387.

[19] N. Carlini, D. Wagner, Towards evaluating the robustness of neural networks, in: Proc. IEEE European Symposium on Security and Privacy (EuroS&P), IEEE, 2017, pp. 39–57.

[20] F. Croce, M. Hein, Sparse and imperceivable adversarial attacks, in: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), 2019, pp. 4724–4732.

[21] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: Proc. Int. Conf. on Learning Representations, 2018.
URL https://openreview.net/forum?id=rJzIBfZAb

[22] A. Modas, S.-M. Moosavi-Dezfooli, P. Frossard, Sparsefool: a few pixels make a big difference, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 9087–9096.

[23] X. Dong, D. Chen, J. Bao, C. Qin, L. Yuan, W. Zhang, N. Yu, D. Chen, Greedyfool: Distortion-aware sparse adversarial attack, in: Proc. Adv. Neural Inf. Process. Syst., Vol. 33, 2020, pp. 11226–11236.

[24] Y. Fan, B. Wu, T. Li, Y. Zhang, M. Li, Z. Li, Y. Yang, Sparse adversarial attack via perturbation factorization, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2020, pp. 35–50.

[25] B. Wu, B. Ghanem, lp-box admm: A versatile framework for integer programming, IEEE Trans. Pattern Anal. Mach. Intell. 41 (7) (2018) 1695–1708.

[26] J.-H. Luo, J. Wu, Autopruner: An end-to-end trainable filter pruning method for efficient deep model inference, Pattern Recognition 107 (2020) 107461.

[27] Y. He, X. Zhang, J. Sun, Channel pruning for accelerating very deep neural networks, in: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), 2017, pp. 1389–1397.

[28] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent., Springer, 2015, pp. 234–241.

[29] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, J. Li, Boosting adversarial attacks with momentum, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 9185–9193.

[30] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (6) (2017) 84–90.

[31] C. Zhao, X. Lv, S. Dou, S. Zhang, J. Wu, L. Wang, Incremental generative occlusion adversarial suppression network for person reid, IEEE Trans. Image Process. 30 (2021) 4212–4224. doi:10.1109/TIP.2021.3070182.

[32] X. Wang, Y. Chen, W. Zhu, A survey on curriculum learning, IEEE Trans. on Pattern Anal. Mach. Intell. (2021) 1–1doi:10.1109/TPAMI.2021.3069908.

[33] P. Soviany, R. T. Ionescu, P. Rota, N. Sebe, Curriculum learning: A survey, International Journal of Computer Vision 130 (6) (2022) 1526–1565.

[34] X. Gastaldi, Shake-shake regularization, CoRR abs/1705.07485 (May 2017). arXiv:1705.07485.
URL http://arxiv.org/abs/1705.07485

[35] S. Zagoruyko, N. Komodakis, Wide residual networks, in: E. R. H. Richard C. Wilson, W. A. P. Smith (Eds.), Proceedings of the British Machine Vision Conference (BMVC), BMVA Press, 2016, pp. 87.1–87.12. doi:10.5244/C.30.87.
URL https://dx.doi.org/10.5244/C.30.87

[36] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.

[37] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, T. Mann, Data augmentation can improve robustness, in: A. Beygelzimer,

Y. Dauphin, P. Liang, J. W. Vaughan (Eds.), Advances in Neural Information Processing Systems, 2021.
URL https://openreview.net/forum?id=kgVJBBThdSZ

[38] C. Gong, D. Wang, M. Li, V. Chandra, Q. Liu, Keepaugment: A simple information-preserving data augmentation approach, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 1055–1064.

[39] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

[40] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, in: Y. Bengio, Y. LeCun (Eds.), 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings, 2014.
URL http://arxiv.org/abs/1312.6034

Suorong Yang received the B.S. degree in the department of Computer Science and Technology from Nanjing University in 2019. He is pursuing the Ph.D. degree in Computer Science and Technology at Nanjing University under the supervision of Prof. Furao Shen. His research interests include computer vision, data augmentation, generative adversarial network, etc.

Jinqiao Li received the B.S. degree in College of Computer and Information in 2019 from Hohai University, and the M.Sc. degree in Computer Science and Technology from Nanjing University, in 2022. His current research interests include computer vision and adversarial attack, etc.

Tianyue Zhang received the B.S. degree in the department of Computer Science and Technology from Nanjing University in 2013. She is pursuing the Ph.D. degree in Computer Science and Technology at Nanjing University under the supervision of Prof. Furao Shen. Her research interests include computer vision, few-shot learning, etc.

Jian Zhao (Senior Member, IEEE) received the B.S. degree from Nanjing University, Nanjing, China, the M.Sc. degree from the Hamburg University of Technology, Hamburg, Germany, and the Dr. Sc. degree in electrical engineering from the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland. From 2010 to 2015, he was a Research Scientist with the Institute for Infocomm Research, A*STAR, Singapore. Currently, he is an Associate Professor with the School of Electronic Science and Engineering, Nanjing University. His research interests include deep neural networks, mathematical optimization, and wireless communication networks. Dr. Zhao was honored with the Dengfeng Scholars Program of Nanjing University in 2015, IEEE Globecom 2008 Best Paper Award, and the 2009 Chinese Government Award for Outstanding Self-Financed Students Abroad.

Furao Shen (Member, IEEE) received the B.Sc. and M.Sc. degrees in mathematics from Nanjing University, Nanjing, China, in 1995 and 1998, respectively, and the Ph.D. degree from the Tokyo Institute of Technology, Tokyo, Japan, in 2006. He is currently a Full Professor of computer science and technology with Nanjing University. His current research interests include neural computing and robotic intelligence.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: