Pattern Recognition 149 (2024) 110240

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Network characteristics adaption and hierarchical feature exploration for robust object recognition

Weicheng Xie [a,b,c], Cheng Luo [a,b,c], Gui Wang [d,e], Linlin Shen [a,b,c,*], Zhihui Lai [a,b,c], Siyang Song [f]

[a] National Engineering Laboratory for Big Data System Computing Technology, School of Computer Science and Software Engineering, Shenzhen University, China
[b] Shenzhen Institute of Artificial Intelligence and Robotics for Society, China
[c] Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, China
[d] College of Computer Information System, Wenzhou-Kean University, China
[e] Department of Computer Science, University of Nottingham Ningbo, China
[f] School of Computing and Mathematical Sciences, University of Leicester, UK

ARTICLE INFO

Keywords:
Robust object recognition
Attention-based dropout
Adaptive characteristics
Hierarchically-salient features

ABSTRACT

Recent advances in deep networks have achieved appealing performances on object recognition tasks, due to their robust feature learning abilities. Besides the generated deep features, other network characteristics, e.g., inter-layer weight matrix and their back-propagated derivatives, may behave complementarily in feature learning in terms of generalization and robustness performances. However, characteristics adaptivity to different databases is not well studied. Meanwhile, current algorithms are apt to explore the most salient features for better generalization performance, while the hierarchically-salient features that may be beneficial for network robustness are not fully explored. Thus, we propose an attention module to make network characteristics adaptive to different training tasks, which can be further combined with the dynamic dropout algorithm to suppress salient neurons to explore more SndMS (Second Most Salient) features for robust recognition. The proposed algorithm has two main merits. First, the complementarity of network characteristics is taken into account when conducting training on different databases; Second, with the exploration of more SndMS neurons for hierarchically-salient feature representation learning, the network robustness against adversarial perturbations or fine-grained differences can be enhanced. The extensive experiments on seven public databases show that the proposed attention-based dropout largely improves the network robustness, without compromising the generalization performance, compared with related variants and state-of-the-art (SOTA) algorithms. Algorithm codes are available at https://github.com/lingjivoo/ACAD.

1. Introduction

With the development of computer vision, models for optimizing deep learning-based robust networks against data variations have attracted increasing interest, which is frequently resorted to the optimization of robust feature representation, and the learned information is mainly specific to the considered training samples. In contrast, network weights are optimized by all training samples, and can be treated as a specific global representation for them [1], which are complementary to network features. Specifically, for each sample, in addition to the regular network feature, the network weights will also be changed during the training according to this sample. Consequently, combining these two types of information may result in a more robust and complementary representation for this sample.

In addition to latent features and network's weights, the information embedded in back-propagated derivative w.r.t. (with respect to) feature or weight is also informative for network adaptivity. Samples contributing more to the back-propagation gradient should have a larger probability to be selected and augmented to improve the network robustness and generalization performances [2], and using derivatives to weigh feature maps allows features to be more transferable [3]. However, the application of the network's weights to learn feature representations has not been fully explored, let alone the use of their derivatives for feature adaption.

To take weights and derivatives into account for the adaptivity of feature representation and make use of their complementarity during training, this paper resorts to the network adaptivity-resemble modules, e.g. [4], where **characteristics** are defined as the features

* Corresponding author at: National Engineering Laboratory for Big Data System Computing Technology, School of Computer Science and Software Engineering, Shenzhen University, China.
E-mail address: llshen@szu.edu.cn (L. Shen).

https://doi.org/10.1016/j.patcog.2023.110240
Received 7 April 2023; Received in revised form 13 December 2023; Accepted 26 December 2023
Available online 5 January 2024

generated by fully connected (FC) layers, network weights, and the related **derivatives**. Specifically, we employ an attention mechanism to achieve training adaptivity by optimizing characteristics contributions with back propagation [5].

Characteristics adaptivity allows networks to locate the most salient features, which may yield reasonable generalization performance at the inference stage. However, such an encoding strategy with the most salient features is not robust against the sample perturbation, since current effective adversarial attacks are often performed on the most salient regions by imposing perturbation noises [6]. To learn features for more robust recognition, the researchers have recently paid attention to hierarchical features or the SndMS features [7], where *hierarchical saliency features are assertively defined* as the features with multiple saliency levels, i.e. the most salient, SndMS and non-salient features, based on the intensity of the response. Xie et al. [8] explored SndMS features for object localization and semantic segmentation.

To decrease network complexity, the dropout algorithm [9] has been frequently employed to learn hierarchical feature representations. It is suggested to dropout the feature maps deterministically [10] or generate a dropout mask based on the attention output [11] to extend the localization regions. Other variants of dropout are also proposed to encourage neurons with small activations [12], strengthen the influence of less important units, or hide the most discriminative regions stochastically. Though activated less-salient features, these studies did not take into account the hierarchically-salient features for recognition. Meanwhile, the update of dropout probabilities is mainly based on the instance-specific feature cue, the dataset-common cues are not sufficiently considered.

In this work, different from previous studies that drop the most salient features mainly based on the activations, we resort to exploring the hierarchically-salient features with dynamic dropout probability based on characteristics adaptivity. Specifically, by making the dropout probabilities adapt to network characteristics, i.e. features, weights and their derivatives, our attention-based dropout enables the network to dynamically explore SndMS features. In this way, we can trade off the generalization and robustness performances, and especially improve the network robustness against adversarial perturbations. The main contributions of this work are summarized as follows:

- An attention model is proposed to make use of the complementarity of different network characteristics and adapt them to different databases. To the best of our knowledge, this work is the first to adapt network features, weights, and derivative w.r.t. weights for feature representation;
- An attention-based dropout is proposed to dynamically suppress the salient features and activate more SndMS features, so as to produce hierarchical features for robust recognition of objects with fine-grained differences;
- Extensive experimental results on seven publicly available datasets show that the proposed algorithm can not only well maintain the generalization performances, but also largely improve network robustness against adversarial perturbations, compared with the related variants and SOTAs.

This work is structured in the following sections. The related works are surveyed in Section 2. The proposed algorithm is demonstrated in Section 3. Then the experimental results and the corresponding illustrations are demonstrated in Section 4. Finally, the conclusion and discussions are presented in Section 5.

2. Related works

2.1. Dynamic feature representation learning

For dynamic feature learning, the attention mechanism has been widely employed to allow networks to focus on the specific activation region [13], features of object parts [14], subtle visual structure [15], what and where to emphasize or suppress the intermediate features [10], weakly supervised metric and template learning integrated with sample reliability representation [16].

However, these algorithms mainly used the information of a training batch for the feature representation of a specific sample. By contrast, network weights encode the information of the entire training samples [1]. However, they are rarely used for the dynamic feature representation, let alone the complementarity of these features and weights for adapting to different databases. In this work, this complementarity will be taken into account for the feature representation training based on the attention mechanism.

2.2. Hierarchical feature exploration

While traditional feature representation learning mainly highlights the most salient features, the usefulness of SndMS features for the network's robustness against perturbation noises, especially for the adversarial noises [6], is not explored. Various algorithms [7] for exploring hierarchical features with different levels of salient responses were developed.

By randomly dropping or suppressing features, the variants of the dropout algorithm [17,18], and attention-based region and channel zeroing were widely used to enhance network representation capacity and improve its robustness against perturbation noises. Based on the attention output, the dropout mask was generated with the proposed iterative ADL (attention and dropout layer) [10] or the prior of salient regions [11], which is further used to guide networks to explore less discriminative parts for object localization. Keshari et al. [12] proposed a deterministic dropout, i.e. the guided dropout, to take into account the neurons with small activations.

In contrast, we explore the hierarchically-salient features based on dynamic dropout probabilities of characteristics adaptivity, thus, SndMS features could be activated in the way of characteristic dynamic perception, for robust recognition. More importantly, characteristics adaptivity-based dropout can generate adaptive hierarchical features for different datasets.

2.3. Trade off between generalization and robustness

The network generalization, as well as robustness, are both popular metrics to evaluate the performance of a network. Stutz et al. [19] revealed that the robustness metric is not contradictory to generalization when the attacked samples are embedded to a low-dimensional manifold. Pang et al. [20] proposed a diversity-promoting regularizer to improve adversarial robustness, without deteriorating generalization performance.

Wang et al. [21] suggested to automatically adapt attention to facial regions with different discrimination abilities and scales to improve generalization capacity. Correspondingly, by suppressing attention maps or distracting from the original heat map, Chen et al. [6] introduced a successful attack on recognition models.

Although network generalization and robustness are not contradictory, it is a challenge to trade off these two goals. In this work, we use the characteristics adaptivity to maintain the network generalization, an attention-based dropout is then proposed to dynamically activate the SndMS responses and produce hierarchical features, so as to improve network robustness.

3. The proposed algorithm

In this section, we introduce the main motivation, the overall framework, and the specific modules of the proposed algorithm sequentially.

3.1. Motivation

The main motivations of the network characteristic adaptivity and hierarchical feature exploration in our algorithm are shown in Fig. 1.

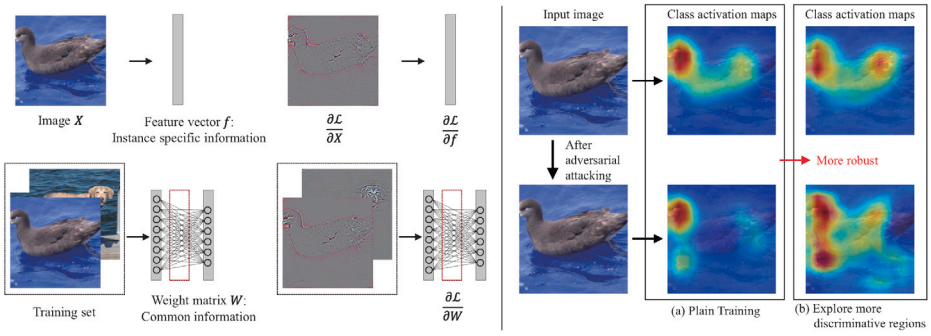


Fig. 1. Motivation of network characteristic adaptivity and hierarchical feature exploration. Left: four different characteristics represent complementary information for recognition. Right: general recognition with only the most discriminative regions performs less robust than that with more discriminative cues. When attacked by FGSM [22], a network with plain training attends to suppress discriminative regions (a). (b) The network that explores more discriminative cues keeps attending to complete class-related regions.

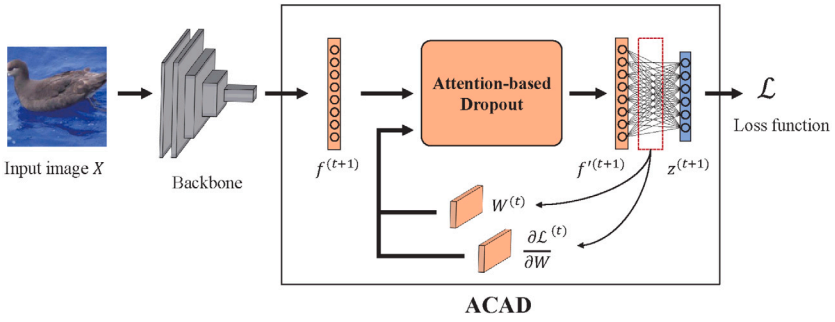


Fig. 2. Diagram of network framework with the proposed ACAD (Adaptive Characteristic Attention-based Dropout). ACAD is performed on the last FC layer, which is clarified in Fig. 4. W and $\frac{\partial \mathcal{L}}{\partial W}$ are obtained in the preceding BST (batch-size training).

- As shown in Fig. 1, feature vector f represents instance-specific information; learned weight matrix W implies common information among the whole training set; and the backpropagated gradient ($\frac{\partial \mathcal{L}}{\partial f}$ or $\frac{\partial \mathcal{L}}{\partial W}$) implies sensitive parts of features or weight matrix. It is also revealed in [23] that different network characteristics behave diversely on different databases, which motivates us to leverage the complementarity of these characteristics based on their adaptivity. In this work, we use this complementary information, instead of solely instance-specific features, for maintaining the generalization capacity in robust network learning.
- Fig. 1 shows that learning-based classifiers easily attend to objects' most discriminative regions. However, they may ignore the less discriminative cues for recognition robustness, i.e. it may cause a large performance drop when their focused regions are attacked and distracted. To alleviate this problem, we attend to the most discriminative regions and suppress their contribution to recognition with characteristic-adaptive dropout probabilities. In this way, classifiers could explore less discriminative cues and produce hierarchical features to improve network robustness against image corruptions or adversarial attacks.

3.2. Algorithm overview

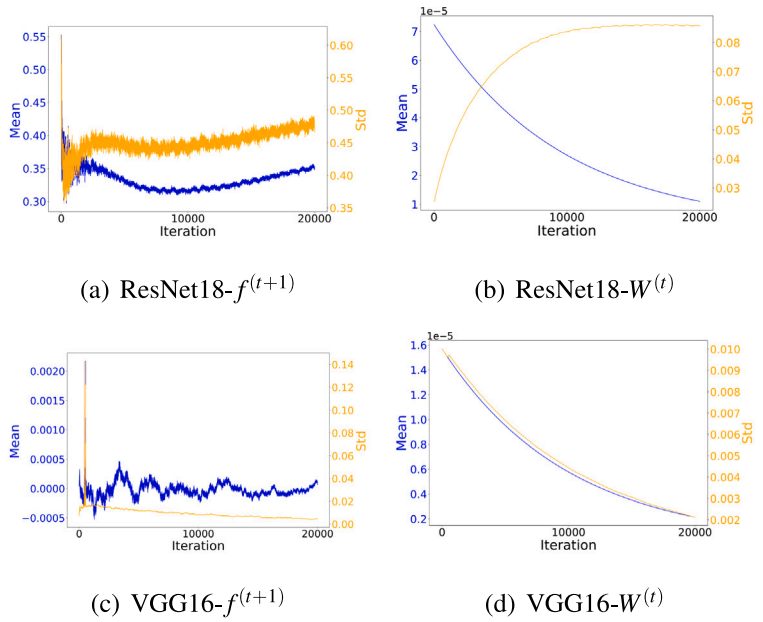
The proposed algorithm is illustrated in Fig. 2, it is shown that our algorithm is applied to the embedding feature representation. Specifically, given a FC (fully connected) layer at the $(t+1)$th training iteration, we jointly feed (1) its input $f^{(t+1)} \in \mathbb{R}^n$, (2) its weight matrix $W^{(t)} \in \mathbb{R}^{\#\text{class} \times n}$ and (3) the derivatives $\frac{\partial \mathcal{L}}{\partial W}^{(t)} \in \mathbb{R}^{\#class \times n}$ of the network loss $\mathcal{L}$ w.r.t. the weight matrix, into an attention model, aiming to adapt characteristic contributions to the training on different databases. The attention output is further used for the dynamic update of dropout probabilities to enable the network to explore hierarchical features, i.e. $f'^{(t+1)}$, for robust recognition.

3.3. Characteristic selection

It is revealed in the study [24] that the gradient information w.r.t. network parameters before the current iteration is beneficial for feature representation, which motivated us to leverage the derivatives w.r.t. weights and feature representation from the preceding BST (batch-size training), for the optimization of feature representation in current BST. However, mismatching of latent information representation may happen due to the possible large variation of features or weights between consecutive BSTs. If this case happens, the derivatives w.r.t. features or weights in each two consecutive BSTs also represent mismatched information cues, indicating that the features or weights in the preceding BST are unsuitable to be equipped with those in current BST for feature representation learning. To study this variation, the evolution curves of their averages against the numbers of batch sizes are presented in Fig. 3.

As shown in Fig. 3, the features behave relatively unstably compared with the weights for both two networks, that is, the features between two consecutive BSTs, i.e. $f^{(t)}$ and $f^{(t+1)}$, show relatively larger variations compared with $W^{(t)}$ and $W^{(t+1)}$. This is because the feature derivatives from the preceding BST are specific to the preceding batch of samples, which are likely to be largely different from the feature derivatives specific to the current batch. By contrast, the weight derivatives behave stably since they reflect all the information of the already learned samples.

Consequently, $\frac{\partial \mathcal{L}}{\partial f}^{(t)}$ represents mismatched information with that by $\frac{\partial \mathcal{L}}{\partial f}^{(t+1)}$, which is not applicable for the following feature representation learning in the $(t+1)$th BST, while only $W^{(t)}$, $\frac{\partial \mathcal{L}}{\partial W}^{(t)}$ and $f^{(t+1)}$ are employed.



(a) ResNet18-$f^{(t+1)}$ (b) ResNet18-$W^{(t)}$

(c) VGG16-$f^{(t+1)}$ (d) VGG16-$W^{(t)}$

Fig. 3. Evolutions of means and std (standard deviations) of FC layer features f and weights W for CIFAR100 against the number of training batches for ResNet18 and VGG16.

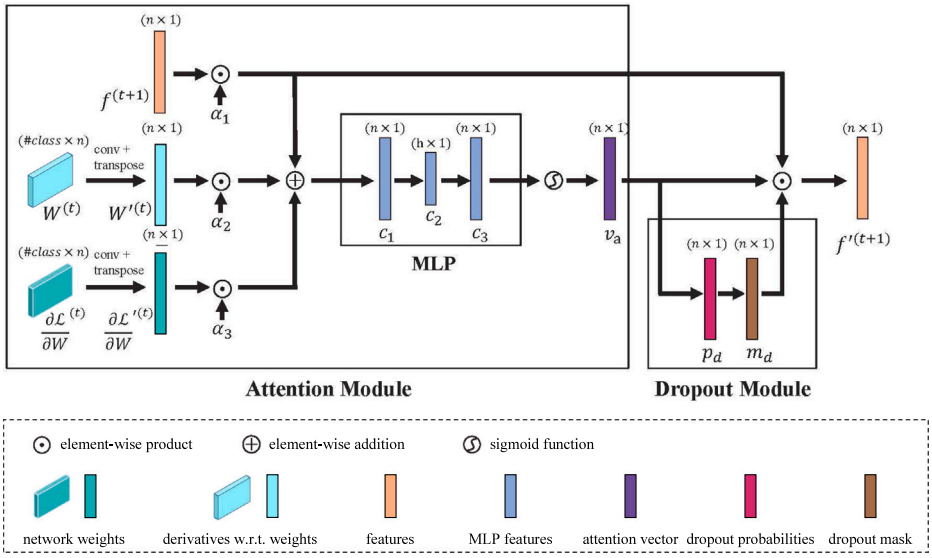


Fig. 4. Diagram of the proposed ACAD module, which consists of two sub-modules, i.e. characteristic attention and attention-based dropout.

3.4. Characteristic attention

To make characteristic contributions adaptive to different databases, an attention operator is proposed to dynamically adjust the regularization weights of different characteristics. The diagram of the proposed ACAD is presented in Fig. 4, where a MLP (multi-layer perception) is employed to normalize different characteristics before the attention.

As shown in Fig. 4, the proposed attention mechanism involves three characteristics, the input vector of the last FC layer in the current BST, i.e. $f^{(t+1)}$, the weight matrix between the FC layer input and output in the preceding BST, i.e. $W^{(t)}$, and the derivatives of the loss w.r.t. $W^{(t)}$ in the preceding BST, i.e. $\frac{\partial \mathcal{L}}{\partial W}^{(t)}$. In order to unify the dimensions of different characteristics, $W^{(t)}$ and $\frac{\partial \mathcal{L}}{\partial W}^{(t)}$ are compressed by the corresponding convolution operation ($h_w^{1\times1}$ and $h_{wg}^{1\times1}$, respectively) with a filter size of 1×1. In addition, regularization weight $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ is introduced to scale each characteristic, which also determines the characteristic contribution for recognition. The fusion of different characteristics is followed by a MLP with an activation function to generate an attention vector $v_a \in \mathbb{R}^n$ as follows

$$v_a = \sigma(c_3) \tag{1}$$

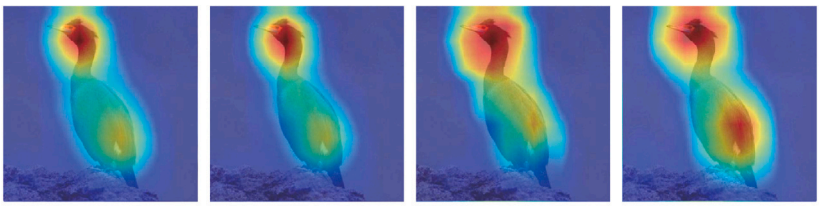
where σ denotes the added sigmoid activation function, c_3 is the MLP representation of the weighted FC output, and formulated as follows

$$c_3 = MLP(c_1) \triangleq W_1 c_2 = W_1(W_0 c_1) \tag{2}$$

where c_2 denotes the hidden layer output in Fig. 4, whose dimension varies in different models. The weighted FC output c_1 is formulated as follows

$$c_1 = \alpha_1 f^{(t+1)} + \alpha_2 h_w^{1\times1}(W^{(t)})^{\mathrm{T}} + \alpha_3 h_{wg}^{1\times1}(\frac{\partial \mathcal{L}}{\partial W}^{(t)})^{\mathrm{T}} \tag{3}$$

where $h_w^{1\times1}$ and $h_{wg}^{1\times1}$ denote convolution operations with the filter size of 1×1 to compress characteristic dimensions, and T denotes the transpose operation.



(a) Non. (b) Ori. (c) ADL (d) ACAD

Fig. 5. Grad-CAM (Gradient-weighted Class Activation Mapping) of the last convolution layer output of ResNet18 using different dropout strategies. Notations of 'Non.', 'Ori.' and 'ADL' are the abbreviations of 'Non-Dropout', 'Original Dropout', and 'Iterative Attention and Dropout Layer' [10], respectively.

With the proposed characteristic attention, adaptive contribution weighting of various characteristics for different datasets is achieved, which not only explores both instance-specific and dataset-common cues but also takes the network training situation and iteration direction into account.

3.5. Attention-based dropout

While the attention mechanism can explore the most salient features to enhance generalization performance, the SndMS features that are critical to robustness performance have not received enough attention. Thus, an attention-based dropout is proposed to enable the network to explore the attention features with hierarchical-salience levels, so as to alleviate the influence of perturbation noises, e.g. adversarial noises on recognition performance. Specifically, the distribution of the attention map is regularized, i.e. dropout is employed to suppress the most salient features, which encourages the characteristics attention module to explore the SndMS features.

As shown in Fig. 4, the proposed dropout is based on the attention vector obtained in Eq. (1). More precisely, the output of the attention module, i.e. v_a, is mapped to the dropout probability p_d in a normalized range as follows

$$p_d = \mathrm{IC}(v_a) \tag{4}$$

where $\mathrm{IC}(\cdot)$ is an increasing function of v_a for mapping the attention of more salient features to higher dropout probabilities. Fig. 5 shows the motivation of the employed positive correlation between p_d and v_a, i.e. the network is enabled to explore more diverse attentions in SndMS regions. In this work, $\mathrm{IC}(\cdot)$ is set as the affine transformation as follows

$$\mathrm{IC}(v_a) = \gamma \cdot v_a + \beta \tag{5}$$

where $\gamma = 0.6$ and $\beta = 0.2$, i.e. the dropout probability falls in the region of $[0.2, 0.8]$. In this way, the most salient features are still retained but suppressed with a relatively larger dropout probability.

Based on the updated dropout probabilities p_d, a dropout mask m_d is generated with independently and identically distributed sampling, i.e. each unit of m_d obeys the Bernoulli distribution as follows

$$m_d \sim \mathrm{Bernoulli}(1 - p_d) \tag{6}$$

For attention-based dropout, the Hadamard product of three vectors, i.e. $f^{(t+1)}$, attention vector v_a in Eq. (1), and dropout mask m_d in Eq. (6) yields a new fusion feature $f'^{(t+1)}$ as:

$$f'^{(t+1)} = f^{(t+1)} * v_a * m_d \tag{7}$$

where $*$ denotes the Hadamard product, $f'^{(t+1)}$ is further used to infer the classification probabilities and yield the Softmax loss for network training.

The proposed attention-based dropout can suppress the most salient features with characteristics-adaptivity probabilities to produce hierarchically-salient features, and thus it differs from the iterative ADL (attention and dropout layer) [10] that discards these salient features deterministically. Fig. 5 presents the difference between the feature maps generated with non-dropout, original dropout, ADL, and our dropout, it shows that our dropout can locate broader and interconnected attention regions, i.e. hierarchically-salient features, which enable the network to better discriminate objects with fine-grained differences.

Consequently, the proposed characteristics attention-based dropout enables networks to explore the attention units with hierarchical-salience levels, and learn features with better robustness performance.

3.6. Network training and inference

For the forward propagation of ACAD, the Softmax loss is calculated as:

$$\mathcal{L} \equiv \mathcal{L}(x, y) = -log(\frac{e^{z_y^{(t+1)}}}{\sum_j e^{z_j^{(t+1)}}}) = -log(\frac{e^{W_y^{(t)} f'^{(t+1)}}}{\sum_j e^{W_j^{(t)} f'^{(t+1)}}}) \tag{8}$$

where y is the ground-truth label of the sample x, $f'^{(t+1)}$ is the output feature vector of attention-based dropout in Eq. (7), and the bias term is omitted for brevity.

For the back-propagation of the ACAD, the regularization weights of different characteristics, i.e. $\{\alpha_1, \alpha_2, \alpha_3\}$ in Eq. (3) are updated based on the derivations of $\mathcal{L}$ w.r.t. these characteristics, where chain rule based on the correlation between $\{\alpha_1, \alpha_2, \alpha_3\}$ and c_1 in Eq. (3) is employed as follows

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \alpha_1} = \frac{\partial \mathcal{L}}{\partial v_a} \frac{\partial v_a}{\partial c_1} f^{(t+1)}, \\ \frac{\partial \mathcal{L}}{\partial \alpha_2} = \frac{\partial \mathcal{L}}{\partial v_a} \frac{\partial v_a}{\partial c_1} h^{1\times1}(W^{(t)})^{\mathrm{T}}, \\ \frac{\partial \mathcal{L}}{\partial \alpha_3} = \frac{\partial \mathcal{L}}{\partial v_a} \frac{\partial v_a}{\partial c_1} h^{1\times1}(\frac{\partial \mathcal{L}}{\partial W}^{(t)})^{\mathrm{T}} \end{cases} \tag{9}$$

where the derivative $\frac{\partial v_a}{\partial c_1}$ is easily derived based on Eqs. (1) and (2). In order to calculate $\frac{\partial \mathcal{L}}{\partial v_a}$, the derivative of $\mathcal{L}$ w.r.t. the intermediate variable of $f'^{(t+1)}$ in Eq. (7) is used, i.e. $\frac{\partial \mathcal{L}}{\partial v_a}$ is formulated as $\frac{\partial \mathcal{L}}{\partial f'^{(t+1)}} * f^{(t+1)} * m_d$, as back-propagation of m_d is not required.

In the inference stage, only the attention module before the FC layer is applied, while the dropout module is omitted. For the attention module, only f and W are used to calculate the attention vector v_a, since $\frac{\partial \mathcal{L}}{\partial W}$ is unavailable at the inference stage. For a better understanding of our hierarchically-salient features, the diffusion model is used to explain their generation in the supplementary material.

4. Experimental results

4.1. Database and experimental setting

We test our algorithm using a four-kernel Nvidia TITAN GPU Card and Pytorch platform. Seven publicly employed object databases, i.e. FM (FashionMNIST) [25], C10 (CIFAR10) [26], C100 (CIFAR100) [26], CUB (CUB-200-2011) [27], Cars (Stanford-Cars) [28], Aircraft (FGVC-Aircraft) [29], and ImgN-1K (ImageNet-1K) [30] are used to evaluate the performance.

We use ResNet18 as the backbone in our main experiment. For fine-grained datasets, i.e. CUB, Cars, and Aircraft, we resized images to

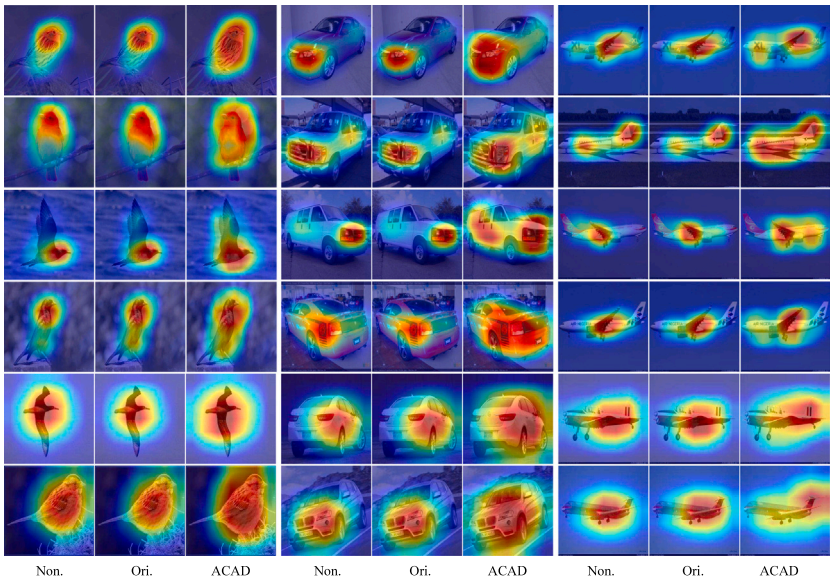


Fig. 6. Grad-CAM (Gradient-weighted Class Activation Mapping) of 'Non.', 'Ori.' and 'ACAD' dropout on samples of CUB-200-2011, Stanford-Cars, and FGVC-Aircraft.

512 × 512 and randomly cropped them with the size of 448 × 448 for data augmentation. The models are pre-trained on the ImageNet-1K dataset [30] and fine-tuned for 200 epochs. The learning rate is decreased by a factor of 0.1 for every 80 epochs, and set to 0.001 for the pre-trained weights or 0.01 for new parameters.

4.2. Algorithm analysis

4.2.1. Toy experiment of quantitative analysis for hierarchically-salient features

To unveil the fundamental contribution of the SndMS (second most salient) features (or less salient features) to the network robustness in an intuitive way, a toy experiment was conducted to evaluate the accuracy, and robustness of the most salient or SndMS features against adversarial noises or corruption.

In this toy experiment, we split each image in the training set of CUB-200-2011 into different regions (including the most salient region, SndMS region, and non-salient region), according to its Grad-CAM. These Grad-CAMs indicate how much different parts of the object contribute to its classification. The warmer the color is, the more salient the region is. The experimental setting is presented in Fig. 7. As shown in Fig. 7(c), to show the importance of the SndMS regions, we erase the most salient regions and fill these blanks with random patterns, and enforce the model training to focus more on the SndMS regions. For comparison, we erase the SndMS regions and fill these blanks with random patterns in Fig. 7(b). We also evaluate these models' classification accuracy, robustness against FGSM, and corruption (Gaussian noise) in Table 1.

Table 1 shows that training with the most salient regions could get better classification performance, whereas training with the SndMS regions has better robustness. We can conclude that the SndMS regions are crucial for robust classification, as they provide complementary object cues to the most salient regions that are perturbed with adversarial attack or corruption.

4.2.2. Visualization of feature maps

To give insight into the feature maps generated by the proposed algorithm, the feature maps of the last convolution layer generated with the dropout variants of 'Non.', 'Ori.' and 'ACAD' are shown in Fig. 6.

As shown in Fig. 6, the proposed attention-based dropout ACAD can explore broader salient regions with larger latent semantic variations compared with 'Non.' and 'Ori.'. Take the image in the 2nd row and 6th column for example, the large activation responses on the headlamp and the side window show the most salient and the SndMS regions, respectively. In this way, ACAD is able to locate attention regions that are more hierarchical, thus enabling the network to better encode object details with fine-grained features.

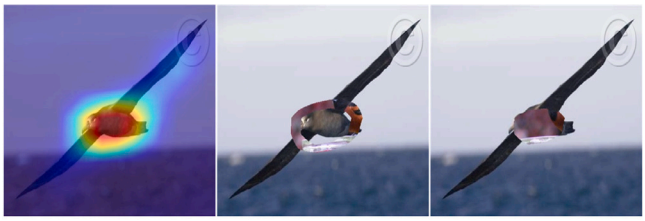


Fig. 7. (a) Grad-CAM and the image generated by replacing (b) the second most salient region and (c) the most salient region, with random patterns.

Table 1
The classification accuracy (Acc. %) and robustness against RobAdv (adversarial perturbations, %) and RobCor (corruptions, %) with the Complete (complete image), the contribution of the MostSalient (most salient) regions, i.e. the SndMS (second most salient) regions are refilled, and the contribution of SndMS regions, i.e. the most salient regions are refilled.

Metric	Complete	MostSalient	SndMS
Acc.	85.57	77.65	75.92
RobAdv	10.58	8.72	8.91
RobCor	9.84	8.31	8.66

4.2.3. Characteristic evolution

To study the variation of regularization weights of the employed characteristics for different databases, the evolution of the coefficients $\{\alpha_i\}$ in Eq. (3) during the training of four datasets are demonstrated in Fig. 8.

Fig. 8 shows that the weights of different characteristics varied dynamically with the iteration epochs, which tend to approach constants as the learning rate decays. Meanwhile, the contributions of the weight derivatives are decreasing for Fashion-MNIST and CIFAR10, while increasing for CIFAR100, due to the reason that the training of CIFAR100 converges relatively slower. From another aspect, while

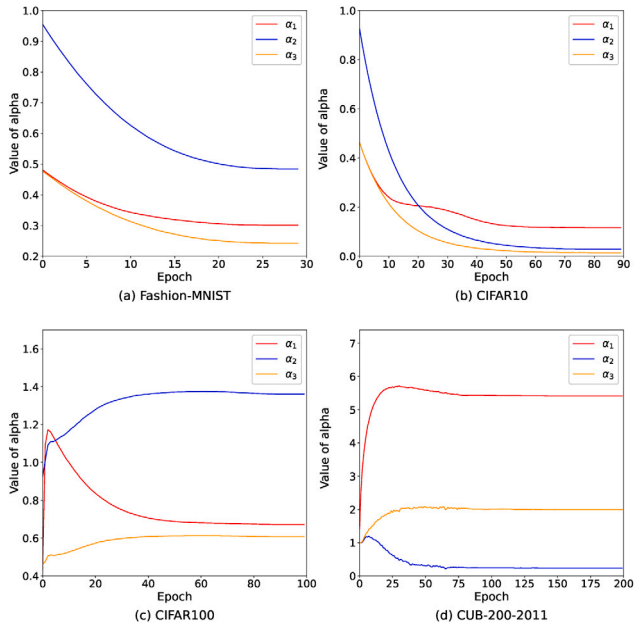


Fig. 8. The evolutions of weights $\{\alpha_1, \alpha_2, \alpha_3\}$ in Eq. (3) for Fashion-MNIST, CIFAR10, CIFAR100 and CUB-200-2011.

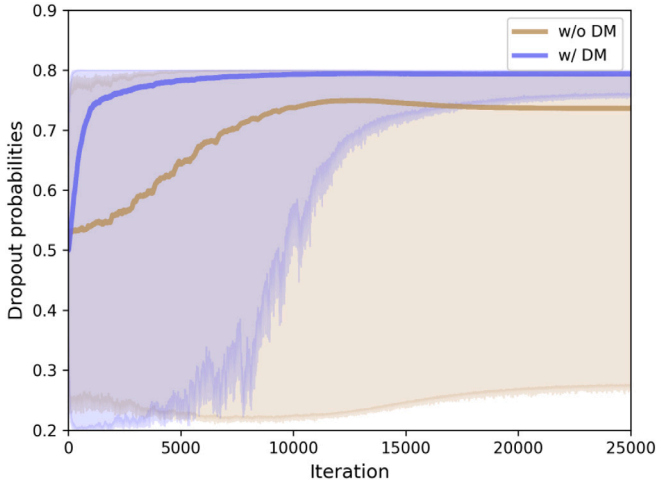


Fig. 9. Evolutions of dropout probabilities p_d in Eq. (4) with (w/) DM (Dropout Module) and without (w/o) DM during training on Fashion-MNIST. Each vertical slice shows a histogram of p_d. The solid curve indicates the mean of p_d and the shade shows the distribution of p_d.

the weight dropout of W outperforms the feature dropout of f for Fashion-MNIST and CIFAR100, the feature dropout performs better for CIFAR10. As shown in Fig. 1, the features reflect the salient neurons, and the weights reflect the correlation between the feature neurons and the categories. When object features show large granularity, the exploration of salient neurons is preferred. The correlation between features and categories performs better for fine-grained objects.

4.2.4. Dropout probability evolution

Besides of characteristic evolution, we also study the variation of dynamic dropout probabilities by showing their evolution in Fig. 9. Fig. 9 shows that the distribution of dropout probabilities by DM converges to a range between 0.7 and 0.8 as the training progresses. By contrast, the probability distribution via the method without DM still has a wide range between 0.25 and 0.8 at the end of training. This observation demonstrates that the proposed DM helps activate unattended neurons with low dropout probabilities, encouraging the network to take into account the contributions of more neurons.

Table 2
The computational complexities of different models.

Methods	FLOPs	Parameters	Training time
Non-dropout (ResNet-18)	502.87M	11.437M	1.50 h
Guided$_{(2019)}$ [12]	502.87M	11.439M	2.60 h
ADL$_{(2019)}$ [10]	502.87M	11.437M	3.13 h
Disout$_{(2020)}$ [31]	502.87M	11.436M	2.13 h
R-Drop$_{(2021)}$ [32]	502.87M	11.437M	1.80 h
DropChanBlock$_{(2021)}$ [33]	502.91M	11.437M	1.58 h
Focused-Drop$_{(2022)}$ [34]	502.99M	11.437M	3.15 h
Fre-Drop$_{(2022)}$ [35]	502.87M	11.463M	2.53 h
Late-Drop$_{(2023)}$ [36]	502.87M	11.437M	1.55 h
ACAD (ours)	502.89M	11.447M	1.88 h

4.2.5. Computational complexity analysis

To study the computational complexity and the training overhead of the proposed algorithm, we show the FLOPs (Floating-point Operations), the number of parameters and the training time of different models in Table 2.

Table 2 shows that our model does not introduce obvious increases in FLOPs and parameters compared with other related models, while this additional overhead primarily arises from the computation of attention scores for activation units. Regarding the training time, our algorithm requires much less training overhead than Guided Dropout, ADL, Focused-Drop and Fre-Drop.

4.2.6. Recognition and generalization performance

To evaluate the generalization recognition performance over the testing dataset, the proposed algorithm is compared to other related variants and several state-of-the-art algorithms (run by ourselves with the same learning rate strategy as our method), including the dropout variants with different clustering variables [23] and the attention variants on a single characteristic. We demonstrate the comparison of the performances in Table 3, and the performances of the 'Non.', 'Ori.' dropout and the proposed ACAD on Fashion-MNIST, CIFAR10, CIFAR100 in Table 3. For a fair comparison, the same random seeds are employed.

Table 3 shows that ACAD outperforms the original dropout by the margins of 0.18% on FashionMNIST, 0.23% on CIFAR10, and 1.21% on CIFAR100. Compared with other variants and state-of-the-art dropout algorithms, our method also shows competitive performances on these three datasets.

4.3. Robustness performance

4.3.1. Robustness metric

To quantitatively evaluate the robustness of the proposed algorithm, the metric introduced in [38] is employed:

$$\begin{cases} \rho_{adv}(F) = E_{\mu}[\Delta_{adv}(x, F)] \\ \Delta_{adv}(x, F) = argmin_{\delta x}\{\|\delta x\|_{\infty} : F(x + \delta x) \neq F(x)\}. \end{cases} \tag{10}$$

where E_{μ} is the expectation w.r.t. the sample distribution, $F(x)$ outputs the predicted label of the sample x, δx denotes the perturbation direction of an adversarial attack. The robustness performances of the related variants against FGSM [22] on FM and C100 are shown in Table 4.

Table 4 shows that the proposed ACAD achieves the largest maximum-allowed perturbation intensity, i.e. ρ_{adv}, for both FM and C100, among eight related variants. Since the proposed attention-based dropout is based on characteristics adaptivity, it enables the network to explore more hierarchically-salient features to alleviate the attack, and hence improve robustness over the related variants.

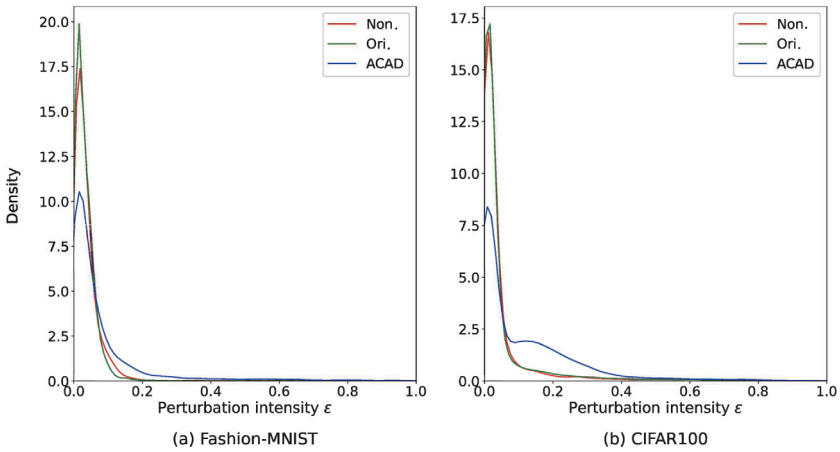


Fig. 10. Distributions of maximum-allowed perturbation intensities (ϵ) by 'Non.', 'Ori.' dropout and ACAD trained on Fashion-MNIST and CIFAR100.

Table 3
The average accuracies (%) and their standard deviations (%) of several related variants, state-of-the-art dropout algorithms, and the proposed ACAD on FM, C10, and C100 with ResNet18. The best performance is marked in bold.

Method	FM	C10	C100
Non-dropout	94.05 ± 0.09	94.56 ± 0.14	76.02 ± 0.35
Original$_{(2012)}$ [9]	94.14 ± 0.04	94.55 ± 0.03	76.20 ± 0.27
Spatial$_{(2015)}$ [17]	94.09 ± 0.15	94.55 ± 0.12	76.25 ± 0.21
Biased$_{(2018)}$ [18]	94.01 ± 0.09	94.50 ± 0.03	76.12 ± 0.09
Crossmap$_{(2018)}$ [18]	94.02 ± 0.17	94.60 ± 0.15	76.00 ± 0.20
Cluster – $f_{(2019)}$ [23]	94.13 ± 0.15	94.56 ± 0.15	76.30 ± 0.34
Cluster – $W_{(2019)}$ [23]	94.14 ± 0.12	94.66 ± 0.17	76.87 ± 0.18
Cluster – $\frac{\partial \mathcal{L}}{\partial x}{}_{(2019)}$ [23]	94.12 ± 0.12	94.54 ± 0.22	76.68 ± 0.30
Cluster – $\frac{\partial \mathcal{L}}{\partial W}{}_{(2019)}$ [23]	94.26 ± 0.15	94.50 ± 0.19	76.66 ± 0.15
Cluster – $f + \frac{\partial \mathcal{L}}{\partial x}{}_{(2019)}$ [23]	94.22 ± 0.25	94.55 ± 0.13	76.34 ± 0.38
Cluster – $f + W_{(2019)}$ [23]	94.13 ± 0.11	94.58 ± 0.14	76.33 ± 0.30
Guided$_{(2019)}$ [12]	94.22 ± 0.13	94.71 ± 0.10	75.73 ± 0.29
ADL$_{(2019)}$ [10]	94.06 ± 0.19	94.22 ± 0.13	74.01 ± 0.34
Disout$_{(2020)}$ [31]	94.15 ± 0.13	94.66 ± 0.10	76.28 ± 0.29
FDD-2D$_{(2020)}$ [37]	**94.37** ± 0.05	94.73 ± 0.02	76.80 ± 0.11
R-Drop$_{(2021)}$ [32]	94.08 ± 0.12	93.84 ± 0.72	73.29 ± 0.64
DropChanBlock$_{(2021)}$ [33]	94.15 ± 0.09	94.32 ± 0.11	72.54 ± 0.40
Focused-Drop$_{(2022)}$ [34]	92.94 ± 0.14	93.54 ± 0.15	74.60 ± 0.14
Fre-Drop$_{(2022)}$ [35]	94.33 ± 0.12	94.57 ± 0.20	75.02 ± 0.27
Late-Drop$_{(2023)}$ [36]	94.28 ± 0.12	94.76 ± 0.14	75.99 ± 0.22
ACAD	94.32 ± 0.05	**94.78** ± 0.12	**77.41** ± 0.31

To study the distributions of maximum-allowed perturbation intensities, the densities of these intensities based on 'Non.', 'Ori.' and 'ACAD' for Fashion-MNIST and CIFAR100 are shown in Fig. 10.

Fig. 10 shows that the maximum-allowed perturbation intensities of the proposed ACAD distribute more on large values than 'Non.' and 'Ori.' for both Fashion-MNIST and CIFAR100, which illustrates the robustness of the proposed algorithm in terms of the quantitative metric in Eq. (10).

4.3.2. Adversarial robustness

We further evaluate the robustness of the proposed ACAD against widely used adversarial attacks. To make the comparison convincing and comprehensive, both white and black box attacks are employed for the evaluation. More precisely, four adversarial attack algorithms, i.e. FGSM [22], PGD [39], JSMA [40], Newtonfool [41], are employed for the testing. For robustness evaluation against two additional latest attack algorithms, please refer to the supplementary material.

PGD [39] performed the iterative attack via projecting the perturbed samples into the feasible region. JSMA (Jacobian Saliency Map) [40] used the Jacobian matrix to derive the saliency map from input to output, so as to attack the output structure by perturbing only a small part of the input features. Newtonfool [41] is an untargeted attacker that tries to decrease the largest predicted probability by gradient descent, where the step size is determined adaptively. The adversarial robustness toolbox[1] is employed to implement these attack algorithms.

Recognition accuracies (%) against the FGSM attack with increasing normalized perturbation intensities on Fashion-MNIST and CIFAR100 are shown in Fig. 11. The robustness performances of the proposed algorithm and other related dropout variants against the four adversarial attacks are shown in Table 5. Robustness (%) of 'Non.', 'Ori.' dropout and the proposed ACAD against FGSM and PGD for CUB, Cars, Aircraft, and ImgN-1K are demonstrated in Table 6.

Fig. 11 shows that the proposed ACAD achieves the best robustness against FGSM attack with varying perturbation intensities.

Table 5 further justifies that ACAD outperforms its variants and state-of-the-art dropout algorithms in terms of robustness. For the Fashion-MNIST dataset, ACAD outperforms the original dropout [9] by the margins of 14.79% and 13.80% when $\epsilon = 0.03$ and $\epsilon = 0.12$. For the CIFAR100 dataset, ACAD greatly outperforms the original dropout by the margins of 37.09% and 11.58% when $\epsilon = 0.03$ and $\epsilon = 0.25$.

Table 6 shows that the robustness improved by the proposed ACAD is impressive on fine-grained databases and ImageNet-1K, where ACAD outperforms the original dropout by 16.63%, 36.04%, 40.00%, and 25.59% for CUB, Cars, Aircraft, and ImageNet-1K, respectively. These results show that the proposed characteristics-adaptivity dropout can largely enhance the network adversarial robustness, due to the better exploration of hierarchically-salient features.

4.4. Ablation study

To investigate the performance of each characteristic or module on the proposed ACAD, we provide an ablation study on FM (Fashion-MNIST) and C100 (CIFAR100) in terms of recognition accuracy and robustness in Table 7, where ResNet-18 is used as the baseline.

As shown in Table 7, AM (Attention Module) based on a single characteristic, i.e. f or W, could improve the baseline performance, which indicates that instance-specific information (represented by f) and dataset-common information (represented by W) both contribute to classification. Meanwhile, f and W turn out to be complementary in terms of generalization and robustness, since the variant with $f + W + DM$ outperforms both variants with $f + DM$ and $W + DM$. Since the $\frac{\partial \mathcal{L}}{\partial W}$ is not as stable as W, mere $\frac{\partial \mathcal{L}}{\partial W}$ is unable to bring improvements. However, on the basis of f and W, the characteristic $\frac{\partial \mathcal{L}}{\partial W}$ further brings a performance gain.

[1] https://github.com/Trusted-AI/adversarial-robustness-toolbox

Table 4
Robustness comparison of related dropout variants on FM and C100 in terms of the metric in Eq. (10) with FGSM.

Dataset	Non.	Ori. [9]	Spatial [17]	Biased [18]	Crossmap [18]	Cluster-f [23]	Focused-Drop [34]	Late-Drop [36]	ACAD
FM	0.040	0.034	0.041	0.038	0.046	0.053	0.034	0.038	**0.072**
C100	0.048	0.047	0.050	0.044	0.040	0.083	0.052	0.045	**0.110**

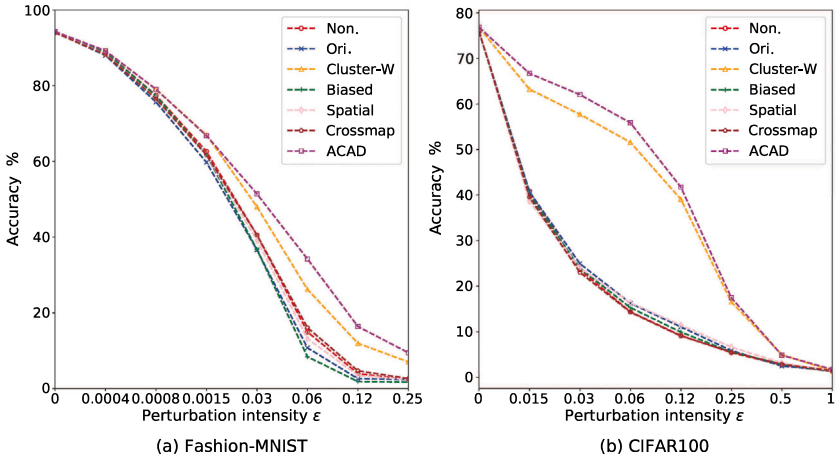


Fig. 11. Robustness accuracy (%) of different dropout variants against FGSM with increasing ϵ in L_∞ norm for Fashion-MNIST and CIFAR100.

Table 5
Robustness (%) against various non-targeted adversarial attacks on FM, C10, and C100. The perturbation intensity of FGSM is 0.03 in L_∞ norm. The perturbation intensities of PGD on FM , C10, and C100 are 0.03, 8 (out of 255), and 8 (out of 255) in L_∞ norm, respectively. The intensity step is 10. The maximum distortion of JSMA is set as 0.1. †means that the improvement is significant under the significance level of 0.05 with t-testing.

Method	FGSM			PGD			JSMA			NewtonFool		
	FM	C10	C100	FM	C10	C100	FM	C10	C100	FM	C10	C100
Non.	40.55	58.28	23.60	10.81	41.03	7.86	13.07	2.17	18.89	32.48	18.01	1.42
Ori.$_{(2012)}$ [9]	36.64	59.66	24.95	7.71	41.75	8.50	12.20	2.54	15.30	27.37	19.03	1.17
Spatial$_{(2015)}$ [17]	39.17	55.94	24.03	8.37	37.37	7.79	13.67	2.48	13.30	30.20	12.31	1.18
Crossmap$_{(2018)}$ [18]	40.59	58.31	23.03	8.23	43.57	7.50	11.96	1.72	12.97	33.30	**27.41**	1.49
Biased$_{(2018)}$ [18]	36.75	58.79	23.95	8.25	41.15	8.26	12.22	2.98	16.54	29.45	17.48	1.27
Cluster-$f_{(2019)}$ [23]	41.59	52.60	41.35	10.88	37.07	21.54	14.29	2.91	15.55	30.67	16.16	1.65
Cluster-$W_{(2019)}$ [23]	48.03	57.94	57.72	19.75	41.42	35.60	14.86	3.38	20.21	32.43	16.03	2.41
Cluster-$\frac{\partial \mathcal{L}}{\partial x}$ $_{(2019)}$ [23]	48.67	52.27	44.46	21.02	36.45	23.71	13.95	3.07	17.24	33.99	17.59	1.81
Cluster-$\frac{\partial \mathcal{L}}{\partial W}$ $_{(2019)}$ [23]	48.40	54.40	57.09	21.54	38.79	35.12	14.41	3.79	16.83	35.02	17.08	2.46
Cluster-$f + \frac{\partial \mathcal{L}}{\partial x}$ $_{(2019)}$ [23]	44.27	51.84	60.31	17.10	36.13	17.63	14.49	3.13	11.06	33.82	16.54	1.56
Cluster-$f + W_{(2019)}$ [23]	43.11	51.78	47.62	13.12	37.45	25.92	13.23	2.80	16.83	31.49	16.10	1.84
ADL$_{(2019)}$ [10]	44.90	48.18	27.64	13.33	32.92	10.57	12.62	1.85	20.26	36.90	3.22	1.64
Disout$_{(2020)}$ [31]	29.91	59.57	20.90	6.72	41.85	5.15	11.80	2.21	20.59	29.33	15.73	1.02
Guided$_{(2019)}$ [12]	47.98	**62.79**	25.04	21.98	45.04	8.89	14.27	2.57	17.38	37.83	17.17	1.37
FDD-2D$_{(2020)}$ [37]	50.87	–	60.04	21.39	–	39.44	–	–	-	–	–	–
R-Drop$_{(2021)}$ [32]	33.80	61.56	27.66	10.93	**48.54**	12.07	11.67	**8.49**	**30.21**	27.43	15.83	2.78
DropChanBlock$_{(2021)}$ [33]	29.94	57.91	27.31	5.10	46.12	16.87	9.30	1.46	16.54	27.35	17.30	2.69
Focused-Drop$_{(2022)}$ [34]	42.63	61.03	24.25	12.19	42.68	7.84	14.96	2.24	18.47	31.91	19.93	2.46
Fre-Drop$_{(2022)}$ [35]	41.08	56.82	22.57	17.69	44.76	9.55	14.37	0.61	13.51	37.85	21.03	1.58
Late-Drop$_{(2023)}$ [36]	39.58	62.62	29.71	10.30	47.51	12.73	14.69	2.84	19.47	34.42	26.86	2.85
ACAD-f	50.15	58.04	61.41†	23.86†	42.58	42.27†	**16.96**†	5.04†	19.23	38.46†	18.50	3.06†
ACAD-w	48.10	57.87	61.49†	15.10	42.83	42.90†	13.36	5.11†	20.25	31.45	18.10	2.34
ACAD-$f + W$	51.14†	58.04	**62.05**†	23.60†	43.48	43.42†	16.23†	5.13†	19.76	38.89†	19.22	2.95†
ACAD-$f + W + \frac{\partial \mathcal{L}}{\partial W}$	**51.43**†	59.83	62.04†	**25.38**†	41.72	**43.48**†	16.23†	5.20†	20.48	**39.50**†	19.11	**3.27**†

Since the proposed ACAD without DM is actually an attention model, we compare this variant with the baseline to test the performance of AM. Table 7 shows that there are gains of accuracy over the baseline on both two datasets, this is because AM enables the network to dynamically explore the salient features based on characteristics-adaptivity. However, overdependence on salient features in recognition reduces the robustness of the network against noise or corruption. This analysis is supported as AM seems to be helpless for robustness improvement.

Compared to the network trained with sole AM, the additional DM could further improve network robustness on the basis of retaining recognition performance. For example, the last two rows of Table 7 indicate that DM achieves an improvement of accuracy by the margin of 1.05% on CIFAR100, and the last two rows indicate that DM achieves improvements of 11.62% and 38.45% in terms of robustness on Fashion-MNIST and CIFAR100, respectively. These appealing results in terms of both accuracy and robustness performances indicate the

Table 6
Robustness performances (%) of 'Non.', 'Ori.' dropout and the proposed ACAD against FGSM and PGD for CUB, Cars, Aircraft, and ImgN-1K. The perturbation intensities of FGSM and PGD on ResNet18 for CUB, Cars, and Aircraft are 0.03 and 2 (out of 255) in L_∞ norm. The perturbation intensity of FGSM and PGD on ResNet152 for ImgN-1K are 0.03 and 8 (out of 255) in L_∞ norm. The intensity step is 10.

	FGSM				PGD			
	CUB	Cars	Aircraft	ImgN-1K	CUB	Cars	Aircraft	ImgN-1K
Non.	0.77	0.33	0.55	14.15	3.00	7.55	3.15	0.62
Ori.	1.41	1.61	1.84	18.50	5.47	17.20	11.38	1.33
ACAD	**18.04**†	**37.65**†	**41.84**†	**44.09**†	**12.64**†	**46.77**†	**47.52**†	**11.48**†

Table 7
Ablation study of ACAD in terms of recognition and robustness performances on FM and C100. DM is the abbreviation of the dropout module based on different combinations of characteristics in Eq. (3). The best and second best are labeled with bold and underline, respectively.

AM			DM	Accuracy		Robustness	
f	W	$\frac{\partial \mathcal{L}}{\partial W}$		FM	C100	FM	C100
×	×	×	×	94.05	76.02	40.55	23.60
✓	×	×	×	94.14	76.37	40.28	23.51
✓	×	×	✓	94.31	77.16	50.15	61.41
×	✓	×	×	94.19	76.13	39.61	23.80
×	✓	×	✓	94.17	77.22	48.10	61.49
✓	✓	×	×	94.23	76.12	38.91	23.91
✓	✓	×	✓	**94.34**	<u>77.33</u>	<u>51.14</u>	**62.05**
✓	×	✓	×	94.27	76.05	37.82	23.11
✓	×	✓	✓	94.33	76.98	45.06	61.37
✓	✓	✓	×	94.23	76.36	39.81	23.59
✓	✓	✓	✓	<u>94.32</u>	**77.41**	**51.43**	<u>62.04</u>

effectiveness of hierarchically-salient features explored by the proposed DM.

5. Discussion and conclusion

During the learning of robust feature representation, network characteristics, i.e. FC features, inter-layer weights, and their derivatives, perform diversely on different databases. Meanwhile, hierarchically-salient features are not fully explored in the discrimination of fine-grained object differences. To this end, this work proposes an attention module for dynamic adaption of characteristics contributions during training, as well as an attention-based dropout to explore hierarchical-salience levels of features. Extensive results on four general and three fine-grained object recognition problems show that our algorithm can largely improve the network robustness, without compromising its generalization performance, compared with the related variants and state-of-the-art algorithms.

Although competitive robustness is achieved by our algorithm, there is still room for further improvement. First, more characteristics on network's lower layers should be investigated for adaptivity. Second, since additional runtime overhead primarily arises from the computation of attention scores for activation units, we will develop more efficient methods for estimating attention scores to reduce this overhead. Third, while our approach is specifically designed to enhance network robustness, it achieves only marginal improvements in classification performance, which demands further exploration. Lastly, our algorithm is general and shall be explored for other tasks, like object localization or segmentation.

CRediT authorship contribution statement

Weicheng Xie: Conceptualization, Formal analysis, Methodology, Writing – review & editing, Investigation, Supervision, Validation, Writing – original draft, Funding acquisition. **Cheng Luo:** Data curation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing. **Gui Wang:** Data curation, Validation, Visualization, Writing – review & editing. **Linlin Shen:** Conceptualization, Formal analysis, Funding acquisition, Methodology, Supervision, Writing – review & editing. **Zhihui Lai:** Formal analysis, Methodology, Writing – review & editing. **Siyang Song:** Formal analysis, Methodology, Writing – review & editing.

Declaration of competing interest

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.

Data availability

Algorithm codes will be public available upon acceptance.

Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful comments and constructive suggestions. We are also very grateful to our colleague Basker George for proofreading the linguistics of this work. The work was supported by the National Natural Science Foundation of China under grants no. 62276170, 82261138629, the Science and Technology Project of Guangdong Province, China under grants no. 2023A1515011549, 2023A1515010688, the Science and Technology Innovation Commission of Shenzhen, China under grant no. JCYJ20220531101412030.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.patcog.2023.110240.

References

[1] H. Qi, M. Brown, D.G. Lowe, Low-shot learning with imprinted weights, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 5822–5830.
[2] A. Despande, S. Kamath, K. Subrahmanyam, Better generalization with adaptive adversarial training, in: Proc. Int. Conf. Mach. Learn. Workshop, 2019.
[3] Z. Wang, H. Guo, Z. Zhang, W. Liu, Z. Qin, K. Ren, Feature importance-aware transferable adversarial attacks, in: Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 7639–7648.
[4] X. Li, W. Wang, X. Hu, J. Yang, Selective kernel networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 510–519.
[5] X. Wang, F. Wu, J. Wang, Self-adaptive embedding for few-shot classification by hierarchical attention, in: Proc. IEEE Int. Conf. Multimedia Expo, Jul. 2020, pp. 1–6.
[6] S. Chen, Z. He, C. Sun, J. Yang, X. Huang, Universal adversarial attack on attention and the resulting dataset DAmageNet, IEEE Trans. Pattern Anal. Mach. Intell. 44 (4) (2022) 2188–2197.
[7] Y. Liu, Z. Lu, J. Li, T. Yang, Hierarchically learned view-invariant representations for cross-view action recognition, IEEE Trans. Circuits Syst. Video Technol. 29 (8) (2018) 2416–2430.
[8] J. Xie, C. Luo, X. Zhu, Z. Jin, W. Lu, L. Shen, Online refinement of low-level feature based activation map for weakly supervised object localization, in: Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 132–141.

[9] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, 2012, arXiv:1207.0580.
[10] J. Choe, H. Shim, Attention-based dropout layer for weakly supervised object localization, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2214–2223.
[11] J. Choe, D. Han, S. Yun, J.-W. Ha, S.J. Oh, H. Shim, Region-based dropout with attention prior for weakly supervised object localization, Pattern Recognit. 116 (2021) 107949.
[12] R. Keshari, R. Singh, M. Vatsa, Guided dropout, in: Proc. AAAI Conf. Artif. Intell., Vol. 33, 2019, pp. 4065–4072.
[13] R. Hou, H. Chang, B. Ma, S. Shan, X. Chen, Cross attention network for few-shot classification, in: Proc. Conf. Neural Inf. Process. Syst., 2019, pp. 4003–4014.
[14] H. Zheng, J. Fu, T. Mei, J. Luo, Learning multi-attention convolutional neural network for fine-grained image recognition, in: Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5219–5227.
[15] C. Shen, G. Qi, R. Jiang, Z. Jin, H. Yong, Y. Chen, X. Hua, Sharp attention network via adaptive sampling for person re-identification, IEEE Trans. Circuits Syst. Video Technol. 29 (10) (2019) 3016–3027.
[16] M. Tan, Z. Hu, B. Wang, J. Zhao, Y. Wang, Robust object recognition via weakly supervised metric and template learning, Neurocomputing 181 (2016) 96–107.
[17] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, C. Bregler, Efficient object localization using convolutional networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 648–656.
[18] A. Poernomo, D. Kang, Biased dropout and crossmap dropout: Learning towards effective dropout regularization in convolutional neural network, Neural Netw. 104 (2018) 60–67.
[19] D. Stutz, M. Hein, B. Schiele, Disentangling adversarial robustness and generalization, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 6969–6980.
[20] T. Pang, K. Xu, C. Du, N. Chen, J. Zhu, Improving adversarial robustness via promoting ensemble diversity, in: Proc. Int. Conf. Mach. Learn., Vol. 97, 2019, pp. 4970–4979.
[21] Q. Wang, T. Wu, H. Zheng, G. Guo, Hierarchical pyramid diverse attention networks for face recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8323–8332.
[22] I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: Proc. Int. Conf. Learn. Represent., 2015.
[23] Z. Wen, Z. Ke, W. Xie, L. Shen, Clustering-based adaptive dropout for CNN-based classification, in: Proc. IAPR Asian Conf. Pattern Recognit., 2019, pp. 46–58.
[24] D. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Proc. Int. Conf. Learn. Represent., 2015.
[25] H. Xiao, K. Rasul, R. Vollgraf, Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms, 2017, arXiv:1708.07747.
[26] A. Krizhevsky, G. Hinton, Learning Multiple Layers of Features from Tiny Images, Tech. Rep., Dept. Comput. Sci., Univ. Toronto, Canada, 2009.
[27] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The Caltech-UCSD birds-200–2011 dataset, Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.
[28] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3D object representations for fine-grained categorization, in: Proc. IEEE Int. Conf. Comput. Vis. Workshops, 2013, pp. 554–561.
[29] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, A. Vedaldi, Fine-grained visual classification of aircraft, 2013, arXiv:1306.5151.
[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.
[31] Y. Tang, Y. Wang, Y. Xu, B. Shi, C. Xu, C. Xu, C. Xu, Beyond dropout: Feature map distortion to regularize deep neural networks, in: Proc. AAAI Conf. Artif. Intell., 2020, pp. 5964–5971.
[32] L. Wu, J. Li, Y. Wang, Q. Meng, T. Qin, W. Chen, M. Zhang, T.-Y. Liu, et al., R-drop: Regularized dropout for neural networks, Proc. Conf. Neural Inf. Process. Syst. 34 (2021) 10890–10905.
[33] Y. Ding, S. Dong, Y. Tong, Z. Ma, B. Xiao, H. Ling, Channel DropBlock: An improved regularization method for fine-grained visual classification, in: Proc. British Mach. Vis. Conf., 2021, p. 267.
[34] M. Liu, T. Xie, X. Cheng, J. Deng, M. Yang, X. Wang, M. Liu, FocusedDropout for convolutional neural network, Appl. Sci. 12 (15) (2022) 7682.
[35] M. Islam, B. Glocker, Frequency dropout: Feature-level regularization via randomized filtering, in: Proc. Eur. Conf. Comput. Vis. Workshop, 2022, pp. 281–295.
[36] Z. Liu, Z. Xu, J. Jin, Z. Shen, T. Darrell, Dropout reduces underfitting, in: Proc. Int. Conf. Mach. Learn., Vol. 202, 2023, pp. 22233–22248.
[37] Z. Ke, Z. Wen, W. Xie, Y. Wang, L. Shen, Group-wise dynamic dropout based on latent semantic variations, in: Proc. AAAI Conf. Artif. Intell., 2020, pp. 11229–11236.
[38] N. Papernot, P.D. McDaniel, X. Wu, S. Jha, A. Swami, Distillation as a defense to adversarial perturbations against deep neural networks, in: Proc. IEEE Symp. Security Privacy, 2016, pp. 582–597.
[39] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: Proc. Int. Conf. Learn. Represent., 2018.
[40] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z.B. Celik, A. Swami, The limitations of deep learning in adversarial settings, in: Proc. IEEE Eur. Symp. Security Privacy, 2016, pp. 372–387.
[41] U. Jang, X. Wu, S. Jha, Objective metrics and gradient descent algorithms for adversarial examples in machine learning, in: Annu. Comput. Secur. Appl. Conf., 2017, pp. 262–277.

Weicheng Xie is currently an associate professor at School of Computer Science and Software Engineering, Shenzhen University, China. He received the B.S. degree in statistics from Central China Normal University in 2008, the M.S. degree in probability and mathematical statistics and Ph.D. degree in computational mathematics from Wuhan University, China in 2010 and 2013. He has been a visiting research fellow with School of Computer Science, University of Nottingham, UK. His current researches focus on facial expression analysis and robust network design.

Cheng Luo is currently pursuing his master's degree at College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His research interest involves adversarial learning, graph neural network, and video generation. He has published 3 CVPR/ICCV/IJCAI papers.

Gui Wang is currently pursuing her PhD degree at Department of Computer Science, University of Nottingham Ningbo China, Ningbo, China. She is also a Lecturer (Assistant Professor) at Zhejiang Security College. Her research interests include deep learning, facial recognition, and medical image processing.

Linlin Shen (Senior Member, IEEE) received the B.Sc. and M.Eng. degrees from Shanghai Jiaotong University, Shanghai, China, in 1997 and 2000, respectively, and the Ph.D. degree from the University of Nottingham, Nottingham, U.K., in 2005. He was a Research Fellow with the University of Nottingham, working on MRI brain image processing. He is currently a Pengcheng Scholar Distinguished Professor with the School of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. He is also an Honorary Professor with the School of Computer Science, University of Nottingham, and a Distinguished Visiting Scholar with the University of Macao, Macau, China. He serves as the Director of the Computer Vision Institute, AI Research Center for Medical Image Analysis and Diagnosis and China – U.K., a joint Research Laboratory for visual information processing. His research interests include deep learning, facial recognition, analysis/synthesis, and medical image processing. Dr. Shen is listed as the Most Cited Chinese Researchers by Elsevier. He received the Most Cited Paper Award from the Image and Vision Computing journal. His cell classification algorithms were the winners of the International Contest on Pattern Recognition Techniques for Indirect Immunouorescence Images held by International Conference on Image Processing (ICIP) 2013 and International Conference on Pattern Recognition (ICPR) 2016.

Zhihui Lai received the B.S. degree in mathematics from South China Normal University, M.S. degree from Jinan University, and the Ph.D. degree in pattern recognition and intelligence system from Nanjing University of Science and Technology (NUST), China, in 2002, 2007 and 2011, respectively. He has been a Research Associate, Postdoctoral Fellow and Research Fellow at The Hong Kong Polytechnic University. His research interests include face recognition, image processing and content-based image retrieval, pattern recognition, compressive sense, human vision modelization and applications in the fields of intelligent robot research.

Siyang Song is currently a Lecturer (Assistant Professor) at the University of Leicester. He is also an affiliated researcher at the Department of Computer Science and Technology, University of Cambridge. His current research interests include affective computing, graph representation learning, computer vision and machine learning.