Journal of Systems Architecture 170 (2026) 103602

Contents lists available at ScienceDirect

Journal of Systems Architecture

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

FGAA: Enhancing adversarial robustness in AIoT-enabled smart systems via Fine-Grained Activation Alignment☆

Wenxin Kuang [a], Fengxiao Tang [b,*], Jiayang Liu [c], Yupeng Hu [a,e,*], Keqin Li [d]

[a] College of Computer Science and Electronic Engineering, Hunan University, 410082, Changsha, China
[b] School of Computer Science and Engineering, Central South University, 410083, Changsha, China
[c] Institute of Science Tokyo, 152-8550, Tokyo, Japan
[d] State University of New York, New Paltz, 12561, NY, USA
[e] Xiangjiang Laboratory, 410000, Changsha, China

ARTICLE INFO

Keywords:
Artificial intelligence of things (AIoT) security
Model robustness
Adversarial example attack
Adversarial training (AT)

ABSTRACT

In security-critical applications within Artificial Intelligence of Things (AIoT) systems, ensuring robust defense against adversarial attacks is paramount for both security and privacy. Although Adversarial Training (AT) is widely recognized as an effective strategy to enhance network robustness, empirical evidence reveals a significant disparity in feature activation between adversarial and natural examples. Current AT techniques, while partially addressing these discrepancies, generally offer only moderate improvements in robustness—often at the expense of accuracy on natural examples. In this work, we introduce the overlap ratio (OR), a novel metric that quantitatively assesses differences in feature activation between adversarial and natural inputs. Building on this insight, we propose a Fine-Grained Activation Alignment (FGAA) strategy that operates at an individual feature level, effectively reducing activation discrepancies while maintaining high model accuracy. By integrating FGAA with conventional defense methods, our approach minimizes activation differences and significantly bolsters overall model robustness. This enhancement is critical for secure, privacy-preserving AIoT deployment. Extensive experiments on three datasets and multiple models demonstrate that FGAA reduces activation frequency discrepancies by 29.69% and achieves an improvement of up to 39.84% in robustness compared to standard adversarial training alone. Notably, on the SVHN dataset, FGAA achieves a natural accuracy of 94.01%, nearly 10% higher than the 84.10% attained with AT, underscoring its potential to advance the state-of-the-art in deep neural network robustness for secure AIoT-enabled smart societies.

1. Introduction

Deep Neural Networks (DNNs) have become a cornerstone in areas such as image recognition, autonomous driving, healthcare, and finance, often achieving performance levels that surpass human capabilities [1–3]. Despite their success, numerous studies have demonstrated that DNNs are susceptible to adversarial attacks—small, carefully crafted perturbations to input data that can lead to incorrect predictions. This vulnerability is particularly concerning in the context of Artificial Intelligence of Things (AIoT) systems, where security and privacy risks can have severe consequences [4,5]. In AIoT-enabled smart societies, intelligent devices and sensors interact with AI-driven models, making them prime targets for adversarial manipulation. In such environments, adversarial examples can be exploited by attackers to manipulate sensor readings, override control signals, or compromise data integrity, which may result in equipment malfunctions, unauthorized access, cascading failures, significant data breaches, and substantial economic and reputational losses [6–8]. In response to these threats, the research community has developed robust attack strategies that expose the vulnerabilities of DNNs, along with corresponding defense mechanisms designed to mitigate these weaknesses [9–11]. Adversarial Training (AT) is widely recognized as one of the most effective defense techniques [12–14]. However, a significant trade-off exists: while AT improves the robustness of models against adversarial attacks, it often reduces accuracy on natural (unperturbed) examples [15,16]. This trade-off between enhancing adversarial robustness and preserving generalization performance is a major challenge, especially for

☆ This article is part of a Special issue entitled: 'SPASS' published in Journal of Systems Architecture.

* Corresponding authors.
E-mail addresses: wenxinkuang@hnu.edu.cn (W. Kuang), tangfengxiao@csu.edu.cn (F. Tang), ljyljy@nus.edu.sg (J. Liu), yphu@hnu.edu.cn (Y. Hu), lik@newpaltz.edu (K. Li).

https://doi.org/10.1016/j.sysarc.2025.103602
Received 31 March 2025; Received in revised form 5 October 2025; Accepted 13 October 2025
Available online 27 October 2025

security-critical AIoT applications where both security and accuracy are crucial.

1.1. Motivation

In AIoT-enabled smart societies, where vast amounts of sensitive data are generated, processed, and exchanged, the susceptibility of DNNs to adversarial attacks poses a serious threat to security and privacy. Attackers can inject adversarial examples during training or testing phases, leading to abnormal model behavior and potentially compromising the integrity of smart city infrastructures, healthcare monitoring systems, and industrial control networks. Studies have shown that adversarial and natural examples exhibit distinct activation patterns in feature maps. For example, [17–19] focus predominantly on visual assessments of these differences without providing a detailed quantitative analysis. Therefore, it is crucial to rigorously analyze these differences and design defense strategies that optimize both robustness and accuracy in AIoT environments.

1.2. Our work

Building on these insights, we conducted a series of experiments to examine the impact of adversarial and natural examples on network activations, as detailed in Section 3.1. In a typical neural network layer, feature maps are three-dimensional, consisting of height (H), width (W), and depth (C). The depth dimension corresponds to channels, while the height and width define the spatial layout. Based on our analysis across spatial and channel dimensions, we introduce the overlap ratio (OR) to quantify differences between the feature-activation distributions of adversarial and natural examples. Our results show that adversarial examples consistently exhibit higher activation magnitudes compared to natural examples, indicating that certain features are prone to over-activation. Although AT helps narrow the gap in activation magnitudes between adversarial and natural examples, a significant discrepancy remains. Motivated by this observation, we systematically explored a range of alignment strategies — including channel-only, spatial-only, combined spatial-channel, and feature-wise approaches — to further mitigate these differences (see Section 4.4 for details). Our experiments indicate that the Fine-Grained Activation Alignment (FGAA) strategy, which operates at the feature level, is the most effective in enhancing model robustness by minimizing the discrepancies between feature activation distributions. The FGAA module can be seamlessly integrated into existing neural network architectures and combined with adversarial training to achieve optimal robustness.

In the context of AIoT security, perception systems frequently encounter perturbations such as compression artifacts, sensor noise, or environmental interference, which can cause shifts in internal feature activations and degrade decision reliability. The proposed FGAA addresses this issue by more stably aligning task-relevant features, thereby reducing the influence of perturbation-sensitive activations. Moreover, its modular design enables flexible adaptation to the resource budgets of edge devices, making FGAA a practical component for AIoT-enabled smart systems.

1.3. Main contributions

We summarize our contributions as follows:

- **Quantitative Analysis of Feature Activation Differences:** We identify significant differences in the feature activation distributions between adversarial and natural examples across spatial and channel dimensions. To measure these differences, we introduce the OR metric, demonstrating that these discrepancies adversely affect model robustness, particularly as adversarial training does not fully eliminate differences in channel activation.
- **Development of the FGAA Module:** We propose the FGAA module, which aligns features during adversarial training through a three-step process: establishing relationships between feature maps and categories, evaluating feature importance based on map weights, and applying fine-grained alignment guided by these importance measures.
- **Extensive Evaluation Across Models and Datasets:** We integrate the FGAA module with various adversarial training techniques and evaluate its performance on multiple models and datasets. Our results show that FGAA not only improves the alignment of feature activation distributions between adversarial and natural examples but also enhances overall model robustness. Notably, on the SVHN dataset, FGAA reduces the negative impact of adversarial training on natural example accuracy.

The rest of the paper is organized as follows. Section 2 reviews related work on two classes of defense augmentation methods based on adversarial training. In Section 3, we describe our proposed FGAA strategy in detail and provide a formal description of the problem. Section 4 presents a comprehensive experimental analysis and validation of the FGAA module across multiple models and datasets. Finally, Section 5 summarizes the paper and discusses future work.

2. Related work

In this section, we will first discuss the inherent limitations of AT defense methods. We will then provide a brief overview of research focused on improving the effectiveness of existing AT methods, with an emphasis on two key aspects: network structure optimization and feature activation adjustment.

2.1. Network structure optimization

To address the limitations of AT, various strategies have explored optimizing model structures by combining model pruning or structural design with traditional AT methods to enhance robustness. For instance, some approaches utilize model compression as a constraint objective for AT [20], while others combine stochastic activation pruning with AT to defend against adversarial attacks [21]. Additional efforts have focused on designing training schemes based on model compression to address security issues in specific domains, such as recurrent neural networks in natural language processing [22]. Other methods employ single pruning and model fine-tuning, reducing dependence on AT [23], or guide pruning techniques to extract robust sub-networks from large, non-robust models [24]. However, because efficiency optimization is also a core goal of these methods, their focus on efficiency often limits the extent to which their defensive capabilities can be improved.

2.2. Feature activation adjustment

In parallel with network structure optimization, a number of studies have focused on improving model robustness by refining feature activations. These methods explore how adversarial perturbations alter the activation patterns of neural networks and propose corrective mechanisms to bridge the gap between adversarial and natural examples. Bai et al. [18] identified two key characteristics in channel activations when comparing adversarial and natural examples: adversarial examples not only exhibit higher activation magnitudes but also display more uniform activation frequencies. Based on these observations, they introduced the Channel-wise Activation Suppression (CAS) method to mitigate redundant activations and enhance robustness. Building upon CAS, subsequent work proposed the Channel-wise Importance-based Feature Selection (CIFS) [19] strategy, which specifically targets the suppression of negatively correlated channels that are disproportionately amplified in adversarial examples. However, both CAS and

CIFS primarily focus on channel-wise activations, neglecting the spatial dimension of feature maps, and their evaluations are largely qualitative. Consequently, over-activated features induced by adversarial perturbations are not comprehensively addressed, limiting the overall defensive effectiveness. More recently, Wu et al. [25] proposed RSA, a comprehensive defense method that integrates feature refinement, activation suppression, and alignment modules. RSA employs consistency constraints and knowledge distillation to maintain predictive performance on natural examples. Nevertheless, the incorporation of multiple auxiliary components results in a cumbersome and computationally expensive framework. Similarly, the StayFocused framework [26] employs spatial hyper-spherical constraints and channel-adaptive prompt calibration to improve adversarial robustness. While effective, its reliance on multi-head training and modular integration introduces additional complexity and overhead. Furthermore, many of these methods are evaluated only under standard attacks and lack generalization assessments against adaptive or unforeseen adversaries.

2.3. Technical challenges

Although prior works have attempted to improve model robustness through network structure optimization and feature activation adjustment, several technical challenges remain unresolved. First, **coarse-grained alignment** strategies dominate current approaches, which typically operate at the channel level and fail to account for the nuanced spatial patterns critical for distinguishing adversarial from natural inputs. Second, **robustness–accuracy trade-off** remains a persistent issue: improving defense performance often compromises natural accuracy, and balancing this trade-off effectively is non-trivial. Third, **limited adaptability** to diverse and adaptive attack types undermines the generalizability of many methods, which are often tuned for specific threat models. Finally, **structural complexity and poor deployability** hinder practical use in real-world IoT settings—many existing methods rely on deep, multi-branch, or task-specific modules that inflate model size and computational demands.

To address these limitations, we propose the FGAA framework, which enables fine-grained feature alignment across both spatial and channel dimensions in a modular and easily integrable design. FGAA evaluates the contribution of individual features to correct predictions and selectively suppresses those that are overly sensitive to adversarial perturbations, preserving accuracy on natural data. Unlike the CAS method, which applies coarse and uniform structured suppression at the channel level for features vulnerable to adversarial activation, FGAA offers a more fine-grained and unstructured suppression strategy at the individual feature value level. This enables differential treatment of features, leading to better robustness-accuracy trade-offs. Moreover, FGAA is designed as a lightweight module built on simple fully connected layers, making it easily integrable into existing convolutional neural networks with negligible inference overhead. As shown in Tables 3 and 11, FGAA not only enhances robustness but can, in some cases, also improve natural accuracy, demonstrating its potential to strengthen model generalization beyond adversarial defense.

3. Fine-grained activation alignment

3.1. Empirical feature activation analysis

Our work focuses on analyzing feature activation patterns within neural networks. Table 1 summarizes the notation used in this paper. Feature maps in a layer are defined by three dimensions: width, height, and depth. The spatial dimension encompasses width and height, while the channel dimension corresponds to depth. We define a threshold as Threshold $= \mathrm{MAX} \cdot e^{-2}$, where MAX is the largest value of the space/channel magnitude. Here, MAX corresponds to $\max_j g_{\ell,j}(\boldsymbol{x})$ after global average pooling (GAP), so the threshold is consistent with $\tau_\ell(\boldsymbol{x}) = (\max_j g_{\ell,j}(\boldsymbol{x}))e^{-2}$ in Eq. (1). After performing a GAP operation, a spatial or channel unit is considered activated if the magnitude exceeds this threshold. We examined the spatial and channel activation frequencies and magnitudes for both adversarial and natural examples. We use RT to denote regular training and SAT to denote standard adversarial training. A comparison between the RT (*bird*) and SAT (*bird*) rows in Fig. 1 (and likewise for *frog*) shows that the activation distributions across spatial and channel dimensions are more aligned under adversarial training. Specifically, this convergence is evidenced by the alignment of the activations of the adversarial examples towards the activation distribution of the natural examples. After adversarial training, the overlap between the blue (natural) and red (adversarial) regions increases, as evidenced by the expanding purple region representing their intersection. To quantitatively characterize this convergence, we introduce the overlap ratio (OR), a metric that measures the similarity between activation patterns of natural and adversarial inputs. A formal definition follows.

Table 1
Overview of notation and definitions.

Notation	Definition
$\boldsymbol{x}, \boldsymbol{x}'$	Natural input and its adversarial counterpart
$\mathcal{A}_\ell(\boldsymbol{x})$	Activation tensor at layer ℓ
$g_\ell(\boldsymbol{x})$	Channel-wise global-average-pooled vector at layer ℓ
$\tau_\ell(\boldsymbol{x})$	Per-sample threshold for selecting active channels
$\mathcal{I}_\ell(\boldsymbol{x})$	Active-channel index set at layer ℓ
$\mathrm{OR}_\ell(\boldsymbol{x}, \boldsymbol{x}')$	Overlap ratio (IoU after thresholding) between active-channel sets
ϵ_{stab}	Numerical-stability constant
C, H, W, N, L	Channels, height, width, batch size, number of classes
$O_c[m, n]$	Local convolutional response for channel c at spatial index (m, n)
S	Batch of input feature maps to an FGAA module
$\tilde{S}$	Aligned feature maps
F	Filter
θ	Network parameters
$P(\boldsymbol{x}; \theta)$	Network output logits
$\mathcal{O}(\boldsymbol{x}; \theta, F)$	FGAA readout logits used in $\mathcal{L}_{FGAA}$
λ	Weight of the FGAA loss term
Γ	Number of FGAA modules; $\gamma \in \{1, \ldots, \Gamma\}$ indexes modules

Formal definition of OR. For layer ℓ with activations $\mathcal{A}_\ell(x) \in \mathbb{R}^{C\times H\times W}$, let $g_\ell(x) = \mathrm{GAP}(\mathcal{A}_\ell(x)) \in \mathbb{R}^C$ (GPA over $H \times W$) and $\tau_\ell(x) = \left(\max_j g_{\ell,j}(x)\right) e^{-2}$. Define the active-index set $\mathcal{I}_\ell(\boldsymbol{x}) = \{\, j \;:\; g_{\ell,j}(\boldsymbol{x}) \geq \tau_\ell(\boldsymbol{x}) \,\}$. Given an adversarial example $\boldsymbol{x}'$, the layer-wise overlap ratio (IoU after thresholding) is

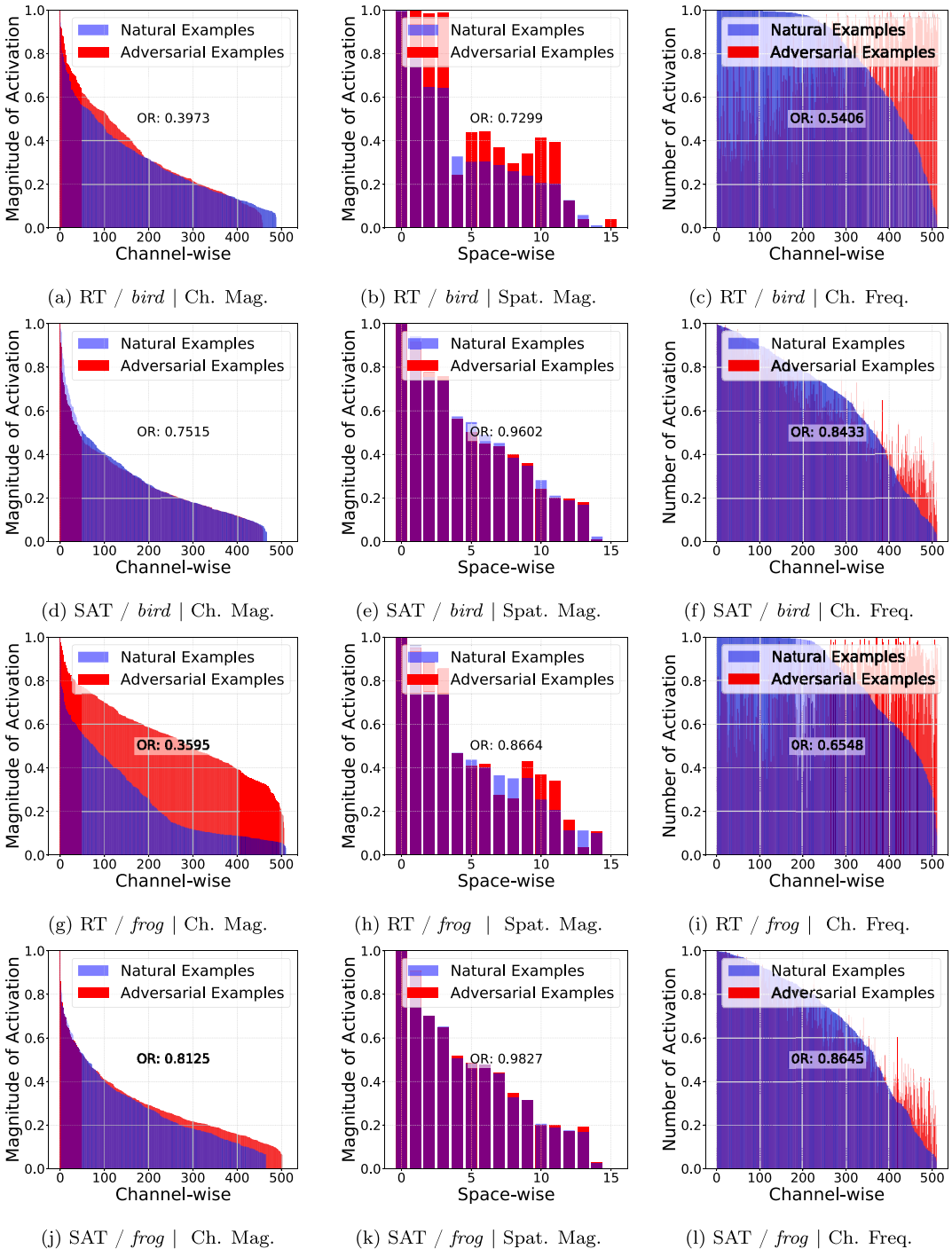
$$\mathrm{OR}_\ell(\boldsymbol{x}, \boldsymbol{x}') = \frac{\left|\mathcal{I}_\ell(\boldsymbol{x}) \cap \mathcal{I}_\ell(\boldsymbol{x}')\right|}{\left|\mathcal{I}_\ell(\boldsymbol{x}) \cup \mathcal{I}_\ell(\boldsymbol{x}')\right| + \epsilon_{\text{stab}}}, \tag{1}$$

where $\epsilon_{\text{stab}} = 10^{-8}$ is a numerical-stability constant independent of the PGD budget ϵ.

As shown in Fig. 1(a), (d), or (g), (j), both distributions nearly double their OR on channel activation after adversarial training. In Fig. 1(b), (e), or (h), (k), which depicts the spatial activation perspective, both distributions exhibit nearly the same magnitude of spatial activation post-adversarial training. A similar pattern is observed for channel-level activation frequencies. Overall, the adversarially trained model shows closer alignment in both spatial/channel activation magnitude and channel activation frequency compared to the model without AT.

3.2. Framework of FGAA

Our empirical observations, as detailed in Section 3.1 and further validated in Section 4.5, indicate that the OR of the feature activation distributions between natural examples and adversarial examples is closely related to the robustness of the model. In particular, a lower overlap ratio tends to signify a higher vulnerability to adversarial perturbations. We devised the FGAA at the feature level to further enhance the defensive effect of adversarial training and improve the adversarial robustness of neural networks. FGAA operates at a fine-grained feature level to address discrepancies caused by adversarial activations. The architecture of FGAA is shown in Fig. 2. First, FGAA determines the



(a) RT / *bird* | Ch. Mag. (b) RT / *bird* | Spat. Mag. (c) RT / *bird* | Ch. Freq.

(d) SAT / *bird* | Ch. Mag. (e) SAT / *bird* | Spat. Mag. (f) SAT / *bird* | Ch. Freq.

(g) RT / *frog* | Ch. Mag. (h) RT / *frog* | Spat. Mag. (i) RT / *frog* | Ch. Freq.

(j) SAT / *frog* | Ch. Mag. (k) SAT / *frog* | Spat. Mag. (l) SAT / *frog* | Ch. Freq.

Fig. 1. Penultimate-layer activations on CIFAR-10 with ResNet18 (PGD-20; classes *bird*, *frog*) under natural and adversarial inputs. Each triplet shows three panels in left-to-right order: channel-wise average activation magnitude (Ch. Mag.), spatial-wise average activation magnitude (Spat. Mag.), and channel activation frequency (Act. Freq.). Magnitudes are sorted in descending order; frequency is ranked by adversarial inputs. Methods: RT (regular training) and SAT (standard adversarial training).

importance of features by constructing relationships between these features and the ground-truth or predicted labels. Specifically, the weight corresponding to a feature map represents the importance of the feature on that map. The original feature values activated by adversarial examples are then fine-tuned and aligned based on the importance of the features. Next, the intermediate outputs in FGAA, which carry detailed feature information, are used to set the adversarial training loss function, thereby improving the model's robustness. As the network layers deepen, the extracted information becomes more complex and abstract. Embedding the FGAA module in the deeper layers of a model helps establish a direct relationship with the target category and suppress non-robust high-level features. Finally, the output features of the FGAA module are passed to the Fully Connected (FC) layer of the base model, thus completing the embedding of the FGAA module into the target model. It is important to note that the FGAA module can be integrated into most neural network models, enabling them to perform adversarial training and enhance the original model's robustness.

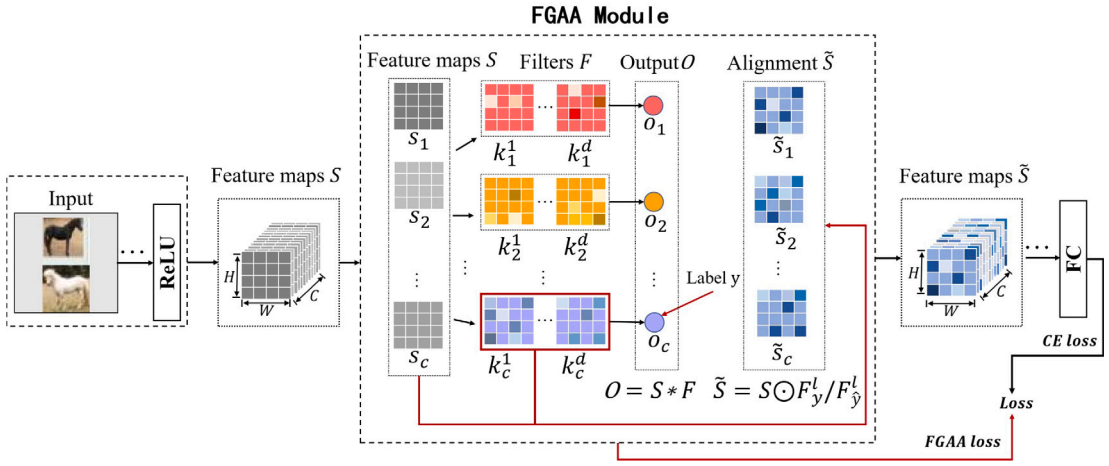


Fig. 2. The framework of the FGAA: (1) Establish the correspondence between the feature maps and the category labels. (2) Select the weights corresponding to the labeled categories to assess the importance of the feature values. (3) Perform fine-grained alignment of the raw feature activations based on the feature importance. Different colors distinguish various categories, while the shade of each color indicates the importance of the feature value for classifying a particular category. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.3. Feature importance acquisition

Taking epoch data as an example, let $S(N, C_{in}, H, W)$ represent the input feature map size of the FGAA module. By using a convolution kernel to perform feature transformation, we can obtain:

$$OUTPUT\left(N_i, C_{out_j}\right) = \sum_{k=0}^{C_{in}-1} S\left(N_i, c\right) * F\left(C_{out_j}, c\right), \tag{2}$$

where N and C denote the batch size and channel count of the activation feature maps, respectively. The indices i and j refer to the batch and channel positions, respectively. Here, F represents the convolution kernel parameters, and "$*$" denotes the convolution operation, which in practice corresponds to the cross-correlation operation [27].

Specifically, for the cth input feature map $S \in \mathbb{R}^{H\times W\times C}$, with a kernel size of $A\times B$, the indices a and b represent specific feature values within the convolution kernel. The values of the output feature indexed by (m, n) are calculated as follows:

$$O_c[m,n] = \sum_{a=0}^{A-1}\sum_{b=0}^{B-1} s_c[a+m, b+n] \times k_c[a,b]. \tag{3}$$

In our implementation, this convolution operation is equivalently realized through an FC layer that serves dual purposes: it generates classification logits while simultaneously providing feature importance weights. The size of the convolution kernel matches the input feature map size ($A = H, B = W$), which allows the feature maps to be transformed into a one-dimensional output corresponding to the number of predicted categories. This design establishes a direct connection between the FC layer's weight matrix and feature importance.

The FC layer processes the feature tensor to produce classification logits:

$$\mathcal{O}^{\gamma}(\boldsymbol{x}) = \mathrm{FC}^{\gamma}(\mathrm{flatten}(S^{\gamma})), \tag{4}$$

where the flatten operation reshapes $S^{\gamma} \in \mathbb{R}^{N\times C\times H\times W}$ appropriately for the FC layer to produce logits $\mathcal{O}^{\gamma} \in \mathbb{R}^{N\times L}$. Here, γ indexes individual FGAA modules within the network. The FGAA module design is highly flexible and can be integrated into most neural network architectures in arbitrary quantities. We denote the total number of FGAA modules as Γ, with each module indexed by $\gamma \in \{1, 2, \ldots, \Gamma\}$. The specific value of Γ varies depending on the network architecture and performance requirements—for instance, we typically employ $\Gamma = 2$ modules in ResNet18 for CIFAR-10, while deeper networks or more complex tasks may benefit from additional modules. Each γth FGAA module operates independently with its own FC layer parameters, processing its input features S^{γ} to produce module-specific outputs $\mathcal{O}^{\gamma}$.

During the training phase, the ground truth labels of the data are readily available. We use the weight values $F^l = \left[K_1^d[a,b], K_2^d[a,b], \ldots, K_c^d[a,b]\right]$ corresponding to the category y_l of the ground truth labels for subsequent activation alignment. In the testing phase, we use the weight values corresponding to the predicted label category $\hat{y}_l$. In our design, the convolutional kernel index d is directly matched to the label index l, such that $d = l$, where $d, l \in \{1, 2, \ldots, L\}$, and L denotes the total number of ground-truth categories. For instance, in the CIFAR-10 dataset, where $L = 10$, we define $d = l = 1, 2, \ldots, 10$. This one-to-one correspondence ensures that each kernel is explicitly associated with a unique semantic class, thereby identifying the features that contribute most to the category.

3.4. Feature activation alignment

The next step is to perform adversarial activation alignment on the output of the ReLU layer. Assuming that the output of the ReLU layer (the input feature map of the FGAA module) is denoted $S \in \mathbb{R}^{N\times C\times H\times W}$, we can align the activation of adversarial examples on the feature maps based on the feature importance obtained in Section 3.3:

$$\text{Alignment } \tilde{S} = \begin{cases} S \odot F_y^l, & \text{(training phase)} \\ S \odot F_{\hat{y}}^l, & \text{(test phase)}\ . \end{cases} \tag{5}$$

Here, $\odot$ denotes the Hadamard product operation. Feature alignment depends on feature importance F^l. In practice, each FGAA module operates through a coordinated sequence: it first generates classification logits $\mathcal{O}^{\gamma}$ for loss computation, then extracts the corresponding feature importance weights F^l from its FC weight matrix. These weights are applied via element-wise multiplication $\tilde{S}^{\gamma} = S^{\gamma} \odot F^l$ to align the features, which are subsequently propagated to the next layer.

Unlike CAS and CIFS that only consider channel-level activation, FGAA performs feature alignment in a more fine-grained manner (feature-wise) across both spatial and channel dimensions. This preserves valuable spatial information while selectively modulating feature activations based on their importance to classification.

3.5. Loss function for FGAA with AT

Network models with the FGAA module can integrate with existing AT techniques (e.g., SAT [12], TRADES [28], and MART [29]) to

enhance model robustness. During AT, the FGAA module dynamically aligns features activated by adversarial examples that do not contribute to the classification of natural examples, thereby improving the robustness of the original model. Let $P(\boldsymbol{x};\theta) \in \mathbb{R}^L$ denote the network output, and $\mathcal{O}(\boldsymbol{x};\theta,F) \in \mathbb{R}^L$ denote the FGAA readout logits. We use cross-entropy on logits: $\mathcal{L}_{ce}^{\text{logits}}(z,y) = -\log\big(\text{softmax}(z)_y\big)$. Let $\boldsymbol{x}'$ denote the adversarial example. The alignment loss of the FGAA module can be expressed as:

$$\mathcal{L}_{FGAA}(\boldsymbol{x}',y;\theta,F) = \mathcal{L}_{ce}^{\text{logits}}(\mathcal{O}(\boldsymbol{x}';\theta,F),y). \tag{6}$$

In summary, the overall AT loss function of the model, using SAT as an example, is:

$$\mathcal{L}_{AT}(\boldsymbol{x}',y;\theta,F) = \mathcal{L}_{ce}^{\text{logits}}\big(P(\boldsymbol{x}';\theta),\ y\big) \ + \ \frac{\lambda}{\Gamma}\sum_{\gamma=1}^{\Gamma}\mathcal{L}_{FGAA}^{\gamma}(\boldsymbol{x}',y;\theta,F^{\gamma}). \tag{7}$$

λ denotes the weighting factor of the FGAA loss component, which controls the degree of feature alignment enforced during training. Larger values enforce stronger alignment, and we tune λ empirically. Γ denotes the total number of FGAA modules. Dividing by Γ averages the FGAA loss across modules, so λ controls the per-module alignment strength regardless of how many modules are used. For instance, when deploying FGAA in ResNet18, the parameters are configured as follows: $\lambda = 2$ (alignment strength), and $\Gamma = 2$ (number of FGAA modules used). The training procedure for the neural network with FGAA is outlined in Algorithm 1.

Algorithm 1 FGAA with Robust Training

Input: Dataset $\mathcal{D}$; network $\mathcal{F}$ with FGAA modules $\{\gamma = 1,\dots,\Gamma\}$; epochs T; batch size τ; PGD steps κ; step size α; ℓ_∞ budget ϵ; loss weight λ.
Output: Robust model $\mathcal{F}(\theta)$.

for $t = 1$ to T **do**
  for each mini-batch $(\boldsymbol{x},y)$ of size τ **do**
    Adversary generation (PGD): initialize $\boldsymbol{x}' \leftarrow \boldsymbol{x} + \text{Uniform}(-\epsilon,\epsilon)$.
    for $s = 1$ to κ **do**
      $g \leftarrow \nabla_{\boldsymbol{x}'}\mathcal{L}_{AT}(\boldsymbol{x}',y;\theta,\mathbf{F})$ ▷ maximize Eq. (7) w.r.t. input
      $\boldsymbol{x}' \leftarrow \Pi_{B_\epsilon(\boldsymbol{x})}\big(\boldsymbol{x}' + \alpha\,\text{sign}(g)\big)$; $\boldsymbol{x}' \leftarrow \text{clip}(\boldsymbol{x}',0,1)$
    end for
    FGAA forward with alignment:
    for each FGAA module $\gamma = 1,\dots,\Gamma$ **do**
      compute class-wise logits $\mathcal{O}^{\gamma}(\boldsymbol{x}';\theta,F^{\gamma}) \in \mathbb{R}^L$ ▷ FGAA readout logits
      pick importance weights $F^{(l)}$ from the module FC weights,
      where l=y in training and l=$\hat{y}$=$\arg\max P(\boldsymbol{x}';\theta)$ in testing
      element-wise alignment $\tilde{S}^{\gamma} \leftarrow S^{\gamma} \odot F^{(l)}$ and propagate $\tilde{S}^{\gamma}$ to next layers
    end for
    compute main logits $P(\boldsymbol{x}';\theta) \in \mathbb{R}^L$
    Loss: $\mathcal{L}_{CE} \leftarrow \mathcal{L}_{ce}^{\text{logits}}\big(P(\boldsymbol{x}';\theta),y\big)$
    $\mathcal{L}_{FGAA} \leftarrow \frac{1}{\Gamma}\sum_{\gamma=1}^{\Gamma}\mathcal{L}_{ce}^{\text{logits}}\big(\mathcal{O}^{\gamma}(\boldsymbol{x}';\theta,F^{\gamma}),y\big)$
    $\mathcal{L}_{AT} \leftarrow \mathcal{L}_{CE} + \lambda\mathcal{L}_{FGAA}$ ▷ matches Eq. (7)
    update parameters $(\theta,\mathbf{F})$ by gradient descent on $\mathcal{L}_{AT}$
  end for
end for

Although FGAA introduces a slight increase in training time, its inference-time overhead is negligible, as it operates through feature-level alignment without adding substantial parameters or complexity. Importantly, FGAA is modular and can be selectively applied to specific layers, enabling a flexible trade-off between robustness and efficiency. Furthermore, its internal structure, composed of simple fully connected layers, is compatible with common model compression techniques such as pruning, quantization, and knowledge distillation, making it suitable for deployment in resource-constrained IoT environments.

Table 2
Hyperparameters used for training FGAA_ResNet18 on CIFAR-10 and SVHN.

Parameters	FGAA_SAT	FGAA_Trades	FGAA_Mart
batch_size	128	128	128
epoch	200	100	120
lr	e−2	e−2	e−2
weight_decay	$2e^{-4}$	$2e^{-4}$	$2e^{-4}$
FGAA_beta	2	2	2
num_steps	10	10	10
epsilon	8/255	0.031	0.031
alpha	2/255	0.007	0.007
beta	–	4.0	5.0

4. Experiments

4.1. Experimental setup

Models & Datasets. The FGAA module is designed for flexible integration into various neural network architectures. In our experiments, we incorporated FGAA into ResNet18, VGG16, and WideResNet34-10, denoted as FGAA_ResNet18, FGAA_VGG16, and FGAA_WideResNet34-10, respectively. These models were trained on three widely used image classification benchmarks: CIFAR-10 [30], SVHN [31], and Fashion-MNIST [32]. These datasets are extensively used in adversarial robustness research due to their diversity, varying levels of visual complexity, and the availability of well-established baselines, making them suitable for controlled and reproducible evaluation of defense mechanisms.

While these datasets are not explicitly tailored for IoT applications, they provide a representative and practical foundation for assessing the fundamental behavior of our method. FGAA is modular and model-agnostic, and its design allows seamless adaptation to resource-constrained models typically deployed in IoT settings. In future work, we plan to further validate FGAA's effectiveness using IoT-specific datasets — such as those featuring low-resolution imagery, embedded sensors, or edge-device benchmarks — to demonstrate its real-world applicability in constrained environments.

For adversarial training, we adopted three well-known AT methods: SAT [12], TRADES [28], and MART [29], using their official parameter settings whenever available. Table 2 details the key hyperparameters for training FGAA_ResNet18 on CIFAR-10.

Attacks Methods. To evaluate the contribution of FGAA in improving model robustness, we employed a variety of strong adversarial attacks, including FGSM [33], PGD-20 [12], CW_∞ [34], Avg-PGD-100 [35] and AutoAttack [14]. The FGSM, PGD-20, CW_∞, and Avg-PGD-100 attacks used to challenge the FGAA-based defense strategy are based on Eq. (7). In contrast, AutoAttack retains its official settings.[1] For comparison, we also included the CAS defense strategy. Unless otherwise specified, robustness evaluation is conducted using the model obtained from the last epoch. The code is available at.[2]

4.2. Robustness evaluation

4.2.1. White-box attack

The models compared are the final AT epoch (*last*) and the AT checkpoint that achieves the highest robust validation accuracy under PGD-20 (*best*). The defense results are reported in Table 3. Our proposed FGAA method achieves nearly the same natural accuracy as the CAS method, but with significantly higher robustness on CIFAR-10

[1] https://github.com/fra31/auto-attack.
[2] https://github.com/wenxinkuang/FGAA.

Table 3
White-box robustness (%) of the best and last checkpoint of ResNet18 with the FGAA module, obtained through AT on the CIFAR-10 and SVHN datasets. (best = highest *validation* robust acc under PGD-20; last = final epoch.)

Dataset	Defense	Natural		FGSM		PGD-20		CW_∞	
		Best	Last	Best	Last	Best	Last	Best	Last
CIFAR-10	SAT	83.70	83.60	61.81	59.52	49.50	54.50	49.22	46.78
	CAS_SAT	86.16	86.24	61.75	59.27	53.53	46.56	64.11	51.70
	FGAA_SAT	**86.31**	86.19	61.53	**60.84**	**62.73**	**62.18**	**86.67**	**86.62**
	Trades	84.04	84.52	64.08	64.78	54.12	53.09	52.06	51.86
	CAS_Trades	85.50	85.50	64.59	65.23	53.04	53.02	56.31	56.18
	FGAA_Trades	81.97	82.06	61.46	60.79	**58.21**	**57.45**	**75.79**	**75.45**
	Mart	80.90	82.31	64.74	65.34	56.17	55.16	52.11	51.61
	CAS_Mart	86.41	87.07	63.73	63.80	59.50	55.52	71.64	65.16
	FGAA_Mart	85.53	85.83	61.40	60.61	**59.59**	**58.16**	**83.60**	**83.59**
SVHN	SAT	89.40	89.41	67.52	67.27	53.06	53.21	48.53	48.43
	CAS_SAT	86.90	93.96	93.42	73.57	30.26	56.34	51.06	59.52
	FGAA_SAT	**93.59**	93.57	72.98	72.85	**61.32**	**61.26**	**85.94**	**86.56**
	Trades	91.78	91.31	73.68	73.36	60.00	59.98	55.46	55.24
	CAS_Trades	91.99	91.99	74.22	73.85	59.81	59.72	58.06	58.06
	FGAA_Trades	**92.66**	**92.21**	**74.31**	73.38	**61.54**	**61.09**	**78.40**	**80.13**
	Mart	84.10	88.17	72.80	71.50	67.98	65.60	66.85	64.79
	CAS_Mart	93.71	94.28	74.38	74.91	62.58	60.93	68.23	66.49
	FGAA_Mart	**94.01**	93.97	73.61	73.51	**65.24**	**65.30**	**87.65**	**87.85**

and SVHN. The channel suppression method CAS suppresses channels activated by adversarial examples to a certain extent, but it also suppresses features that are important for the prediction of natural examples. In contrast, FGAA aligns features activated by adversarial examples in a more fine-grained manner, significantly reducing the gap between the feature distributions of adversarial and natural examples. Consequently, FGAA effectively defends against more aggressive attacks such as PGD-20, Avg-PGD-100, and CW_∞, showing great potential particularly against CW_∞ attacks. This is because FGAA employs a finer-grained feature-level weighting rather than the uniform channel weighting used by CAS. The alignment provided by FGAA is more precise. After the original feature maps pass through the FGAA alignment module, features that correctly classify adversarial examples are amplified while those that misclassify are suppressed, effectively purifying adversarial examples and preventing misclassification. It is worth noting that models integrated with the FGAA module do not always outperform all baselines under every attack scenario. Specifically, under the FGSM attack, FGAA-equipped models may exhibit slightly lower robustness compared to some baseline defenses. This phenomenon can be attributed to the nature of FGSM itself. FGSM is a single-step attack that introduces relatively coarse perturbations. Many defense methods (e.g., adversarial training) are more effective against fine perturbations from multi-step attacks. This does not imply the defense methods are ineffective, but rather that they are designed to handle better complex attacks (like PGD) rather than simple single-step FGSM perturbations. Thus, the performance drop under FGSM reflects the nature of the attack, not a limitation of FGAA. Moreover, on datasets like SVHN, the FGAA not only demonstrates improved robustness but also achieves higher natural accuracy than the baseline adversarial training (AT), illustrating FGAA's potential to boost both robustness and generalization. Additional results on other architectures, including VGG16 and WideResNet34-10, as well as broader dataset evaluations, are provided in Section 4.7.

4.2.2. Stability analysis

Compared with RT, AT is prone to overfitting [36]. During AT, the robust accuracy on the training data and test data will show opposite trends after a certain epoch: one continues to increase steadily while the other decreases. Therefore, many studies [12,18,19,36] favor evaluating the final (last epoch) model obtained from AT rather than the best model obtained at a certain epoch. Nevertheless, the best model is an important reference. As shown in Table 3, the CAS strategy exhibits a significant gap between the robustness accuracy of the last and best models obtained under all three AT methods, indicated by underlined values in the table. The last and best models obtained in CAS_SAT mode have a robustness gap of up to 9.93% in defending against CW_∞ attacks, whereas FGAA maintains this gap within approximately 1%. This indicates that the CAS strategy cannot consistently capture robust features during training. Consequently, further training after reaching the peak robust model at a certain epoch is largely unproductive. In contrast, the model trained with our proposed FGAA module demonstrates greater stability. The primary reason for the poor model fitting with the CAS strategy is its direct suppression of features across the entire channel.

Table 4
Robustness (%) of FGAA combined with different adversarial training (AT) methods under the AutoAttack benchmark on CIFAR-10 using ResNet18.

Defense method	Natural accuracy (%)	AutoAttack accuracy (%)
SAT	83.60	56.20
CAS_SAT	86.24	59.34
FGAA_SAT	86.19	**62.68**
Trades	84.52	60.77
CAS_Trades	85.50	60.94
FGAA_Trades	82.06	**62.41**
Mart	82.31	58.74
CAS_Mart	87.07	57.31
FGAA_Mart	85.83	**63.61**

4.2.3. AutoAttack

AutoAttack is currently the most widely used method for evaluating robustness. AutoAttack integrates three white-box attacks (APGD-CE, APGD-T, and FAB-T) and one black-box attack (Square Attack). The results of FGAA against AutoAttack are shown in Table 4. The FGAA strategy, when combined with different AT methods, consistently achieves impressive robustness results. While FGAA does not always yield the highest natural accuracy (e.g., CAS_SAT slightly exceeds FGAA_SAT, and CAS_Trades outperforms FGAA_Trades in clean accuracy), it consistently achieves the strongest robustness under AutoAttack. This reflects FGAA's design emphasis on stabilizing task-relevant features against perturbations, leading to robustness gains at the cost of marginal reductions in natural accuracy. Overall, FGAA provides a favorable robustness–accuracy balance compared with baseline and CAS methods.

Table 5
Robustness (%) for different λ_{attack} values. The values of λ_{attack} indicate the locations of the attack target within the model. $\lambda_{attack} = 0$ indicates that the attack is launched against the last layer of the model, $\lambda_{attack} = \infty_1$ indicates that the attack targets the first FGAA module of ResNet18, $\lambda_{attack} = \infty_2$ indicates that the attack targets the second FGAA module, and $\lambda_{attack} = \infty$ indicates that the attack is directed solely against the entire FGAA module. The base model is ResNet18, and the data is CIFAR-10.

Defenses	λ_{attack}	Natural	FGSM	PGD-20	CW_∞	Avg-PGD-100
CAS_SAT	0	86.24	88.62	89.79	89.73	89.65
	0.1	–	61.67	60.72	85.25	62.67
	1	–	59.63	49.33	58.18	51.27
	10	–	59.22	44.56	48.12	44.55
	100	–	59.14	44.02	47.67	43.87
	∞_1	–	59.01	43.40	46.08	40.89
	∞_2	–	88.47	89.65	88.95	89.83
	∞	–	59.14	44.27	47.65	43.67
FGAA_SAT	0	86.19	90.24	92.34	92.17	92.19
	0.1	–	66.82	69.05	89.95	69.55
	1	–	61.88	62.83	86.83	64.48
	10	–	60.16	60.55	86.30	62.39
	100	–	59.58	59.48	86.36	61.57
	∞_1	–	56.99	40.92	44.81	38.48
	∞_2	–	90.41	92.75	91.49	93.24
	∞	–	59.48	58.88	86.15	60.70

4.3. Adaptive attack

FGAA enhances the robustness of the original model by incorporating FGAA modules into the base model and integrating it with conventional AT. Assuming full awareness of FGAA's defense strategies, we design adaptive attacks specifically for FGAA. Specifically, we generate intensified adversarial examples using an adaptive loss function derived from Eq. (7). This loss function comprises two components: the conventional AT loss $\mathcal{L}_{ce}$ and the FGAA loss $\mathcal{L}_{FGAA}$. The attacker can adjust the parameter λ to control the relative weight of these components. λ takes values from the set $\{0, 0.1, 1, 2, 10, 100, \infty\}$. For $\lambda = 0$, the loss focuses only on $\mathcal{L}_{ce}$, while larger values progressively increase the influence of $\mathcal{L}_{FGAA}$. Table 5 presents the adaptive attack results under different λ_{attack} values. We report results using $\lambda_{\text{attack}} = 2$, balancing both loss components effectively.

4.4. Alignment patterns and effects

Due to the significant discrepancies between the activation distributions of adversarial and natural examples in both spatial and channel dimensions, we aimed to narrow this gap through various alignment strategies. We selected four types of alignment strategies: CAS_SAT for channel-wise activation alignment, essentially a channel-wise feature alignment method; SAA_SAT for spatial dimension activation alignment; CSAA_SAT, which involves alignment from the channel perspective followed by spatial dimension activation alignment; and FGAA_SAT, a FGAA, which offers a more granular alignment in both spatial and channel dimensions. To ensure a fair comparison, we replaced the last block of the base model with the respective alignment module.

Table 6 indicates that solely aligning from the spatial dimension negatively affects the semantics of channel dimension features, leading to a notable decrease in the model's prediction accuracy for natural examples and its ability to defend against adversarial example attacks. This suggests that the correlation between spatial dimension features is weaker compared to that between channel dimension features. However, it is essential to consider the semantics of the spatial dimension; for instance, the defense effect of solely suppressing the channel dimension (CAS_SAT) is less effective than aligning both spatial and channel dimensions simultaneously, as demonstrated by CSAA_SAT and FGAA_SAT. This demonstrates the existence of an activation discrepancy between adversarial and natural examples in the spatial dimension and highlights the potential for enhancing model robustness by mitigating this discrepancy. Among the alignment strategies, FGAA, offering a more fine-grained alignment, exhibits superior defense against more potent attacks such as PGD-20 and CW_∞.

Table 6
Robustness (%) of different activation alignment strategies. The base model is ResNet18, and the dataset used is CIFAR-10. The strategies include CAS for channel-wise alignment, SAA for spatial alignment, CSAA for channel followed by spatial alignment, and FGAA for fine-grained feature-level alignment.

Defenses	Natural	FGSM	PGD-20	CW_∞
CAS_SAT	86.24	59.27	46.56	51.70
SAA_SAT	64.31	42.89	41.40	45.53
CSAA_SAT	86.50	63.04	59.94	83.00
FGAA_SAT	86.19	60.84	**62.18**	**86.62**

4.5. Empirical feature activation analysis of FGAA

The OR (see Eq. (1)) quantifies the similarity of internal feature activations between natural and adversarial inputs. Intuitively, a higher OR suggests that adversarial perturbations have a more limited influence on the model's internal representations, potentially indicating stronger adversarial robustness. To empirically assess this hypothesis, we examine the correlation between OR values and adversarial robustness across multiple defense strategies on the CIFAR-10 dataset along three dimensions: channel magnitude, spatial magnitude, and channel frequency.

As illustrated in Fig. 3, models trained via RT exhibit the lowest OR values across all three dimensions. These low OR values correspond to poor robustness (e.g., 49.50%), indicating that unstable internal activations are associated with model vulnerability. This observation is consistent with the findings of Bai et al. [18], who reported that adversarial examples tend to induce larger and more uniformly distributed activations compared to natural examples. However, their analysis was predominantly qualitative and lacked a precise quantitative metric. In contrast, our use of the OR metric provides a clear and explicit means of quantifying activation consistency.

Compared to RT, conventional adversarial training methods (SAT, TRADES, and MART) consistently yield higher OR values. Among them, TRADES achieves the highest OR values across all three dimensions, as shown in Fig. 3(m), (n), and (o). This result aligns with TRADES also exhibiting the strongest robustness and highest natural accuracy among the three, reinforcing the view that adversarial training helps align internal feature representations between natural and adversarial inputs, thus improving robustness under moderate perturbations.

Notably, across all nine evaluated defense methods, models trained with adversarial strategies consistently exhibit more aligned activation patterns — especially in spatial magnitude and channel frequency — compared to RT. These methods collectively promote a gradual convergence of adversarial activation distributions toward those of natural inputs, thereby reducing their divergence and enhancing robustness. This supports the notion that improving activation distribution alignment is central to effective adversarial defense.

In contrast, CAS-based methods display inconsistent OR performance. While they achieve relatively high OR values in the spatial magnitude dimension, their alignment in channel magnitude and frequency is notably weaker. This partial alignment limits robustness gains and introduces instability. For example, CAS_SAT achieved the best robustness of 53.53% on the PGD-20 dataset, but the robustness of its last model was only 46.56%—lower than all other evaluated defense methods. These results highlight that spatial activation consistency is necessary but not sufficient: strong performance in a single dimension

does not guarantee overall robustness. For instance, although CAS_SAT outperforms FGAA_SAT in spatial magnitude OR, FGAA exceeds CAS by approximately 17% in channel alignment and ultimately achieves superior robustness.

Our proposed FGAA method achieves consistently high OR values across all dimensions, leading to the most substantial and stable improvements in robustness. Specifically, the best model robustness of FGAA_SAT is 62.73%, and the last model robustness is 62.18%. FGAA effectively promotes the convergence of adversarial activation distributions toward those of natural inputs, thereby reducing distributional discrepancies and progressively increasing OR. Our activation analysis confirms that as adversarial activations increasingly resemble those of natural inputs, the model's robustness improves significantly. While CAS effectively reduces spatial activation magnitude discrepancies, it also causes near-complete suppression of certain channels, as shown in Fig. 3(g) and (i). As a coarse-grained channel suppression method, CAS risks discarding important discriminative features. By contrast, FGAA employs a fine-grained, feature-level alignment strategy that selectively suppresses adversarially exaggerated features while preserving critical activations from natural examples. As shown in Fig. 3(l), FGAA provides a more comprehensive regulation of activation distributions, particularly in channel frequency. This targeted regulation substantially reduces the gap in activation frequency between adversarial and natural inputs, leading to enhanced model robustness.

In summary, our analysis demonstrates a strong empirical link between activation distribution alignment and adversarial robustness. Models that maintain high OR across multiple activation dimensions exhibit superior resilience to adversarial perturbations.

4.6. Ablation study

4.6.1. Effectiveness of the FGAA

There is a significant discrepancy between the activation feature distribution of adversarial and natural examples within neural networks. AT is widely acknowledged as one of the most effective methods for enhancing model robustness. However, while AT mitigates this bias to some extent, it often comes at the expense of the accuracy of natural examples. The primary objective of our FGAA module is to address the limitations of AT and further reduce the activation bias. To rigorously evaluate the efficacy of the FGAA module independently, we decouple it from AT. We investigated whether FGAA could enhance robustness without being integrated with AT. The strategies include RT, channel-wise activation suppression with regular training (CAS_RT), fine-grained activation alignment with regular training (FGAA_RT), adversarial training (SAT), and fine-grained activation alignment combined with adversarial training (FGAA_SAT). The results are presented in Table 7.

Without adversarial training, FGAA_RT attains 57.47% robust accuracy under PGD-20, markedly above typical RT levels. This gain does not result from an explicit suppression of adversarially sensitive activations. Instead, FGAA's fine-grained activation alignment emphasizes task-relevant activations in both spatial and channel dimensions. This emphasis reduces the influence of perturbation-sensitive, less relevant features on the final prediction. As a result, the model shows meaningful robustness even under RT.

The base ResNet18 model, when trained with RT, shows significant vulnerability to adversarial attacks, evidenced by 0% robustness against PGD-20 attacks. When adversarial training is applied (SAT), the robustness improves dramatically compared to RT. Embedding the CAS module, which aligns activations from the channel perspective, provides some enhancement in robustness. For instance, the CAS_RT model shows slight improvement in defending against adversarial examples, achieving a 6.8% robust accuracy against the Avg-PGD-100 attack. However, this improvement remains modest. In contrast, the FGAA module shows substantial improvement in robustness, even without AT. The FGAA_RT demonstrates significant adversarial robustness, comparable to models that employ AT. Notably, the FGAA_RT model outperforms the SAT model in defending against several basic attacks such as FGSM and PGD-20. The experimental results indicate that the FGAA module inherently enhances robustness. When combined with adversarial training (FGAA_SAT), the model achieves maximal robustness, effectively defending against powerful attacks such as PGD-20, CW_∞ and Avg-PGD-100.

Our observations confirm that RT leaves models highly susceptible to adversarial attacks. The introduction of AT significantly improves robustness. However, the most substantial gains are observed when employing the FGAA module, both in isolation and in conjunction with adversarial training. This approach not only boosts inherent robustness but also maximizes the model's defensive capabilities against sophisticated adversarial attacks.

Table 7
Robustness (%) of different activation alignment strategies with regular and adversarial training. The base model is ResNet18, and the dataset is CIFAR-10. The strategies evaluated include: ResNet18 with RT, ResNet18 with a channel-wise activation suppression module under RT (CAS_RT), ResNet18 with a fine-grained activation alignment module under RT (FGAA_RT), ResNet18 with SAT, and ResNet18 with a fine-grained activation alignment module combined with SAT (FGAA_SAT).

Defenses	Natural	FGSM	PGD-20	CW_∞	Avg-PGD-100
RT	94.21	8.70	0.0	24.39	0.0
CAS_RT	94.25	38.59	13.57	33.01	6.80
FGAA_RT	93.37	**69.28**	**57.47**	**45.44**	**32.93**
SAT	83.61	59.21	45.50	46.73	42.75
FGAA_SAT	**86.19**	**60.84**	**62.18**	**86.62**	**63.45**

Table 8
Robustness (%) of the FGAA strategy against PGD-20 under various perturbation strengths. The base model is ResNet18, and the dataset is CIFAR-10.

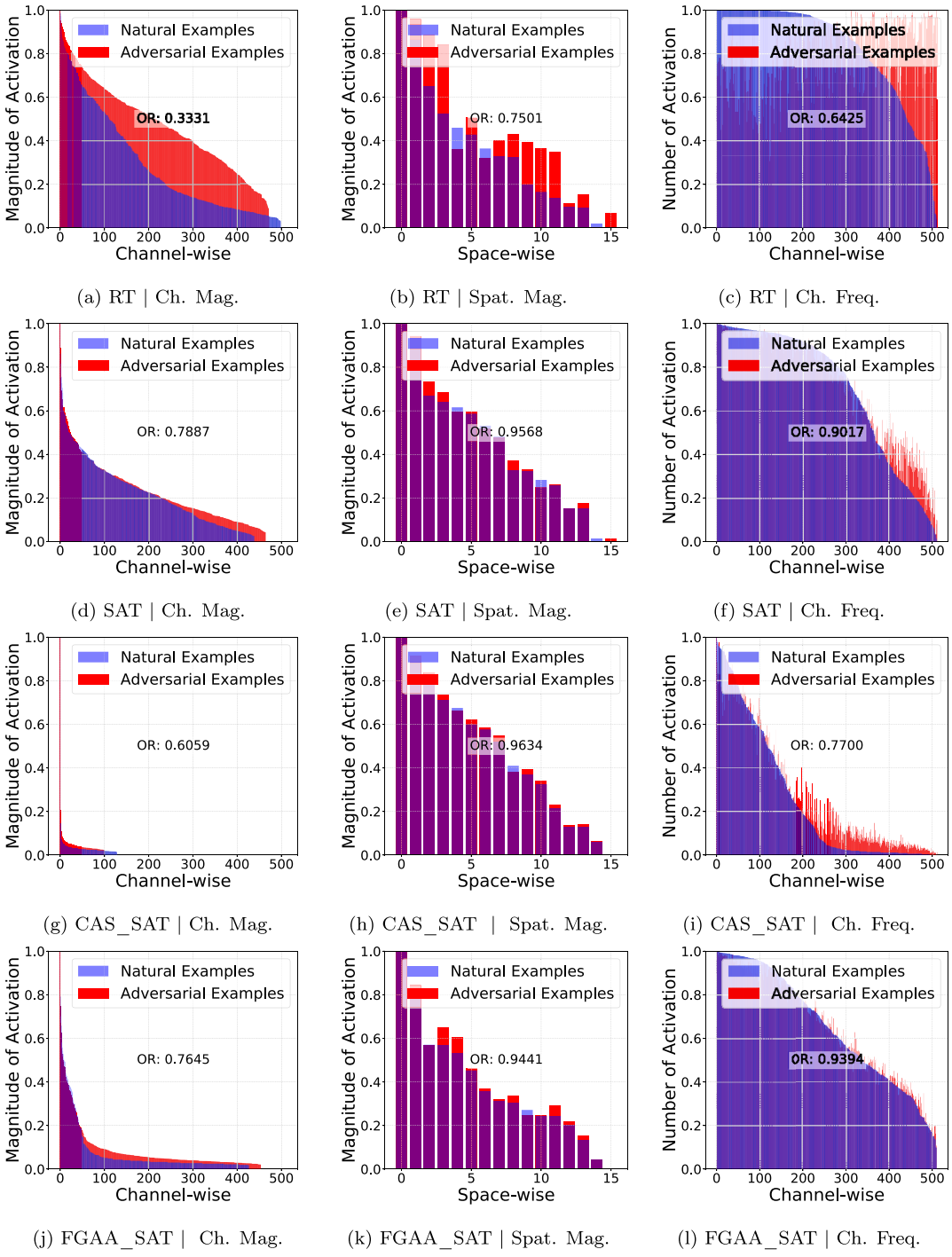
ϵ	2/255	4/255	8/255	16/255	32/255
CAS_SAT	79.15	70.26	48.88	16.64	2.48
FGAA_SAT	**81.24**	**74.88**	**61.95**	**51.33**	**36.43**
CAS_Trades	80.09	72.69	55.99	27.94	5.10
FGAA_Trades	77.22	71.06	**57.36**	**35.02**	**15.91**
CAS_Mart	81.01	73.56	54.37	20.49	2.19
FGAA_Mart	80.69	**74.22**	**58.50**	**41.56**	**27.10**

4.6.2. Robustness under different ϵ settings

In general, a larger value of ϵ indicates a stronger attack and results in lower robustness accuracy of the attacked model. We evaluate the robustness of FGAA with ResNet18 under different attack strengths ϵ using PGD-20 as an example. ϵ represents the level of perturbation, which is typically set to 8/255 empirically, as shown in Table 2. The evaluation results are presented in Table 8. It can be observed that the robustness of FGAA consistently outperforms CAS, regardless of the increase in attack strength. Even when the perturbation strength reaches 32/255, FGAA can still defend against the adversarial examples.

4.6.3. Robustness under different λ settings

In the FGAA module, we use λ to control the strength of feature alignment. To test the robustness of the FGAA training strategy under different alignment strengths, we varied λ values to 0.5, 1, 2, 5, 10, and 20, where larger λ values indicate greater alignment strength. Table 9 presents the model's robustness under white-box attacks for different alignment strengths, using the same λ values for both training and attacks. The results demonstrate that the presence of alignment generally enhances the model's robustness. However, excessive alignment causes the model to overly rely on the FGAA module, thereby neglecting the base model's original classification loss. This results in a significant drop in both robust accuracy and the prediction accuracy for natural



(a) RT | Ch. Mag. (b) RT | Spat. Mag. (c) RT | Ch. Freq.

(d) SAT | Ch. Mag. (e) SAT | Spat. Mag. (f) SAT | Ch. Freq.

(g) CAS_SAT | Ch. Mag. (h) CAS_SAT | Spat. Mag. (i) CAS_SAT | Ch. Freq.

(j) FGAA_SAT | Ch. Mag. (k) FGAA_SAT | Spat. Mag. (l) FGAA_SAT | Ch. Freq.

Fig. 3. Penultimate-layer activations for *airplane* on CIFAR-10 with ResNet18 (PGD-20) under natural and adversarial inputs. Triplets show Ch. Mag., Spat. Mag., Act. Freq. (as in Fig. 1). Methods: RT, SAT, CAS_SAT, FGAA_SAT, TRADES, CAS_TRADES, FGAA_TRADES, MART, CAS_MART, FGAA_MART.

examples. For ResNet18, the model achieves the best balance between robust accuracy and natural example prediction accuracy when $\lambda = 2$.

4.6.4. Attack different target modules

To further test the robustness of the FGAA defense strategy, we conducted an ablation study. We attacked the final layer and the FGAA/CAS layers of the model embedded with the FGAA/CAS module, respectively. The results are shown in Table 10. The FGAA defense strategy demonstrated greater robustness compared to the CAS defense strategy, regardless of whether the attacker targeted the final layer of the trained model or a specific FGAA/CAS layer. Furthermore, the robustness of FGAA exhibited minimal fluctuation with changes in the target attack module. In contrast, under the CAS defense strategy, the robustness will show a decreasing trend once the attack is directed against its specific CAS module.

4.7. More experimental results

4.7.1. FGAA_VGG16 results on CIFAR-10

VGG16 consists of five blocks containing convolutional layers. Similar to FGAA_ResNet18, we replaced the convolutional layers within the last block with FGAA modules. Since there are three convolutional

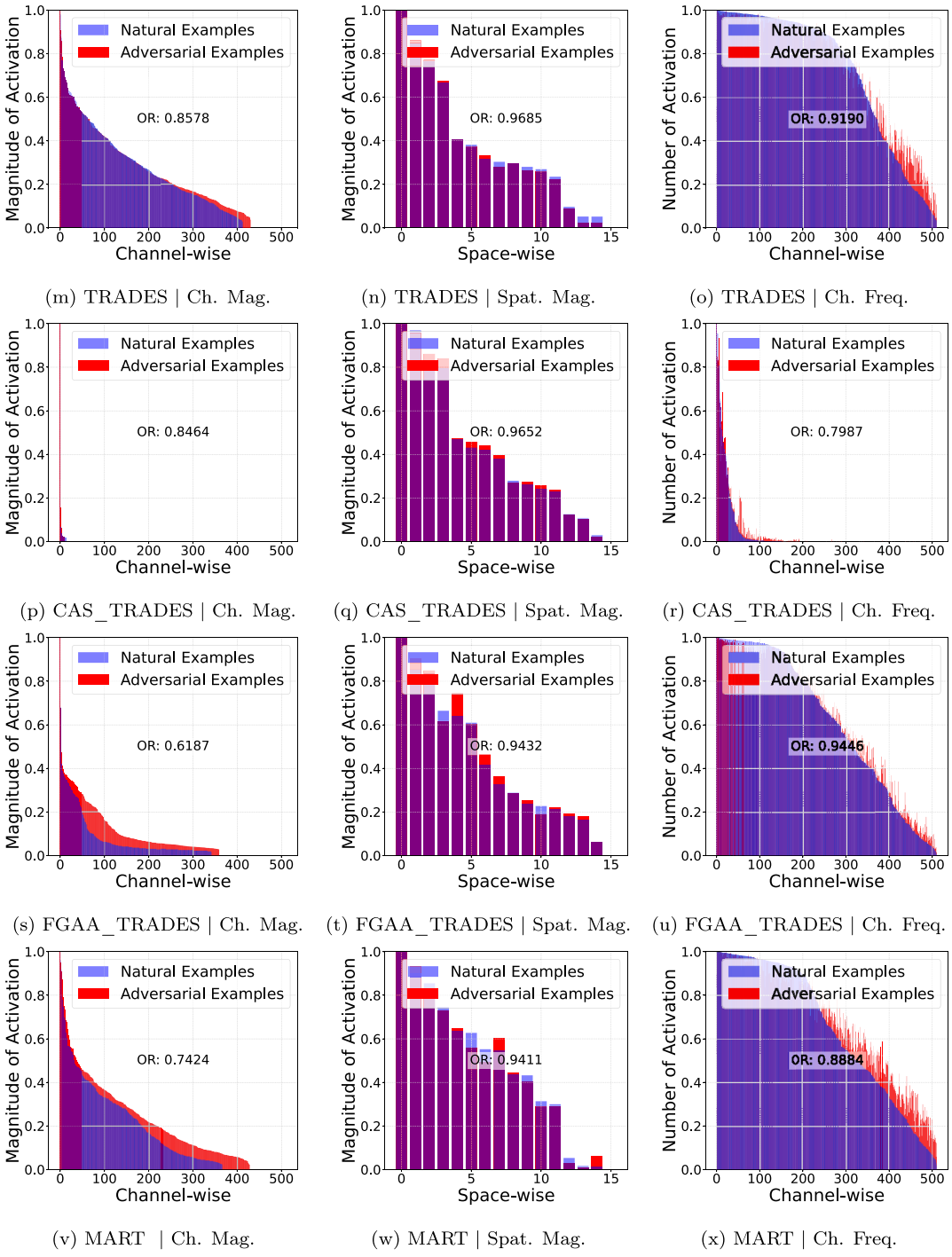
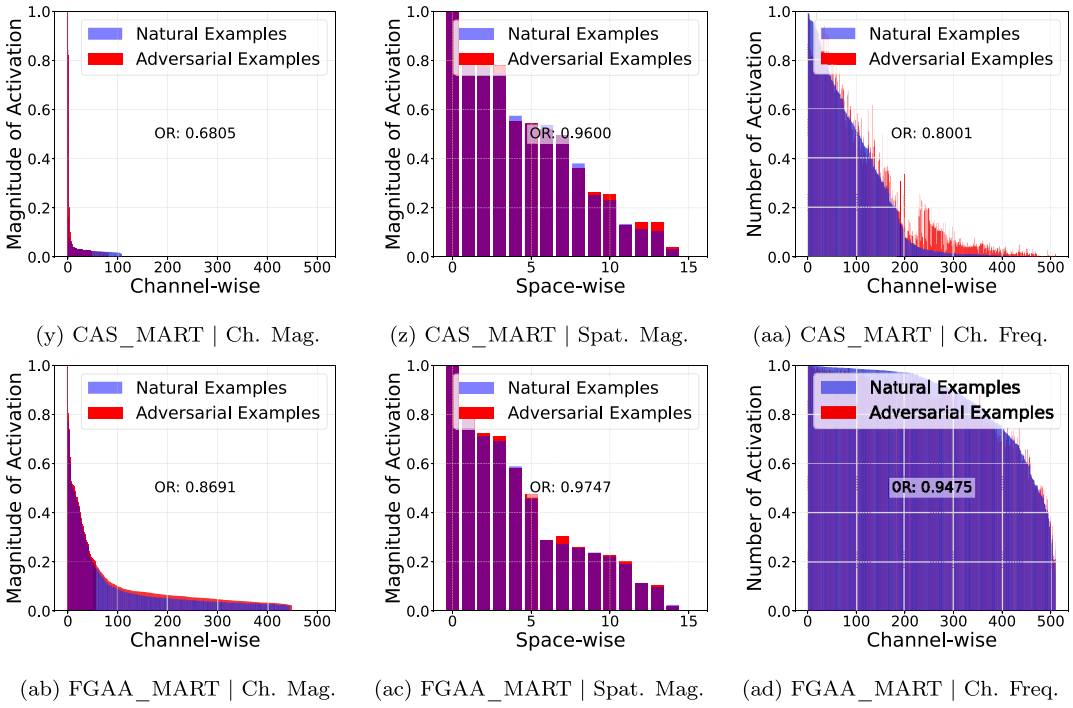


Fig. 3. (*continued*).

layers in the last block, VGG16 includes three FGAA modules, and $\Gamma = 3$. We trained the VGG16 embedded with FGAA modules using SAT, Trades, and Mart. The robust accuracy results of the trained models are shown in Table 11. The natural example prediction accuracy and adversarial robustness accuracy of the three models trained with only adversarial training are less than satisfactory. However, with the addition of the FGAA training strategy, the loss in natural example prediction accuracy is significantly compensated for each AT method. For example, the natural example prediction accuracy of FGAA_Mart is improved by 10.73% compared to the model trained using only Mart. Additionally, FGAA enhances the model's robustness. Notably, FGAA combined with Trades proves to be the most effective, with FGAA_Trades achieving an improvement of up to 22.65% in adversarial robustness accuracy against CW_∞ attacks compared to Trades alone.

4.7.2. FGAA_WideResNet34-10 results on CIFAR-10

When integrated with the network structure of WideResNet-34-10, the FGAA module replaces the convolutional layer in its final convolution block. Thus, two FGAA modules are embedded in WideResNet-34-10 with $\Gamma = 2$. The robust results of FGAA_WideResNet-34-10 are shown in Table 12. It is observed that among the three different models, the FGAA strategy provides the most significant improvement in natural example prediction accuracy for the simpler, straight-type VGG network. The enhancement in robustness for relatively more complex



(y) CAS_MART | Ch. Mag. (z) CAS_MART | Spat. Mag. (aa) CAS_MART | Ch. Freq.

(ab) FGAA_MART | Ch. Mag. (ac) FGAA_MART | Spat. Mag. (ad) FGAA_MART | Ch. Freq.

Fig. 3. (*continued*).

Table 9
Robustness (%) at different λ values in FGAA module. The base model is ResNet18 and the dataset is CIFAR-10.

$\lambda_{training}$	Natural	FGSM	PGD-20	CW_∞
$\lambda = 0.5$	86.01	59.16	58.58	84.90
$\lambda = 1$	86.11	60.46	60.88	85.62
$\lambda = 2$	**86.19**	**60.84**	**62.18**	**86.62**
$\lambda = 5$	85.47	61.10	58.86	81.99
$\lambda = 10$	84.50	59.27	53.59	76.37
$\lambda = 20$	9.99	9.99	9.99	9.99

Table 10
Robustness (%) of FGAA_SAT and CAS_SAT under different attack targets. The left side of the "/" symbol refers to attacking the final layer of the trained robust model, while the right side of the "/" symbol indicates attacks directed at the specific FGAA/CAS layers.

Defenses	FGSM	PGD-20	CW_∞
CAS_SAT	60.07/60.18	46.31/44.75	50.63/47.96
FGAA_SAT	**60.84**/59.45	**62.18/58.82**	**86.6/86.09**

Table 11
Robustness (%) of VGG16 models with and without the FGAA module at the best and last checkpoints obtained through AT on CIFAR-10.

last_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	76.81	58.05	46.70	45.85
FGAA_SAT	**81.99**	54.00	42.54	**53.04**
Trades	79.03	57.17	45.79	45.52
FGAA_Trades	**81.67**	**59.53**	**53.24**	**68.17**
Mart	71.29	57.39	49.56	45.21
FGAA_Mart	**82.02**	56.64	45.70	**51.60**
best_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	74.05	57.43	48.14	46.59
FGAA_SAT	**78.90**	57.25	**52.04**	**65.45**
Trades	78.92	56.87	45.98	46.29
FGAA_Trades	**78.34**	**58.45**	**55.37**	**69.16**
Mart	71.61	57.84	50.21	45.09
FGAA_Mart	**77.53**	57.10	**51.31**	**59.50**

Table 12
Robustness (%) of WideResnet34-10 models with and without the FGAA module at the best and last checkpoints obtained through AT on CIFAR-10.

last_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	85.94	62.14	48.83	52.26
FGAA_SAT	84.65	62.09	**60.93**	**78.44**
Trades	86.97	62.11	50.34	54.70
FGAA_Trades	84.73	**67.03**	**54.81**	**68.01**
Mart	85.97	63.27	52.98	51.95
FGAA_Mart	**86.43**	**66.01**	**53.24**	**74.44**
best_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	86.81	65.70	53.20	53.26
FGAA_SAT	85.06	61.07	**60.47**	**79.44**
Trades	85.90	61.86	50.19	54.82
FGAA_Trades	84.37	**67.16**	**56.98**	**67.73**
Mart	83.27	63.31	53.46	54.11
FGAA_Mart	**83.49**	**66.41**	**57.84**	**73.81**

models like WideResNet-34-10 is not as pronounced as for VGG16 and ResNet18. However, the FGAA strategy consistently offers substantial improvements against CW_∞ attacks across all models and datasets. This is because FGAA leverages labeling information to guide the alignment of activation features, increasing prediction margins and thus achieving the greatest success in defending against margin-based CW_∞ attacks.

4.7.3. FGAA_ResNet18 results on Fashion-MNIST

In addition to validating the FGAA training strategy on CIFAR-10 and SVHN, we further conducted experiments on Fashion-MNIST. Table 13 lists the training parameters used for the experiments on Fashion-MNIST. The results are shown in Table 14. Overall, across all models and datasets, adversarial training Mart results in the greatest loss of prediction accuracy on natural examples. However, our FGAA training strategy consistently compensates for this loss. On the simpler Fashion-MNIST dataset, models have difficulty creating stronger adversarial

Table 13
Hyperparameters used for training FGAA_ResNet18 on Fashion-MNIST.

Parameters	FGAA_SAT	FGAA_Trades	FGAA_Mart
batch_size	128	128	128
epoch	200	100	120
lr	e−1	2e−1	e−1
weight_decay	$2e^{-4}$	$5e^{-4}$	$2e^{-4}$
FGAA_beta	2	2	2
num_steps	10	10	10
epsilon	8/255	0.031	0.031
alpha	2/255	0.007	0.007
beta	–	4.0	5.0

Table 14
Robustness (%) of ResNet18 with and without the FGAA module at the best and last checkpoints obtained through AT on Fashion-MNIST.

last_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	92.00	88.26	86.01	86.04
FGAA_SAT	**92.32**	**88.08**	**87.02**	**88.28**
Trades	92.60	89.79	88.57	88.54
FGAA_Trades	92.02	89.39	88.22	**89.60**
Mart	87.17	84.47	83.32	83.10
FGAA_Mart	**92.68**	**87.76**	**87.44**	**91.82**
best_checkpoint	Natural	FGSM	PGD-20	CW_∞
SAT	92.44	89.90	88.85	88.80
FGAA_SAT	92.13	89.60	**89.05**	**90.39**
Trades	92.52	89.94	88.58	88.60
FGAA_Trades	92.07	89.46	**88.66**	**89.90**
Mart	87.31	84.78	83.78	83.56
FGAA_Mart	**92.55**	**88.10**	**88.01**	**91.74**

examples, as evidenced by the adversarial robustness accuracy almost always exceeding 80%. Even in this scenario, where the potential for improvement is limited, adding FGAA still enhances the model's robustness. For instance, on the Fashion-MNIST dataset, FGAA provides the most significant improvement for Mart, with a maximum increase of 8.72% in robustness accuracy.

5. Conclusion

There are significant deviations in the feature activation distributions of neural networks for adversarial versus natural examples, whether analyzed from a spatial or channel perspective. Although AT can reduce these discrepancies, it often does so at the cost of natural example accuracy, which is especially problematic in security-critical AIoT applications where both robustness and precision are paramount. To address this challenge, we propose the FGAA module, which aligns feature activations at the single-feature level. By integrating the FGAA strategy into existing defense methods, we can significantly reduce the activation discrepancies between adversarial and natural examples, thereby enhancing overall model robustness—a crucial requirement for secure and privacy-preserving AIoT-enabled smart societies. Our study contributes to the development of deep learning techniques for security-critical domains, particularly in the context of smart infrastructures where the reliable processing of sensitive data is essential. Through this research, we aim to provide new insights and solutions for the security and privacy challenges in AI-driven IoT-enabled smart societies, thereby fostering further advancements in this field.

Nevertheless, FGAA is not without limitations. First, while our module suppresses redundant activations, it does not eliminate them, leaving room for further model compression and optimization. Second, although our empirical results and the OR metric offer strong evidence of FGAA's effectiveness, a formal theoretical framework remains lacking. In future work, we will explore margin-based generalization bounds and feature-manifold alignment theories to provide a rigorous foundation for FGAA and to guide the development of even more efficient and provably robust defenses. Beyond theory, we will also validate FGAA in AIoT-specific settings such as edge vision sensors, TinyML models, and federated learning, where resource constraints and adversarial threats intersect, to further assess practical robustness and deployment feasibility.

CRediT authorship contribution statement

Wenxin Kuang: Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Fengxiao Tang:** Supervision, Project administration, Conceptualization. **Jiayang Liu:** Writing – review & editing. **Yupeng Hu:** Supervision, Resources, Project administration, Funding acquisition. **Keqin Li:** Supervision.

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.

Acknowledgments

This work was supported by the Xiangjiang Laboratory Science and Technology Project (25XJ03013); the YueLuShan Center Industrial Innovation (2024YCII0110); the Hunan Science and Technology Innovation Leading Talents Project under Grant No. 2021RC4019.

References

[1] Risto Miikkulainen, Jason Liang, Elliot Meyerson, Aditya Rawal, Dan Fink, Olivier Francon, Bala Raju, Hormoz Shahrzad, Arshak Navruzyan, Nigel Duffy, et al., Evolving deep neural networks, in: Artificial Intelligence in the Age of Neural Networks and Brain Computing, Elsevier, 2024, pp. 269–287.
[2] Yupeng Hu, Wenxin Kuang, Zheng Qin, Kenli Li, Jiliang Zhang, Yansong Gao, Wenjia Li, Keqin Li, Artificial intelligence security: Threats and countermeasures, ACM Comput. Surv. 55 (1) (2021).
[3] Qin Liu, Yu Peng, Qian Xu, Hongbo Jiang, Jie Wu, Tian Wang, Tao Peng, Guojun Wang, Shaobo Zhang, MARS: Enabling verifiable range-aggregate queries in multi-source environments, IEEE Trans. Dependable Secur. Comput. 21 (4) (2023) 1994–2011.
[4] Shaobo Zhang, Yimeng Pan, Qin Liu, Zheng Yan, Kim-Kwang Raymond Choo, Guojun Wang, Backdoor attacks and defenses targeting multi-domain ai models: A comprehensive review, ACM Comput. Surv. 57 (4) (2024) 1–35.
[5] Shulan Wang, Qin Liu, Yang Xu, Hongbo Jiang, Jie Wu, Tian Wang, Tao Peng, Guojun Wang, Protecting inference privacy with accuracy improvement in mobile-cloud deep learning, IEEE Trans. Mob. Comput. 23 (6) (2023) 6522–6537.
[6] Kashi Sai Prasad, P Udayakumar, E Laxmi Lydia, Mohammed Altaf Ahmed, Mohamad Khairi Ishak, Faten Khalid Karim, Samih M Mostafa, A two-tier optimization strategy for feature selection in robust adversarial attack mitigation on internet of things network security, Sci. Rep. 15 (1) (2025) 2235.
[7] Hassan Khazane, Mohammed Ridouani, Fatima Salahdine, Naima Kaabouch, A holistic review of machine learning adversarial attacks in IoT networks, Futur. Internet 16 (1) (2024) 32.
[8] Wenxin Kuang, Qizhuang Liang, Peng Sun, Wei Fu, Qiao Hu, Yupeng Hu, Unveiling the pruning risks on privacy vulnerabilities of deep neural networks, in: ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2025, pp. 1–5.
[9] Huafeng Kuang, Hong Liu, Xianming Lin, Rongrong Ji, Defense against adversarial attacks using topology aligning adversarial training, IEEE Trans. Inf. Forensics Secur. (2024).
[10] Yuka Ogino, Kazuya Kakizaki, Takahiro Toizumi, Atsushi Ito, Outsmarting biometric imposters: Enhancing iris-recognition system security through physical adversarial example generation and PAD fine-tuning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 1451–1461.
[11] Qian Li, Yuxiao Hu, Yinpeng Dong, Dongxiao Zhang, Yuntian Chen, Focus on hiders: Exploring hidden threats for enhancing adversarial training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 24442–24451.
[12] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, Adrian Vladu, Towards deep learning models resistant to adversarial attacks, in: International Conference on Learning Representations, 2018, URL https://openreview.net/forum?id=rJzIBfZAb.

[13] Jinghui Chen, Yu Cheng, Zhe Gan, Quanquan Gu, Jingjing Liu, Efficient robust training via backward smoothing, Proc. the AAAI Conf. Artif. Intell. 36 (6) (2022) 6222–6230, URL https://ojs.aaai.org/index.php/AAAI/article/view/20571.

[14] Francesco Croce, Matthias Hein, Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, in: Hal Daumé III, Aarti Singh (Eds.), Proceedings of the 37th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 2206–2216, URL https://proceedings.mlr.press/v119/croce20b.html.

[15] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, Aleksander Madry, Robustness may be at odds with accuracy, 2018, URL https://arxiv.org/abs/1805.12152.

[16] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, Aleksander Madry, Adversarial examples are not bugs, they are features, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019, URL https://proceedings.neurips.cc/paper/2019/file/e2c420d928d4bf8ce0ff2ec19b371514-Paper.pdf.

[17] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, Kaiming He, Feature denoising for improving adversarial robustness, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 501–509.

[18] Yang Bai, Yuyuan Zeng, Yong Jiang, Shu-Tao Xia, Xingjun Ma, Yisen Wang, Improving adversarial robustness via channel-wise activation suppressing, in: International Conference on Learning Representations, 2021, URL https://openreview.net/forum?id=zQTezqCCtNx.

[19] Hanshu Yan, Jingfeng Zhang, Gang Niu, Jiashi Feng, Vincent Y.F. Tan, Masashi Sugiyama, CIFS: improving adversarial robustness of CNNs via channel-wise importance-based feature selection, 2021, CoRR, abs/2102.05311, arXiv:2102.05311, URL https://arxiv.org/abs/2102.05311.

[20] Yize Li, Pu Zhao, Ruyi Ding, Tong Zhou, Yunsi Fei, Xiaolin Xu, Xue Lin, Neural architecture search for adversarial robustness via learnable pruning, Front. High Perform. Comput. 2 (2024) 1301384.

[21] Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, Anima Anandkumar, Stochastic activation pruning for robust adversarial defense, 2018, CoRR, abs/1803.01442, arXiv:1803.01442, URL http://arxiv.org/abs/1803.01442.

[22] Jianwei Li, Qi Lei, Wei Cheng, Dongkuan Xu, Towards robust pruning: An adaptive knowledge-retention pruning strategy for language models, 2023, arXiv preprint arXiv:2310.13191.

[23] Hallgrimur Thorsteinsson, Valdemar J Henriksen, Tong Chen, Raghavendra Selvan, Adversarial fine-tuning of compressed neural networks for joint improvement of robustness and efficiency, 2024, arXiv preprint arXiv:2403.09441.

[24] Vikash Sehwag, Shiqi Wang, Prateek Mittal, Suman Jana, Hydra: Pruning adversarially robust neural networks, Adv. Neural Inf. Process. Syst. 33 (2020) 19655–19666.

[25] Yulun Wu, Yanming Guo, Dongmei Chen, Tianyuan Yu, Huaxin Xiao, Yuanhao Guo, Liang Bai, Boosting adversarial robustness via feature refinement, suppression, and alignment, Complex Intell. Syst. 10 (3) (2024) 3213–3233.

[26] Bingzhi Chen, Ruihan Liu, Yishu Liu, Xiaozhao Fang, Jiahui Pan, Guangming Lu, Zheng Zhang, Stay focused is all you need for adversarial robustness, in: Proceedings of the 32nd ACM International Conference on Multimedia, 2024, pp. 6482–6491.

[27] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016, http://www.deeplearningbook.org.

[28] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, Michael I. Jordan, Theoretically principled trade-off between robustness and accuracy, in: International Conference on Machine Learning, 2019.

[29] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, Quanquan Gu, Improving adversarial robustness requires revisiting misclassified examples, in: ICLR, 2020.

[30] Alex Krizhevsky, Learning Multiple Layers of Features from Tiny Images, Technical Report, University of Toronto, 2009.

[31] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y Ng, Reading digits in natural images with unsupervised feature learning, in: NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011.

[32] Han Xiao, Kashif Rasul, Roland Vollgraf, Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms, 2017, arXiv preprint arXiv:1708.07747.

[33] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy, Explaining and harnessing adversarial examples, 2014, URL https://arxiv.org/abs/1412.6572.

[34] Nicholas Carlini, David A. Wagner, Towards evaluating the robustness of neural networks, 2016, CoRR, abs/1608.04644, arXiv:1608.04644, URL http://arxiv.org/abs/1608.04644.

[35] Florian Tramer, Nicholas Carlini, Wieland Brendel, Aleksander Madry, On adaptive attacks to adversarial example defenses, Adv. Neural Inf. Process. Syst. 33 (2020) 1633–1645.

[36] Leslie Rice, Eric Wong, J. Zico Kolter, Overfitting in adversarially robust deep learning, 2020, CoRR, abs/2002.11569, arXiv:2002.11569, URL https://arxiv.org/abs/2002.11569.

Wenxin Kuang received her M.S. degree in Software Engineering from Central South University, China, in 2018. She is currently pursuing a Ph.D. degree in Computer Science and Technology at Hunan University. Her doctoral research focuses on adversarial robustness and security in deep learning systems, particularly for AIoT applications. Her research interests include artificial intelligence security, adversarial attacks and defenses, malware detection, erasure coding, and storage system security.

Fengxiao Tang (Senior Member, IEEE) is a full professor in the School of Computer Science and Engineering of Central South University. He has been an Assistant Professor from 2019 to 2020 and an Associate Professor from 2020 to 2021 at the Graduate School of Information Sciences (GSIS) of Tohoku University. His research interests are unmanned aerial vehicles system, IoT security, game theory optimization, network traffic control and machine learning algorithm. He was a recipient of the prestigious Dean's and President's Awards from Tohoku University in 2019, and several best paper awards at conferences including ICNIDC 2018/2023, GLOBECOM 2017/2018. He was also a recipient of the prestigious Funai Research Award in 2020, IEEE ComSoc Asia-Pacific (AP) Outstanding Paper Award in 2020 and IEEE ComSoc AP Outstanding Young Researcher Award in 2021.

Jiayang Liu received his B.S. degree in 2015 from Dalian Maritime University (DMU) and Ph.D. degree in 2020 from University of Science and Technology of China (USTC). From 2020 to 2022, he was a postdoctoral researcher at the RIKEN Center for Advanced Intelligence Project. He then worked as a research fellow at National University of Singapore (NUS) from 2022 to 2024 and at Nanyang Technological University (NTU) since 2024. He is currently a researcher at Institute of Science Tokyo. His research interests include AI security and information hiding.

Yupeng Hu received the MS and Ph.D. degrees in computer science from Hunan University, China, in 2005 and 2008, respectively. He is an IEEE/ACM Senior Member. He is currently a Professor with the College of Cyberspace Security, and College of Computer Science and Electronic Engineering, Hunan University, China. He was the Dean of the Department of Cyberspace Security and now is the Director of Modern Engineering Training Center (College of Innovation and Entrepreneurship). His main research interest focuses on software and hardware integrated system security and reliability, including AI system security, storage system security, and PUF/CPU security and reliability.

Keqin Li is a SUNY Distinguished Professor of computer science with the State University of New York. He is also a Distinguished Professor with Hunan University, China. His current research interests include cloud computing, fog computing and mobile edge computing, energy-efficient computing and communication, embedded systems and cyber–physical systems, heterogeneous computing systems, big data computing, high-performance computing, CPU–GPU hybrid and cooperative computing, computer architectures and systems, computer networking, machine learning, intelligent and soft computing. He has authored or coauthored more than 770 journal articles, book chapters, and refereed conference papers, and has received several best paper awards. He has chaired many international conferences. He is currently an associate editor of the ACM Computing Surveys and the CCF Transactions on High Performance Computing.