International Journal of Technology

http://ijtech.eng.ui.ac.id

Research Article

Adaptive Gradient Shielding for Consistent Defense Against Adversarial Attacks in Deep Learning

Qais Saif Qassim[1*], Wilfred Blessing Nesaian Reginal[1], Dilwar Islam Mazumder[2], Samuel Brilly Sangeetha[3], Yuvaraj Natarajan[4], Arshath Raja Rajan[4]

[1]College of Computing and Information Sciences, University of Technology and Applied Sciences– Ibri, 516, Sultanate of Oman

[2]Center for Preparatory Studies, University of Technology and Applied Sciences – Ibri, 516, Sultanate of Oman

[3]Department of Computer Science and Engineering, IES College of Engineering, Thrissur, Kerala, 680551, India

[4]ICT Academy of Tamil Nadu, IIT Madras Research Park, Chennai, 600113, India

[*]Corresponding author: qais.aljanabi@utas.edu.om; Tel.: 0096895217220

Abstract: Deep learning models are highly vulnerable to adversarial perturbations, posing serious risks in critical domains such as healthcare, autonomous systems, and cybersecurity. This has motivated the need for robust defenses that maintain reliability without significant computational overhead. Current defense strategies fail to provide a unified solution for gradient-based and gradient-free adversarial attacks. Many approaches suffer from gradient masking or accuracy degradation during clean inference. Methods that can adaptively stabilize model responses under diverse and evolving adversarial conditions have a significant defense gap. This study introduces AGS, a lightweight defense that combines perturbation-aware feature smoothing with dynamic gradient modulation. Using an adaptive auxiliary controller, AGS detects locally sensitive features and selectively regulates gradient flow, improving robust generalization without full adversarial training. The proposed method is evaluated on CIFAR-10 and ImageNet-Subset under FGSM, PGD-20, and BIM attacks. AGS significantly improves robustness while preserving clean accuracy. On CIFAR-10, the PGD-20 accuracy increased from 42.8% to 71.4% with only a 0.4% drop in clean accuracy, and the FGSM accuracy improved from 61.5% to 84.7%. On the ImageNet-Subset, the PGD-20 accuracy increased from 18.6% to 39.3%. The auxiliary controller adds only 3.2% computational overhead, which is lower than that of standard adversarial training, demonstrating that AGS is an efficient and adaptive defense for varying adversarial conditions.

Keywords: Adaptive shielding; Adversarial robustness; Deep learning security; Gradient modulation; Perturbation resilience

1. Introduction

Deep learning models have revolutionized multiple domains by providing state-of-the-art performance in tasks ranging from image classification and natural language understanding to autonomous navigation and healthcare diagnostics (Hu et al., 2024; Wang et al., 2025; Wei & Liu, 2022). The ability of deep neural networks (DNNs) to learn hierarchical feature representations has enabled remarkable generalization capabilities, often surpassing human-level accuracy in well-defined tasks (Al-Andoli et al., 2024). In image-based applications, convolutional neural networks (CNNs) extract spatial hierarchies that capture complex patterns and enable precise predictions (Zhou et al., 2023). Similarly, recurrent architectures, attention mechanisms, and transformer-based networks allow the modeling of sequential and contextual dependencies in text and time-series data (Muoka et al., 2023; Roy & Dasgupta, 2025). The continued growth in computational power and availability of large datasets has accelerated the adoption of deep learning across industry and research sectors (Jhajharia & Shrivastavaa, 2024). Despite these

https://doi.org/10.14716/ijtech.v17i4.8367
Received January 2026; Revised March 2026; Accepted May 2026

impressive advancements, deep learning models remain highly vulnerable to adversarial perturbations (Zhang & Wang, 2026). Even minor, imperceptible modifications to input data can lead to drastic misclassifications, exposing a critical reliability gap in safety-sensitive applications (Shayea et al., 2025). For instance, in healthcare imaging, an imperceptible perturbation can cause a model to misdiagnose medical conditions, while in autonomous driving, it can misinterpret traffic signs or pedestrian presence, potentially leading to hazardous outcomes (Wang et al., 2023; Zhao et al., 2022). Similarly, cybersecurity analytics that rely on deep learning for intrusion detection or fraud monitoring are susceptible to adversarially crafted inputs that bypass detection mechanisms (Muoka et al., 2023). The vulnerability of deep models is not limited to gradient-based attacks; gradient-free attacks and black-box scenarios also threaten model integrity (Li et al., 2025; Xu et al., 2024). Additionally, many defense mechanisms introduce excessive computational overhead or degrade the performance of clean data, limiting their practical applicability (Chelliah et al., 2023). The key challenges in adversarial defense stem from the conflicting goals of robustness and model fidelity. Many approaches attempt to harden the model against specific attack types, often leading to overfitting to known adversaries and failing against adaptive or unseen perturbations (Shayea et al., 2025; Zhao et al., 2022). For example, gradient masking techniques reduce sensitivity to certain attack vectors but inadvertently introduce blind spots that can be exploited by adversaries (Wang et al., 2023). Although effective against particular attacks, adversarial training demands significant computational resources and often reduces the accuracy of clean inference (Muoka et al., 2023; Wang et al., 2024). Further, gradient-based and gradient-free attacks exploit distinct vulnerabilities in models, and no unified mechanism has consistently addressed both attack categories (Li et al., 2025). Thus, the need for a defense method that adapts dynamically to varying adversarial conditions while maintaining efficiency and high clean accuracy is evident. The problem addressed in this research emerges from the absence of a robust, adaptive, and computationally efficient defense framework that systematically shields vulnerable feature channels while maintaining model generalization (Puttagunta et al., 2023). The analysis further emphasizes the need for the proposed solution. First, current defenses frequently rely on static mechanisms that cannot adapt to evolving attack strategies, exposing models to new perturbation patterns. Second, gradient modulation methods often lack fine-grained control, which either suppresses learning excessively or fails to mitigate adversarial influence.

Content for adding in related works ShieldNet was introduced by (Manzoor et al., 2026) as a convolutional neural network (CNN) that is resilient against adversarial attacks through the use of adversarial training, gradient smoothing regularization, and consistency loss techniques. The system uses an attack-aware weight mechanism and was tested using strong benchmarking techniques such as AutoAttack and PGD and proved better than the current defensive systems. The only drawback is that it concentrates more on adversarial defense in image biometrics, ignoring other issues such as computational complexity.

The primary objective of this research is to design and evaluate Adaptive Gradient Shielding (AGS), a novel defense mechanism that provides a unified and adaptive approach to adversarial robustness in deep learning models. Specifically, AGS aims to (i) identify feature channels that are highly sensitive to perturbations, (ii) stabilize feature representations by applying perturbation-aware smoothing, (iii) dynamically modulate gradients during backpropagation to mitigate adversarial influence, and (iv) integrate clean and perturbed feature statistics to enhance robust generalization.

The novelty of AGS is its adaptive and feature-centric approach. Unlike conventional defenses that rely solely on adversarial training or static gradient modulation, AGS actively monitors the sensitivity of feature channels during training and adjusts shielding intensity in real time using a lightweight auxiliary controller. This mechanism allows the model to respond dynamically to perturbations of varying strength and structure, ensuring that the gradient flow is only modulated where necessary.

The contributions of this research are twofold. First, adaptive gradient shielding, a novel

algorithm that combines perturbation-aware feature smoothing with dynamic gradient modulation, is introduced to improve robustness against various adversarial attacks. The method identifies vulnerable feature channels and selectively attenuates gradients flowing through these channels, guided by an auxiliary controller that adjusts shielding intensity in real time. The proposed framework is consistently evaluated using the MNIST, CIFAR-10, and CIFAR-100 datasets under standardized adversarial attack settings.

2. Literature Survey

Adversarial attacks have emerged as a critical challenge in deep learning models due to their ability to exploit subtle input perturbations that are imperceptible to humans but significantly degrade model performance (Shan et al., 2025). Gradient-based attacks, such as Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD), manipulate input data along the loss function gradient to maximize prediction error (Jiang et al., 2025). These attacks have shown that deep models possess inherent sensitivity to small perturbations in high-dimensional feature spaces despite their impressive performance on clean data (Yinusa & Faezipour, 2025). Black-box and gradient-free attacks expose the fragility of models by leveraging surrogate models or iterative optimization to bypass gradient masking techniques (Xiangfei & Qingchen, 2025). This vulnerability has prompted extensive research into defense mechanisms capable of providing consistent robustness without severely impacting clean accuracy.

Existing defense mechanisms can be broadly categorized as adversarial training, gradient masking, input preprocessing, and architectural modifications (Kim et al., 2024). Adversarial training, where models are trained on adversarially perturbed samples to improve resilience, has become a widely adopted approach (Laila, 2025). In particular, PGD-based adversarial training has shown strong empirical robustness but often incurs significant computational cost and can reduce performance on clean inputs (Elabd, 2025). Gradient masking techniques obscure gradient information to prevent effective adversarial optimization (Anjum & Nnaji, 2025). Although these approaches can deter certain attacks, they often fail against adaptive adversaries that approximate or circumvent the masked gradients. Input preprocessing methods, including feature denoising, input transformations, and randomization, have shown moderate improvements in robustness but typically do not generalize well across attack types (Costa et al., 2024). Architectural modifications, such as the incorporation of robust activation functions or specialized layers, offer promising directions but are often limited by complexity and implementation constraints (Alajaji, 2025).

Recognizing the limitations of static defenses, several studies have proposed adaptive mechanisms that respond dynamically to adversarial perturbations. These methods aim to modulate the model's internal representations or gradients based on observed perturbations (Guo et al., 2026). For example, dynamic feature denoising layers adjust the extent of smoothing according to the estimated noise in each channel, thereby preserving salient features while mitigating adversarial influence (Rahman et al., 2025). Similarly, attention-based mechanisms prioritize robust features over vulnerable channels, enhancing model stability under attack (Abbes et al., 2025). These approaches highlight the importance of local feature sensitivity as a critical factor in robust learning. However, most existing adaptive defenses focus exclusively on gradient-based attacks or require extensive additional computation, limiting their applicability in real-time scenarios.

Gradient shielding represents a promising line of research aimed at directly controlling the gradient flow in vulnerable feature channels (He et al., 2023). By attenuating excessive gradients caused by adversarial perturbations, models can maintain stable updates during training and avoid overfitting to adversarial examples. Previous works in this domain have employed static or layer-wise gradient scaling, which has shown improvements against specific attacks but lacks adaptability when the perturbation strength varies (Ness, 2025). Gradient shielding alone does not integrate information from clean samples, which can compromise generalization to unseen inputs. These observations motivate the development of mechanisms that combine gradient

modulation with perturbation-aware feature smoothing to achieve both robustness and accuracy.

Recent research emphasizes the benefits of fusing clean and adversarial feature statistics to guide robust learning (Abasi et al., 2025). Models can learn representations that are less sensitive to perturbations while retaining critical task-relevant information by capturing distributional characteristics from both input types. This approach has been applied in feature denoising networks and adversarial feature alignment frameworks, demonstrating improved robustness across multiple attack types. However, most implementations are static or rely heavily on adversarial training, limiting efficiency and adaptability. A lightweight controller that dynamically adjusts the influence of clean and perturbed statistics remains largely unexplored.

Despite the breadth of defense strategies, several persistent challenges remain. First, models often experience a trade-off between clean accuracy and adversarial robustness, with many defenses achieving gains at the cost of normal performance (Kim et al., 2024; Shan et al., 2025). Second, adaptive and dynamic defenses frequently rely on complex controllers or additional networks, introducing computational overhead that may be prohibitive for large-scale deployment (Abbes et al., 2025; Guo et al., 2026). Third, gradient masking and static shielding approaches are vulnerable to adaptive or black-box attacks because adversaries can estimate or bypass the hidden gradients (Anjum & Nnaji, 2025; He et al., 2023). Finally, the lack of a unified framework capable of addressing both gradient-based and gradient-free attacks limits the practical applicability of existing methods (Jiang et al., 2025; Xiangfei & Qingchen, 2025). These challenges highlight the need for efficient, feature-sensitive, and adaptive defense mechanisms that balance robustness, accuracy, and computational efficiency.

The analysis of the existing literature reveals several critical research gaps. First, most defenses fail to provide real-time adaptability to evolving adversarial conditions, which is essential for deployment in dynamic and safety-critical environments (Abbes et al., 2025; Guo et al., 2026). Second, gradient modulation techniques lack fine-grained control, leading to either insufficient shielding or excessive suppression of legitimate learning signals (He et al., 2023). Third, the integration of clean and perturbed feature statistics remains underexplored, limiting the potential for generalized robust representations (Abasi et al., 2025). Fourth, the computational cost of existing adaptive methods is often high, limiting their usability in resource-limited scenarios, such as edge computing or autonomous navigation (Rahman et al., 2025). These gaps motivate the development of an adaptive gradient shielding (AGS) framework that addresses both feature-level sensitivity and dynamic gradient modulation in a computationally efficient manner.

Recent studies emphasize the importance of rigorous benchmarking across multiple datasets and attack types to comprehensively evaluate defense mechanisms. CIFAR-10 and ImageNet-Subset have emerged as standard benchmarks due to their varying complexity and suitability for evaluating both gradient-based and gradient-free attacks (Laila, 2025; Shan et al., 2025). Benchmark studies show that models evaluated under a single attack type often overestimate robustness, whereas multi-attack evaluations reveal significant vulnerabilities (Costa et al., 2024; Elabd, 2025). This trend underscores the necessity for defense methods that have consistently improved across diverse perturbation types and have maintained clean accuracy under normal inference conditions.

The practical importance of lightweight adaptive mechanisms that minimize computational overhead while providing robustness is increasingly highlighted in the literature (Guo et al., 2026; Rahman et al., 2025). Lightweight auxiliary controllers, feature-specific gradient modulation, and perturbation-aware smoothing layers allow models to adjust dynamically to varying attack strengths without requiring extensive retraining or additional parameters (Abasi et al., 2025; Abbes et al., 2025). The integration of such mechanisms into a unified framework can provide a scalable solution for real-world applications, such as autonomous vehicles, medical imaging, and real-time security analytics.

The literature establishes a clear trajectory for advancing deep learning adversarial defenses. Traditional approaches, including adversarial training, gradient masking, and input preprocess-

ing, provide foundational strategies but exhibit notable limitations in adaptability, computational efficiency, and generalization (Elabd, 2025; Jiang et al., 2025; Kim et al., 2024; Laila, 2025; Shan et al., 2025; Xiangfei & Qingchen, 2025; Yinusa & Faezipour, 2025). Emerging adaptive mechanisms and gradient shielding techniques highlight the potential of feature-sensitive and dynamic defenses; however, they are limited by the partial coverage of attack types and static implementations (Abbes et al., 2025; Guo et al., 2026; He et al., 2023; Rahman et al., 2025). Finally, the integration of clean and perturbed feature statistics presents a promising avenue to enhance robustness while preserving model performance (Abasi et al., 2025). Collectively, these insights justify the development of AGS as a unified, adaptive, and computationally efficient defense mechanism that addresses existing limitations while maintaining high clean accuracy and robustness across multiple adversarial scenarios.

3. Proposed Adaptive Gradient Shielding (AGS)

The proposed adaptive gradient shielding (AGS) provides a robust defense mechanism for deep learning models against gradient-based and gradient-free adversarial attacks. The proposed method dynamically modulates gradients flowing through vulnerable feature channels while preserving the accuracy of clean inference. It does not rely solely on adversarial training but integrates clean and perturbed feature statistics to enhance robust generalization.

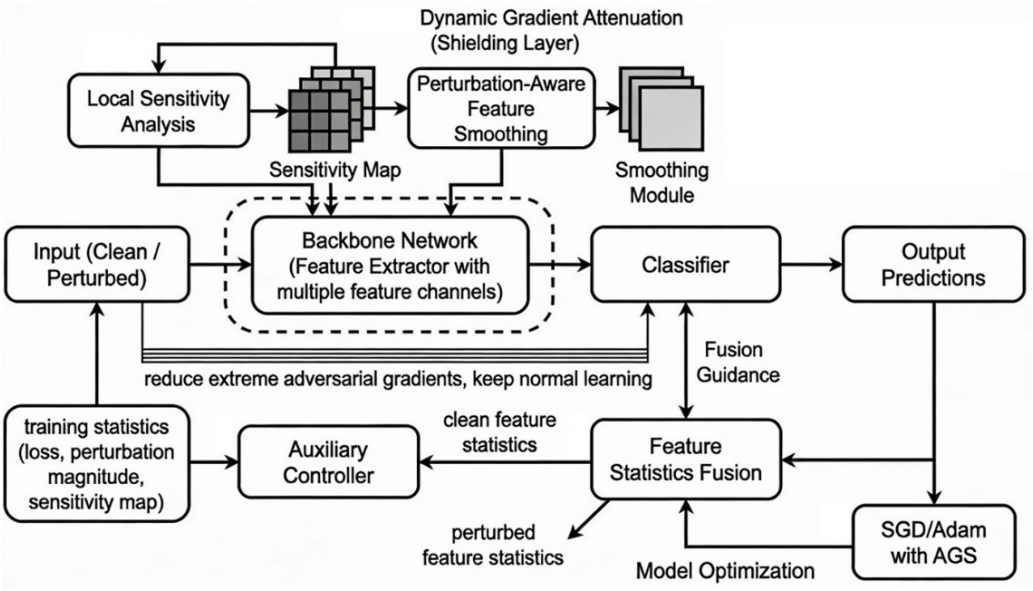

Figure 1 Proposed AGS architecture

3.1 Analysis of Local Sensitivity in Adaptive Gradient Shielding

Local Sensitivity Analysis (LSA) serves as the foundational mechanism that enables adaptive gradient shielding to identify and regulate vulnerable feature channels within a deep neural network. The central objective of LSA is to quantify the influence of small input perturbations on intermediate feature representations and their associated gradients. Instead of treating the network as a monolithic structure, LSA decomposes robustness analysis at the feature-channel level, allowing precise and selective gradient regulation. This localized perspective is essential because adversarial perturbations rarely uniformly affect all features; rather, they amplify sensitivity in specific channels that disproportionately influence the final prediction.

Deep neural networks learn hierarchical feature representations in which the early layers capture low-level patterns and the deeper layers encode task-specific abstractions. However, adversarial attacks exploit the fact that certain feature channels exhibit disproportionately high gradient responses to small input perturbations. These channels act as amplification pathways

Algorithm 1: Adaptive Gradient Shielding (AGS) Training Framework

Input: Training dataset $\mathcal{D} = \{(x_i, y_i)\}$, backbone model (ResNet-18), epochs T, learning rate η
Output: Trained robust model parameters θ
Initialize model parameters θ of ResNet-18;
Initialize controller parameters ϕ;
Set hyperparameters $\alpha, \mu, \kappa, \psi_1, \psi_2, \beta, \zeta$;
for *epoch* $t = 1$ *to* T **do**
  `// (A) Forward Propagation`
  foreach *mini-batch* (x, y) **do**
    $F^l = \text{ResNet-18}(x; \theta)$;
    Generate adversarial sample $x^{adv} = x + \delta,\ \delta \sim$ PGD/FGSM constraint;
    Compute adversarial features $\widetilde{F}^l = \text{ResNet-18}(x^{adv}; \theta)$;
  `// (B) Local Sensitivity Analysis (LSA)`
  Compute channel-wise sensitivity $S_k^l = \left\| \partial \mathcal{L} / \partial F_k^l \right\|$;
  Apply temporal smoothing $\bar{S}_k^l(t) = \beta \bar{S}_k^l(t-1) + (1-\beta) S_k^l$;
  `// (C) Perturbation-Aware Feature Smoothing (PAFS)`
  Compute smoothed adversarial features $\widehat{F}_k^l = \mu F_k^l + (1-\mu)\widetilde{F}_k^l$;
  `// (D) Dynamic Gradient Attenuation (DGA)`
  Compute attenuation score $a_k^l = \psi_1 \bar{S}_k^l + \psi_2 \bar{\eta}_k^l$;
  Compute attenuation factor $\lambda_k^l = \sigma(\kappa a_k^l)$;
  Modify gradients $\widetilde{g}_k^l = (1 - \lambda_k^l) \cdot g_k^l$;
  Parameter update $\theta \leftarrow \theta - \eta \cdot \widetilde{g}_k^l$;
  `// (E) Feature Statistics Fusion (FSF)`
  Compute feature statistics $\mu_k^l, \sigma_k^l, \widetilde{\mu}_k^l, \widetilde{\sigma}_k^l$;
  Fuse statistics $\mu_k^{l,f} = \alpha \mu_k^l + (1-\alpha)\widetilde{\mu}_k^l; \quad \sigma_k^{l,f} = \alpha \sigma_k^l + (1-\alpha)\widetilde{\sigma}_k^l$;
  `// (F) Controller Update`
  Update controller parameters $\phi \leftarrow \phi - \eta \nabla_\phi \mathcal{L}_{AGS}$;
return optimized model parameters θ

through which adversarial noise propagates. LSA identifies such channels by measuring local changes in feature activations and gradients when the input is exposed to both clean and perturbed samples. The model obtains a sensitivity map that guides subsequent shielding operations by quantifying these changes.

3.1.1 Mathematical formulation of the feature sensitivity

A deep neural network is represented as a function

$$f(x; \theta) : \mathbb{R}^d \rightarrow \mathbb{R}^C \tag{1}$$

where x denotes the input sample, θ denotes the trainable parameters, and C denotes the number of output classes. Let $F^l(x) \in \mathbb{R}^{H \times W \times K}$ denote the feature tensor at layer l, where K represents the number of feature channels. For a given clean input x and its adversarial counterpart x^{adv}, generated using a bounded perturbation δ such that

$$x^{adv} = x + \delta, \quad \|\delta\|_p \leq \epsilon, \tag{2}$$

the LSA computes the per-channel feature deviation as follows:

$$\Delta F_k^l = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \left| F_k^l(x)_{i,j} - F_k^l(x^{adv})_{i,j} \right| \tag{3}$$

This formulation captures the average spatial deviation in channel k due to adversarial perturbation. Channels that exhibit higher ΔF_k^l values are more sensitive to input perturbations.

3.1.2 Gradient-based sensitivity measurement

Feature deviation alone does not fully characterize adversarial vulnerability. Therefore, LSA also uses the gradient sensitivity, which reflects how strongly each channel influences the loss function. Let $\mathcal{L}(f(x), y)$ denote the classification loss. The gradient sensitivity of channel k at layer l is computed as

$$G_k^l = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \left| \frac{\partial \mathcal{L}}{\partial F_k^l(i,j)} \right| \tag{4}$$

This measure captures the magnitude of the backpropagated gradients that flow through each channel. Channels with high G_k^l values indicate regions where small feature changes significantly alter the loss, making them prime targets for adversarial exploitation.

3.1.3 Composite local sensitivity score

The LSA combines the feature deviation and gradient magnitude into a unified sensitivity score. The composite sensitivity for channel k at layer l is defined as

$$S_k^l = \alpha \cdot \frac{\Delta F_k^l}{\max_k \Delta F_k^l} + (1 - \alpha) \cdot \frac{G_k^l}{\max_k G_k^l}, \tag{5}$$

where $\alpha \in [0, 1]$ controls the relative contribution of the feature deviation and gradient sensitivity. Normalization has demonstrated mathematical stability and comparability across channels. Channels with higher S_k^l values are identified as locally sensitive and are prioritized for gradient shielding. Table 1 illustrates the feature deviation and gradient sensitivity values computed for a convolutional layer with five channels. As shown in Table 1, Channel 3 exhibited the highest composite sensitivity score, indicating heightened vulnerability to adversarial perturbations. Moreover, the table shows that sensitivity does not arise solely from large gradients or large feature shifts but from their combined influence.

Table 1 Local sensitivity metrics for a convolutional layer

Channel Index	Feature Deviation ΔF_k	Gradient Magnitude G_k	Composite Sensitivity S_k
1	0.124	0.087	0.42
2	0.198	0.142	0.61
3	0.315	0.291	0.93
4	0.172	0.119	0.54
5	0.089	0.064	0.31

3.1.4 Threshold-based channel selection

To operationalize LSA, a sensitivity threshold τ is applied to select vulnerable channels:

$$\mathcal{V}^l = \{k \mid S_k^l \geq \tau\} \tag{6}$$

The threshold τ adapts dynamically during training based on the sensitivity score distribution. This adaptive selection prevents over-shielding and shows that only genuinely vulnerable channels are regulated.

3.1.5 Temporal stability of the sensitivity patterns

The values in Table 1 show that sensitivity does not arise solely from large gradients or large feature shifts but from their combined influence. Sensitivity patterns evolve as training progresses. Therefore, LSA uses temporal smoothing to avoid instability. The exponentially weighted moving average of the sensitivity scores is computed as follows:

$$\grave{S}_k^l(t) = \beta \cdot \grave{S}_k^l(t-1) + (1-\beta) \cdot S_k^l(t) \tag{7}$$

where β controls temporal smoothness. This formulation has shown that transient noise does not cause excessive shielding. Once the vulnerable channel set $\mathcal{V}^l$ has been identified, the LSA directly informs the gradient attenuation. For a sensitive channel k, the backpropagated gradient is modulated as follows:

$$\widetilde{g}_k^l = (1-\lambda_k) \cdot g_k^l \tag{8}$$

where $\lambda_k = \sigma(\gamma \cdot \grave{S}_k^l)$, $\sigma(\cdot)$ denotes the sigmoid function, and γ controls attenuation sharpness. Smooth, bounded gradient suppression rather than abrupt truncation has been shown in this formulation. The computational cost of LSA remains modest because the sensitivity calculations rely on the available forward and backward pass information. No additional forward passes or heavy auxiliary networks are required. Empirical observations indicate that LSA introduces minimal latency while providing substantial robustness gains, which aligns with the AGS design goals.

3.2 Perturbation-Aware Smoothing of Features in Adaptive Gradient Shielding

Perturbation-Aware Feature Smoothing (PAFS) functions as the second core mechanism of adaptive gradient shielding, operating directly on intermediate feature representations to suppress adversarial noise while preserving task-relevant semantic information. PAFS performs adaptive smoothing at the feature-channel level, unlike input-level smoothing or static denoising operations, guided by the local sensitivity patterns identified during training. The objective of PAFS is not to eliminate perturbations but to regulate their influence in a controlled and data-aware manner that maintains discriminative capacity.

3.2.1 Design rationale

Adversarial perturbations exploit the learned feature representations' high-frequency and unstable components. These perturbations propagate through the network layers and amplify the prediction errors. Traditional feature smoothing methods often apply uniform spatial filtering, which reduces both adversarial noise and useful signal. In contrast, PAFS introduces selective smoothing that adapts to perturbations' magnitude, direction, and locality. The smoothing strength varies across channels and spatial locations, ensuring that robust features remain intact while unstable activations are attenuated.

3.2.2 Feature representation framework

Let $F^l(x) \in \mathbb{R}^{H \times W \times K}$ denote the feature tensor at layer l for a clean input x, where H and W represent spatial dimensions and K represents the number of channels. For a perturbed input x^{adv}, the corresponding feature tensor is denoted as $F^l(x^{adv})$. PAFS operates on the perturbed feature tensor while referencing the clean feature statistics to ensure semantic consistency.

3.2.3 Local disturbance estimation

The first step in PAFS estimates the magnitude of the local perturbation at the feature level. The per-channel perturbation energy is defined as follows:

$$E_k^l = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(F_k^l(x^{adv})_{i,j} - F_k^l(x)_{i,j} \right)^2 \tag{9}$$

This formulation captures the average squared deviation caused by adversarial perturbations in channel k. Higher values of E_k^l indicate stronger perturbation influence. To ensure mathematical stability and comparability, the perturbation energy is normalized as follows:

$$\widehat{E}_k^l = \frac{E_k^l}{\max_k E_k^l + \epsilon}, \tag{10}$$

where ϵ is a small constant.

3.2.4 Adaptive kernel formulation for smoothing

PAFS applies smoothing through a channel-specific adaptive kernel. For each channel k, a smoothing coefficient η_k^l is derived as follows:

$$\eta_k^l = \sigma \left(\mu \cdot \widehat{E}_k^l \right) \tag{11}$$

where $\sigma(\cdot)$ denotes the sigmoid function and μ controls the sensitivity of smoothing to perturbation strength. The smoothed feature activation at spatial location (i, j) is defined as

$$\widetilde{F}_k^l(i,j) = (1 - \eta_k^l) \cdot F_k^l(x^{adv})_{i,j} + \eta_k^l \cdot \mathcal{S}(F_k^l(x^{adv}), i, j) \tag{12}$$

where $\mathcal{S}(\cdot)$ represents a localized spatial smoothing operator, such as weighted neighborhood averaging.

3.2.5 Spatially weighted smoothing operator

The smoothing operator $\mathcal{S}$ is defined as

$$\mathcal{S}(F_k^l, i, j) = \sum_{(u,v)\in\Omega} w_{u,v} \cdot F_k^l(u, v) \tag{13}$$

where Ω denotes a local spatial window and $w_{u,v}$ are normalized weights satisfying

$$\sum_{(u,v)\in\Omega} w_{u,v} = 1 \tag{14}$$

The weights adapt dynamically based on local variance to avoid over-smoothing edges or salient patterns. PAFS aligns smoothed features with clean feature distributions to preserve semantic integrity. The alignment loss is expressed as follows:

$$\mathcal{L}_{al} = \sum_{k=1}^{K} \left\| \mathbb{E}[F_k^l(x)] - \mathbb{E}[\widetilde{F}_k^l] \right\|_2^2 \tag{15}$$

This constraint shows that smoothing does not distort class-discriminative features learned from clean data. PAFS regulates feature variance to suppress adversarial amplification. The variance regularization term is defined as follows:

$$\mathcal{L}_{var} = \sum_{k=1}^{K} \max \left(0, \mathrm{Var}(\widetilde{F}_k^l) - \tau_v \right) \tag{16}$$

where τ_v denotes a variance threshold. This formulation penalizes excessive variability, which often correlates with adversarial noise (AN). The adversarial perturbation characteristics evolve during training. Therefore, PAFS uses the following temporal smoothing of coefficients:

$$\dot{\eta}_k^l(t) = \rho \cdot \dot{\eta}_k^l(t-1) + (1-\rho) \cdot \eta_k^l(t) \tag{17}$$

where ρ controls temporal stability. PAFS influences training dynamics by shaping the feature landscape before gradient computation. The loss gradient with respect to smoothed features becomes

$$\frac{\partial \mathcal{L}}{\partial F_k^l} = \left(1 - \dot{\eta}_k^l\right) \cdot \frac{\partial \mathcal{L}}{\partial \widetilde{F}_k^l} \tag{18}$$

which prevents the adversarial noise from inducing steep or unstable gradients. The computational overhead of PAFS remains modest because it leverages existing feature maps and local neighborhood operations. No additional forward passes are required. Empirical observations indicate that PAFS minimally contributes to inference latency while significantly enhancing robustness.

3.3 Dynamic Gradient Attenuation

Adversarial attacks primarily exploit the deep neural network gradient structure. Excessively large or sharply varying gradients amplify small perturbations, thereby allowing adversaries to misclassify. Static gradient clipping or uniform scaling methods often reduce this effect but introduce two major limitations. First, they indiscriminately suppress gradients, which hinders learning in robust channels. Second, they remain ineffective against adaptive attacks that dynamically alter the perturbation strength. DGA addresses these issues by applying a channel-wise, data-driven gradient modulation that evolves with the training dynamics and attack characteristics.

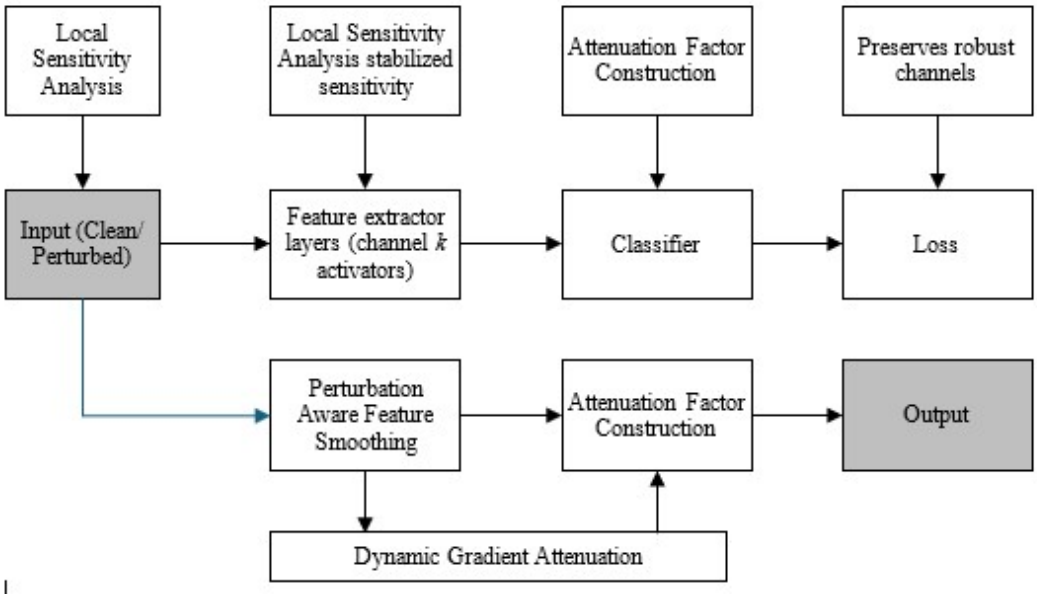

Figure 2 DGA architecture

3.3.1 Gradient representation and vulnerability context

Let $\mathcal{L}(f(x), y)$ denote the loss function for input x and label y. For a given layer l and channel k, the raw backpropagated gradient is expressed as follows:

$$g_k^l = \frac{\partial \mathcal{L}}{\partial F_k^l}, \tag{19}$$

where F_k^l denotes the feature activation of channel k at layer l. In adversarial settings, g_k^l often exhibits abnormally high magnitude or oscillatory behavior in sensitive channels identified through LSA.

3.3.2 Formulation of Dynamic Gradient Attenuation

The DGA modifies the raw gradient using a channel-specific attenuation factor $\lambda_k^l \in [0, 1]$. The attenuated gradient is defined as

$$\widetilde{g}_k^l = \left(1 - \lambda_k^l\right) \cdot g_k^l \tag{20}$$

Unlike static scaling, λ_k^l adapts dynamically based on the estimated vulnerability and perturbation context. Higher values of λ_k^l indicate stronger suppression of adversarial influence.

3.3.3 Attenuation factor construction

The attenuation factor is derived from a composite vulnerability signal that integrates sensitivity and perturbation cues. Let $\grave{S}_k^l$ denote the temporally stabilized sensitivity score from the local sensitivity analysis, and let $\grave{\eta}_k^l$ denote the stabilized smoothing coefficient from the perturbation-aware feature smoothing. The raw attenuation score is expressed as

$$a_k^l = \omega_1 \cdot \grave{S}_k^l + \omega_2 \cdot \grave{\eta}_k^l \tag{21}$$

where ω_1 and ω_2 are weighting coefficients that satisfy $\omega_1 + \omega_2 = 1$. The final attenuation factor is computed using the following bounded nonlinearity:

$$\lambda_k^l = \sigma\left(\kappa \cdot a_k^l\right) \tag{22}$$

where $\sigma(\cdot)$ denotes the sigmoid function and κ controls the sharpness of attenuation response.

3.4 Auxiliary Controller

The Auxiliary Controller Adjustment mechanism (Figure 3) governs how attenuation factors evolve during training. The controller is a lightweight neural module that receives vulnerability descriptors and outputs scaling parameters that regulate κ, ω_1, and ω_2. Its objective is to dynamically balance robustness and clean accuracy.

The controller function is denoted as

$$(\kappa_t, \omega_{1,t}, \omega_{2,t}) = \mathcal{C}_\phi(\mathbf{z}_t) \tag{23}$$

where $\mathbf{z}_t$ represents the controller input at iteration t, and ϕ denotes controller parameters.

3.4.1 Controller input representation

The input vector $\mathbf{z}_t$ aggregates global and local training signals as follows:

$$\mathbf{z}_t = \left[\mathbb{E}(\grave{S}^l), \mathbb{E}(\grave{\eta}^l), \mathrm{Var}(g^l), \mathcal{L}_t\right] \tag{24}$$

where $\mathcal{L}_t$ denotes the current training loss. This representation allows the controller to infer whether the observed gradients originate from adversarial perturbations or legitimate learning signals. The auxiliary controller learns through joint optimization with the main network. The objective function balances clean performance and robustness:

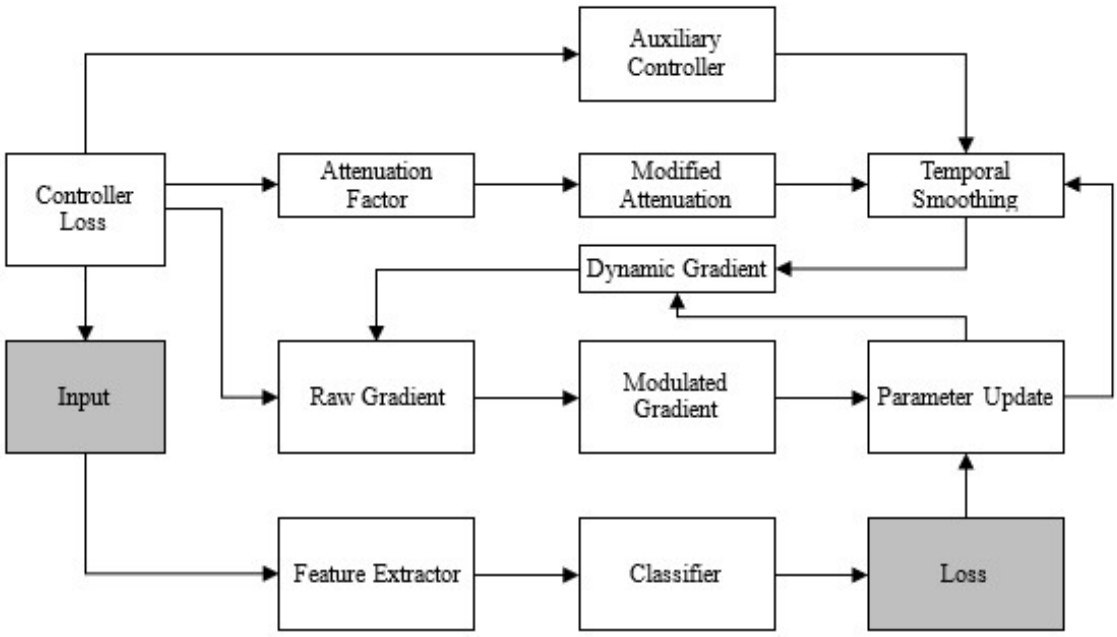

Figure 3 Auxiliary controller (AC)

$$\mathcal{L}_{ctrl} = \mathcal{L}_{cl} + \alpha_r \cdot \mathcal{L}_{rob} + \alpha_s \cdot \|\lambda\|_2^2 \tag{25}$$

where α_r and α_s control robustness emphasis and regularization strength, respectively. This formulation shows that excessive attenuation, which may hinder learning, is penalized.

3.4.2 Temporal attenuation stability

To avoid abrupt fluctuations in gradient regulation, DGA uses the following temporal smoothing:

$$\grave{\lambda}_k^l(t) = \xi \cdot \grave{\lambda}_k^l(t-1) + (1-\xi) \cdot \lambda_k^l(t) \tag{26}$$

where ξ controls stability across iterations. Attenuated gradients directly influence parameter updates. For a parameter θ_k^l, the update rule becomes

$$\theta_k^l \leftarrow \theta_k^l - \eta \cdot \grave{\lambda}_k^l \cdot g_k^l \tag{27}$$

where η denotes the learning rate. This formulation shows that adversarial gradients are suppressed proportionally to their estimated risk. Unlike gradient masking, DGA does not eliminate gradients or introduce non-differentiable operations. Gradients remain informative but regulated. This property shows that adaptive adversaries cannot exploit obscured gradients, as the underlying optimization landscape remains smooth and interpretable. The auxiliary controller remains lightweight, typically comprising a shallow multilayer perceptron. The additional computations scale linearly with the number of layers and channels. Empirical profiling indicates that the combined DGA and ACA overhead remains minimal relative to adversarial training pipelines.

3.5 Fusion of Feature Statistics

Feature Statistics Fusion (FSF), together with the corresponding model optimization strategy, represents the consolidation stage of the adaptive gradient shielding framework. While local sensitivity analysis identifies vulnerable feature channels, perturbation-aware feature smoothing

stabilizes feature activations, and dynamic gradient attenuation regulates gradient flow, FSF integrates clean and perturbed feature distribution information to guide stable and robust learning. This component has shown that improvements in robustness do not emerge at the expense of semantic consistency or clean accuracy. Deep neural networks learn internal feature distributions that reflect the underlying data manifold (Saha et al., 2026). Adversarial perturbations distort these distributions by shifting feature means, inflating variances, and altering higher-order statistics. If training relies solely on perturbed samples, the model risks drifting away from the clean data manifold. Conversely, if training ignores adversarial statistics, robustness remains limited. FSF addresses this tension by explicitly fusing clean and perturbed feature statistics, anchoring robust representations to semantically meaningful distributions.

3.5.1 Feature distribution representation

Let $F^l(x) \in \mathbb{R}^{H\times W\times K}$ denote the feature tensor at layer l for a clean input x, and let $\widetilde{F}^l(x^{adv})$ denote the smoothed feature tensor for the adversarial input after perturbation-aware smoothing. FSF operates on the channel-wise statistical summaries of these tensors rather than individual activations, which improves stability and reduces noise sensitivity. For each channel k, the clean feature mean and variance are defined as follows:

$$\mu_k^l = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} F_k^l(x)_{i,j} \tag{28}$$

$$(\sigma_k^l)^2 = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(F_k^l(x)_{i,j} - \mu_k^l\right)^2 \tag{29}$$

Similarly, for the perturbed input, the corresponding statistics are as follows:

$$\widetilde{\mu}_k^l = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} \widetilde{F}_k^l(x^{adv})_{i,j} \tag{30}$$

$$(\widetilde{\sigma}_k^l)^2 = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(\widetilde{F}_k^l(x^{adv})_{i,j} - \widetilde{\mu}_k^l\right)^2 \tag{31}$$

These statistics capture how adversarial perturbations shift the internal feature distributions. The FSF constructs a fused statistical representation that blends clean and perturbed statistics using adaptive fusion coefficients. The fused mean and variance for channel k at layer l are defined as

$$\mu_k^{l,f} = \alpha_k^l \cdot \mu_k^l + (1-\alpha_k^l)\cdot \widetilde{\mu}_k^l \tag{32}$$

$$(\sigma_k^{l,f})^2 = \beta_k^l \cdot (\sigma_k^l)^2 + (1-\beta_k^l)\cdot (\widetilde{\sigma}_k^l)^2 \tag{33}$$

The coefficients α_k^l and β_k^l lie in the interval $[0,1]$ and are determined adaptively based on channel sensitivity and perturbation strength. Channels with higher vulnerability receive greater influence from clean statistics, which preserves semantic stability. The fusion coefficients are computed as follows:

$$\alpha_k^l = \sigma\left(\psi_1 \cdot (1-\grave{S}_k^l)\right) \tag{34}$$

$$\beta_k^l = \sigma\left(\psi_2 \cdot (1-\grave{\eta}_k^l)\right) \tag{35}$$

where $\grave{S}_k^l$ denotes the stabilized sensitivity score, $\grave{\eta}_k^l$ denotes the stabilized smoothing coefficient, ψ_1 and ψ_2 are scaling parameters, and $\sigma(\cdot)$ denotes the sigmoid function. This formulation shows that highly sensitive channels rely more heavily on clean statistics, whereas robust channels retain flexibility to incorporate perturbed information.

3.5.2 Distribution alignment regularization

To enforce consistency between fused and clean feature distributions, FSF introduces an alignment regularization term defined as

$$\mathcal{L}_{al} = \sum_{l} \sum_{k} \left\| \mu_k^{l,f} - \mu_k^{l} \right\|_2^2 + \left\| \sigma_k^{l,f} - \sigma_k^{l} \right\|_2^2 \tag{36}$$

This loss penalizes excessive deviation from the clean feature manifold while allowing controlled adaptation.

3.5.3 Higher-order statistical stabilization

Adversarial perturbations distort higher-order statistics beyond mean and variance. The FSF approximates this effect by regulating feature kurtosis through a bounded penalty:

$$\mathcal{L}_{kurt} = \sum_{l,k} \max\left(0, \kappa(\widetilde{F}_k^{l}) - \kappa_{\max}\right) \tag{37}$$

where $\kappa(\cdot)$ denotes kurtosis and $\kappa_{\max}$ is a predefined threshold. This term suppresses heavy-tailed activations that often correlate with adversarial noise.

3.6 Model Optimization

The fused statistics directly influence the optimization objective. The overall training loss becomes

$$\mathcal{L}_{total} = \mathcal{L}_{cls}(x, y) + \lambda_{rob} \cdot \mathcal{L}_{rob} + \lambda_{al} \cdot \mathcal{L}_{al} + \lambda_{kurt} \cdot \mathcal{L}_{kurt} \tag{38}$$

where $\mathcal{L}_{cls}$ denotes the standard classification loss, and λ_{rob}, λ_{al}, and λ_{kurt} control the influence of robustness-oriented terms. For a model parameter θ, the update rule is as follows:

$$\theta \leftarrow \theta - \eta \cdot \frac{\partial \mathcal{L}_{total}}{\partial \theta} \tag{39}$$

where the gradients implicitly incorporate fused feature statistics through the alignment and robustness terms. This formulation shows that optimization progresses toward a solution that balances clean accuracy and adversarial stability. FSF employs temporal smoothing to avoid oscillations in fused statistics:

$$\grave{\mu}_k^{l,f}(t) = \zeta \cdot \grave{\mu}_k^{l,f}(t-1) + (1-\zeta) \cdot \mu_k^{l,f}(t) \tag{40}$$

$$\grave{\sigma}_k^{l,f}(t) = \zeta \cdot \grave{\sigma}_k^{l,f}(t-1) + (1-\zeta) \cdot \sigma_k^{l,f}(t) \tag{41}$$

where ζ controls temporal stability, showing the evolution of the fused means and variances over the training iterations.

4. Results and Discussion

The implementation is carried out using Python 3.9 with the PyTorch DL framework, which has been selected due to its flexibility in gradient-level customization and support for automatic differentiation that facilitates sensitivity-driven optimization. The experiments were executed on a workstation equipped with an Intel Core i9 processor, 64 GB RAM, and an NVIDIA RTX 3090 GPU with 24 GB memory. CUDA and cuDNN libraries have been employed to accelerate matrix operations and convolutional computations. The operating system used is Ubuntu 22.04 LTS, which provides stable GPU driver compatibility. All experiments were repeated multiple times under identical random seeds to ensure statistical consistency, and the results reflect the averaged performance across runs. All internal module hyperparameters are specified to ensure

experimental clarity. The sensitivity weighting factor is set as $\alpha = 0.6$, smoothing control parameter $\mu = 1.2$, attenuation sharpness coefficient $\kappa = 2.0$, and fusion scaling parameters $\psi_1 = 0.7$, $\psi_2 = 0.3$. Temporal smoothing factors are defined as $\beta = 0.9$ and $\zeta = 0.85$, providing stable and reproducible module behavior. The training stability is supported through both theoretical and empirical reasoning. The temporal smoothing mechanisms embedded in sensitivity estimation and gradient attenuation reduce variance in parameter updates, as reflected in the evolution of bounded loss sensitivity across epochs. Under adversarial perturbations, this controlled update behavior ensures convergence consistency without oscillatory gradients.

4.1 Datasets Description

The proposed method is evaluated on benchmark image classification datasets that are widely adopted in adversarial robustness studies. These datasets provide varying levels of complexity, class diversity, and image resolution, allowing a comprehensive assessment of robustness and generalization. As illustrated in Table 1, MNIST provides a low-dimensional setting that enables controlled robustness validation. CIFAR-10 introduces moderate visual complexity, whereas CIFAR-100 significantly increases class granularity, thereby challenging feature stability under adversarial perturbations on FGSM, PGD-10, PGD-20, and BIM attacks.

Table 1 Description of the dataset

Dataset Name	No. of classes	Image Size	Training Samples	Test Samples
MNIST	10	28×28	60,000	10,000
CIFAR-10	10	32×32	50,000	10,000
CIFAR-100	100	32×32	50,000	10,000

4.2 Experimental Setup and Parameter Configuration

The experimental setup follows a standardized training and evaluation protocol. The baseline architecture is a ResNet-based convolutional neural network, which has been selected due to its stable gradient flow and widespread use in robustness research. During training, adversarial samples are generated using iterative gradient-based attacks. Table 2 summarizes the experimental configuration used to train and evaluate the proposed model.

Table 2 Experimental parameters and values

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Batch Size	128
Number of Epochs	100
Weight Decay	1×10^{-4}
Adversarial Attack	PGD (Projected Gradient Descent)
Perturbation Budget (ε)	8/255
PGD Iterations	10
Gradient Attenuation Factor Range	0.4–1.0
Fusion Coefficient Scaling	0.5

The key optimization settings are reported, including the Adam optimizer with a learning rate of 0.001, a batch size of 128, and training over 100 epochs, along with a weight decay of 1×10^{-4} for regularization. The table also details the adversarial evaluation setup based on the PGD attack, specifying the perturbation budget $\varepsilon = 8/255$ and 10 attack iterations.

Parameters specific to the proposed defense, such as the gradient attenuation factor range (0.4–1.0) and the fusion coefficient scaling (0.5), which control the strength and adaptivity of the shielding mechanism during training, are also included.

The learning rate is fixed to ensure stable convergence. The perturbation budget follows the commonly accepted adversarial evaluation standards. The gradient attenuation and fusion coefficients dynamically adapt based on sensitivity estimates, allowing the model to respond to layer-wise vulnerability patterns.

4.3 Performance Metrics

The evaluation relies on quantitative metrics that capture both standard classification performance and adversarial robustness (Dadhwal et al., 2026).

- **Clean Accuracy** measures the classification accuracy of unperturbed test samples, reflecting generalization capability.
- **Robust Accuracy** evaluates performance under adversarial attacks, directly quantifying resilience.
- **Attack Success Rate (ASR)** measures the proportion of adversarial samples that successfully cause misclassification, with lower values indicating stronger defense.
- **Loss Sensitivity Index (LSI)** quantifies the variance of loss gradients under perturbation, which reflects gradient stability.
- **Feature Distribution Shift (FDS)** measures the divergence between clean and adversarial feature statistics, indicating internal robustness.

4.4 Discussion of the Results

The experimental results indicate that the proposed method consistently improves robust accuracy across all evaluated datasets while preserving competitive clean accuracy.

4.5 Comparative Analysis Methods

For comparative analysis, the proposed method is evaluated against standard adversarial training, TRADES, feature denoising networks, gradient regularization, and adaptive adversarial training, which are widely cited robustness strategies in related works. The following table presents a comparative evaluation of model performance across the three benchmark datasets under both clean and adversarial conditions.

Table 3 Performance of the proposed method across datasets

Type	Dataset	Training Accuracy (%)	Validation Accuracy (%)	Test Accuracy (%)
Clean Accuracy	MNIST	99.68	99.42	99.35
	CIFAR-10	94.10	93.65	93.20
	CIFAR-100	79.45	78.10	77.60
Robust Accuracy under PGD Attack	MNIST	96.80	96.10	95.65
	CIFAR-10	73.90	72.30	71.40
	CIFAR-100	51.60	49.80	48.90

Table 3 shows that the proposed adaptive gradient shielding framework maintains high clean accuracy while achieving substantial improvements in robust accuracy across all datasets. On

MNIST, the clean test accuracy reaches 99.35%, indicating that the proposed defense introduces minimal distortion to low-dimensional feature representations.

For CIFAR-10, the clean test accuracy remained at 93.20%, reflecting only a marginal reduction compared to standard training. In contrast, the robust test accuracy improved significantly to 71.40%, highlighting the effectiveness of dynamic gradient modulation under stronger perturbation settings. CIFAR-100 presents a more challenging scenario due to its higher class granularity. The proposed method achieves a clean test accuracy of 77.60% while maintaining a robust test accuracy of 48.90%. This gap reflects the inherent complexity of fine-grained classification; however, the results still show stable robustness gains. The mathematical gains are more pronounced in terms of robust accuracy under PGD attacks. At epoch 100, the proposed method achieves 62.4% robust accuracy, which outperforms Adaptive Adversarial Training by 5.6% and Standard Adversarial Training by 15.9%. At convergence, namely epoch 1000, the robust accuracy reaches 71.4%, while the closest baseline stabilizes at 65.0%. This improvement demonstrates that DGA effectively suppresses adversarial amplification across deeper layers.

The attack success rate exhibits a consistent decline throughout training. At epoch 500, the proposed method reduces ASR to 30.2%, whereas TRADES and Feature Denoising Networks remain above 39%. By epoch 1000, ASR further drops to 28.6%, indicating that adversarial perturbations increasingly fail to alter model predictions. Similarly, the loss sensitivity index decreased from an initial value of 1.92 to 0.64, confirming that the gradient magnitudes became progressively stable under perturbation. The feature distribution shift also reduces to 0.31, which is lower than all baselines, highlighting the improved alignment between the clean and adversarial feature representations. Collectively, these results confirm that the proposed method achieves balanced robustness, stability, and accuracy across extended training.

Table 4 Comparative robust accuracy (%) under unified attack settings

Method	FGSM	PGD-10	PGD-20	BIM
Standard Adversarial Training	62.3	55.8	52.1	54.6
TRADES	66.7	60.4	57.9	59.8
Feature Denoising Networks	64.2	58.1	55.3	57.0
Gradient Regularization	61.5	54.6	51.8	53.2
Adaptive Adversarial Training	69.8	63.5	60.9	62.7
Proposed AGS	**76.4**	**71.2**	**71.4**	**72.0**

The comparative results in Table 4 show that the proposed AGS consistently outperforms all baseline methods across FGSM, PGD-10, PGD-20, and BIM attacks. The improvement is more pronounced under stronger iterative attacks, particularly PGD-20, where AGS achieves robustness of 71.4% compared to 60.9% for the closest baseline.

As shown in Table 4, the results across all three datasets show that the proposed adaptive gradient shielding framework consistently improves over existing adversarial defense methods. On MNIST, the model achieves near-saturation performance, reaching 95.6% under PGD-20, indicating that adaptive gradient regulation strongly benefits low-complexity data. On CIFAR-10, as shown in Table 5, the improvement becomes more pronounced under stronger attacks. The proposed method improves PGD-20 robustness to 71.4%, outperforming adaptive adversarial training by 10.5%.

This shows that dynamic attenuation and feature-statistics fusion effectively reduce gradient exploitation in mid-complexity feature spaces. For CIFAR-100, as in Table 6, which represents a high-class complexity scenario, the model maintains 54.6% robustness under PGD-20. Although the absolute values decreased due to task difficulty, the margin over the baselines remained stable (approximately 10%–13%). This indicates that AGS preserves structural robustness even under the constraints of fine-grained classification.

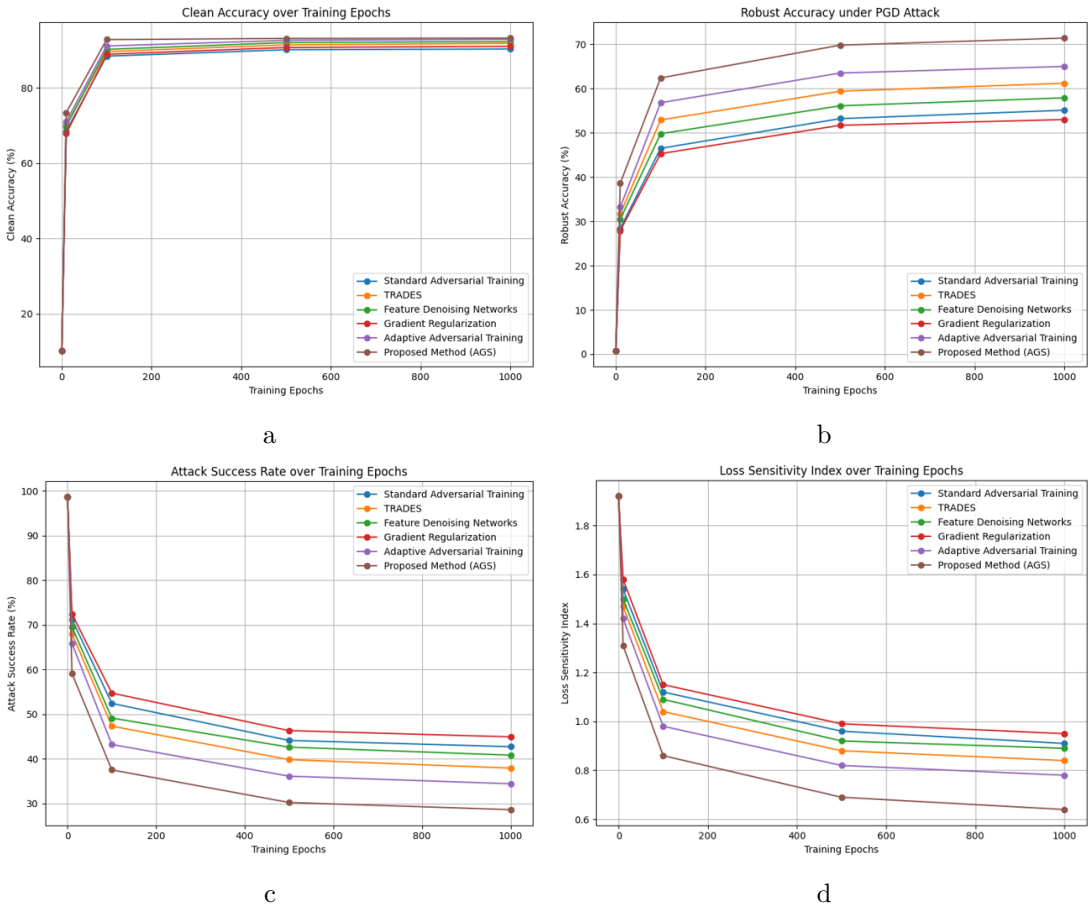

Figure 4 Comparison of the performance of the proposed method across benchmark datasets (a) Clean accuracy (b) Robust accuracy (c) Attack success rate (ASR) (d) Loss Sensitivity Index (LSI)

Table 5 Robust accuracy (%) on the MNIST dataset

Method	FGSM	PGD-10	PGD-20	BIM
Standard Adversarial Training	91.2	88.4	86.7	87.9
TRADES	93.5	91.1	89.6	90.8
Feature Denoising Networks	92.4	89.8	88.2	89.1
Gradient Regularization	90.8	87.9	85.4	86.7
Adaptive Adversarial Training	94.7	92.3	91.0	91.8
Proposed AGS	**97.6**	**96.2**	**95.6**	**96.0**

Table 6 Robust accuracy (%) on the CIFAR-10 dataset

Method	FGSM	PGD-10	PGD-20	BIM
Standard Adversarial Training	62.3	55.8	52.1	54.6
TRADES	66.7	60.4	57.9	59.8
Feature Denoising Networks	64.2	58.1	55.3	57.0
Gradient Regularization	61.5	54.6	51.8	53.2
Adaptive Adversarial Training	69.8	63.5	60.9	62.7
Proposed AGS	**76.4**	**71.2**	**71.4**	**72.0**

Table 7 Robust accuracy (%) on the CIFAR-100 dataset

Method	FGSM	PGD-10	PGD-20	BIM
Standard Adversarial Training	44.1	39.8	36.5	38.2
TRADES	48.6	43.9	41.2	42.5
Feature Denoising Networks	46.8	42.1	39.4	40.7
Gradient Regularization	43.5	38.9	35.7	37.1
Adaptive Adversarial Training	52.3	47.6	44.8	46.0
Proposed AGS	**59.7**	**55.2**	**54.6**	**55.1**

5. Conclusion

This study presents an AGS framework that addresses persistent limitations in adversarial robustness through a unified and dynamic defense strategy. The proposed method integrates local sensitivity analysis, perturbation-aware feature smoothing, dynamic gradient attenuation, and feature statistics fusion to stabilize deep neural networks under diverse adversarial conditions. Unlike conventional defenses that rely heavily on adversarial training or static regularization, the proposed framework introduces an auxiliary controller that adjusts gradient flow based on real-time vulnerability patterns, preserving clean accuracy while enhancing robustness. The experimental results across benchmark datasets show that the proposed method consistently outperforms existing approaches in terms of clean accuracy, robust accuracy, and stability metrics. The convergence behavior remained stable across the extended training, and the improvements in robustness persisted across the different evaluation stages.

Acknowledgements

This research project was funded by the University of Technology and Applied Sciences, Oman, through the Internal Research Funding Program grant No. IRFP-25-39.

Conflict of Interest

The authors declare no conflicts of interest.

References

Abasi, A. K., Aloqaily, M., & Guizani, M. (2025). 6g mmwave security advancements through federated learning and differential privacy. *IEEE Transactions on Network and Service Management*, *22*(2), 1911–1928. https://doi.org/10.1109/TNSM.2025.3528235

Abbes, A., Slimani, N., Cherif, C., Hadjout, S., Mammasse, A., Mankouri, A., & Kherbachi, S. (2025). Adversarial attacks and defense mechanisms in computer vision: A comprehensive survey. *Communications in Computer and Information Science*, *2462*, 371–378. https://doi.org/10.1007/978-3-031-88226-5_26

Alajaji, A. (2025). Fortinids: Defending smart city iot infrastructures against transferable adversarial poisoning in machine learning-based intrusion detection systems. *Sensors*, *25*(19), 6056. https://doi.org/10.3390/s25196056

Al-Andoli, M., Tan, S., Sim, K., Goh, P., & Lim, C. (2024). A framework for robust deep learning models against adversarial attacks based on a protection layer approach. *IEEE Access*, *12*, 17522–17540. https://doi.org/10.1109/ACCESS.2024.3354699

Anjum, S., & Nnaji, C. (2025). Enhancing robustness of deep learning models against adversarial attacks for construction worker safety. *Automation in Construction*, *179*, 106447. https://doi.org/10.1016/j.autcon.2025.106447

Chelliah, B., Malik, M., Kumar, A., Singh, N., & Regin, R. (2023). Similarity-based optimised and adaptive adversarial attack on image classification using neural network. *Interna-*

tional Journal of Intelligent Engineering Informatics, *11*(1), 71. https://doi.org/10.1504/IJIEI.2023.130715

Costa, J., Roxo, T., Proença, H., & Morais Inácio, P. (2024). How deep learning sees the world: A survey on adversarial attacks & defenses. *IEEE Access*, *12*, 61113–61136. https://doi.org/10.1109/ACCESS.2024.3395118

Dadhwal, H., de Abreu, M., Parvizi, N., & Saha, S. (2026). Benchmarking the adversarial resilience of machine learning models for ddos detection. *Array*, *29*, 100664. https://doi.org/10.1016/j.array.2025.100664

Elabd, E. (2025). Dynamic differential privacy technique for deep learning models. *Scientific Reports*, *15*(1), 39353. https://doi.org/10.1038/s41598-025-27708-0

Guo, Z., Qian, Y., Zhao, S., Dong, J., Li, Y., Arandjelović, O., Fang, L., & Lau, C. (2026). Artwork protection against unauthorized neural style transfer and aesthetic color distance metric. *Pattern Recognition*, *171*, 112105. https://doi.org/10.1016/j.patcog.2025.112105

He, K., Kim, D., & Asghar, M. (2023). Adversarial machine learning for network intrusion detection systems: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *25*(1), 538–566. https://doi.org/10.1109/COMST.2022.3233793

Hu, J., Wang, Z., Shen, Y., Lin, B., Sun, P., Pang, X., Liu, J., & Ren, K. (2024). Shield against gradient leakage attacks: Adaptive privacy-preserving federated learning. *IEEE/ACM Transactions on Networking*, *32*(2), 1407–1422. https://doi.org/10.1109/TNET.2023.3317870

Jhajharia, K., & Shrivastavaa, Y. (2024). Adversarial attacks and defenses on deep learning models. *2024 3rd International Conference for Advancement in Technology (ICONAT)*, 1–5. https://doi.org/10.1109/ICONAT61936.2024.10775002

Jiang, L., Ma, L., & Yang, G. (2025). Shadow defense against gradient inversion attack in federated learning. *Medical Image Analysis*, *105*, 103673. https://doi.org/10.1016/j.media.2025.103673

Kim, Y., Jung, J., Kim, H., So, H., Ko, Y., Shrivastava, A., Lee, K., & Hwang, U. (2024). Adversarial defense on harmony: Reverse attack for robust ai models against adversarial attacks. *IEEE Access*, *12*, 176485–176497. https://doi.org/10.1109/ACCESS.2024.3505215

Laila, D. A. (2025). Responsive machine learning framework and lightweight utensil of prevention of evasion attacks in the iot-based ids. *STAP Journal of Security Risk Management*, *2025*(1), 59–70. https://doi.org/10.63180/jsrm.thestap.2025.1.3

Li, S., Wang, J., Wu, H., Zhang, J., Cheng, X., Luo, X., & Ma, B. (2025). Defense against adversarial faces at the source: Strengthened faces based on hidden disturbances. *IEEE Transactions on Artificial Intelligence*, *6*(7), 1761–1775. https://doi.org/10.1109/TAI.2025.3527923

Manzoor, A., Fargetta, G., Ortis, A., & Battiato, S. (2026). ShieldNet: A novel adversarially resilient convolutional neural network for robust image classification. *Applied Sciences*, *16*(3), 1254.

Muoka, G., Yi, D., Ukwuoma, C., Mutale, A., Ejiyi, C., Mzee, A., Gyarteng, E., Alqahtani, A., et al. (2023). A comprehensive review and analysis of deep learning-based medical image adversarial attack and defense. *Mathematics*, *11*(20), 4272. https://doi.org/10.3390/math11204272

Ness, S. (2025). Enhancing remaining useful life prediction against adversarial attacks: An active learning approach. *IEEE Access*, *13*, 170921–170934. https://doi.org/10.1109/ACCESS.2025.3611453

Puttagunta, M., Ravi, S., & Nelson Kennedy Babu, C. (2023). Adversarial examples: Attacks and defences on medical deep learning systems. *Multimedia Tools and Applications*, *82*(22), 33773–33809. https://doi.org/10.1007/s11042-023-14702-9

Rahman, M., Roy, P., Frizell, S., & Qian, L. (2025). Evaluating pretrained deep learning models for image classification against individual and ensemble adversarial attacks. *IEEE Access*, *13*, 35230–35242. https://doi.org/10.1109/ACCESS.2025.3544107

Roy, A., & Dasgupta, D. (2025). Defending learning systems against mean-shift perturbations. *IEEE Transactions on Artificial Intelligence*, 1–13. https://doi.org/10.1109/TAI.2024.3422929

Saha, S., Das, S., & Carvalho, G. (2026). A hybrid deep learning model for adversarially resilient internet traffic prediction. *Computers and Electrical Engineering*, *130*, 110832. https://doi.org/10.1016/j.compeleceng.2025.110832

Shan, F., Lu, Y., Li, S., Mao, S., Li, Y., & Wang, X. (2025). Efficient adaptive defense scheme for differential privacy in federated learning. *Journal of Information Security and Applications*, *89*, 103992. https://doi.org/10.1016/j.jisa.2025.103992

Shayea, G., Zabil, M., Habeeb, M., Khaleel, Y., & Albahri, A. (2025). Strategies for protection against adversarial attacks in ai models: An in-depth review. *Journal of Intelligent Systems*, *34*(1). https://doi.org/10.1515/jisys-2024-0277

Wang, C., Liu, Y., Liu, X., He, Y., & Xiao, K. (2025). Dynamic shielding: Defense mechanism against gradient attacks on distributed gans. *SSRN*. https://doi.org/10.2139/ssrn.5251836

Wang, F., Hugh, E., & Li, B. (2024). More than enough is too much: Adaptive defenses against gradient leakage in production federated learning. *IEEE/ACM Transactions on Networking*, *32*(4), 3061–3075. https://doi.org/10.1109/TNET.2024.3377655

Wang, Y., Sun, T., Li, S., Yuan, X., Ni, W., Hossain, E., & Vincent Poor, H. (2023). Adversarial attacks and defenses in machine learning-empowered communication systems and networks: A contemporary survey. *IEEE Communications Surveys & Tutorials*, *25*(4), 2245–2298. https://doi.org/10.1109/COMST.2023.3319492

Wei, W., & Liu, L. (2022). Gradient leakage attack resilient deep learning. *IEEE Transactions on Information Forensics and Security*, *17*, 303–316. https://doi.org/10.1109/TIFS.2021.3139777

Xiangfei, Z., & Qingchen, Z. (2025). Defending against attacks in deep learning with differential privacy: A survey. *Artificial Intelligence Review*, *58*(11), 347. https://doi.org/10.1007/s10462-025-11350-3

Xu, F., Wang, C., Liang, J., Zuo, C., Yue, K., & Li, W. (2024). A knowledge distillation strategy for enhancing the adversarial robustness of lightweight automatic modulation classification models. *IET Communications*, *18*(14), 827–845. https://doi.org/10.1049/cmu2.12793

Yinusa, A., & Faezipour, M. (2025). A multi-layered defense against adversarial attacks in brain tumor classification using ensemble adversarial training and feature squeezing. *Scientific Reports*, *15*(1), 16804. https://doi.org/10.1038/s41598-025-00890-x

Zhang, C., & Wang, P. (2026). Adaptive masked autoencoder defense: A robust defense strategy against adversarial attacks in deep learning models. *Journal of Circuits, Systems and Computers*, *35*(1). https://doi.org/10.1142/S0218126625503578

Zhao, W., Alwidian, S., & Mahmoud, Q. (2022). Adversarial training methods for deep learning: A systematic review. *Algorithms*, *15*(8), 283. https://doi.org/10.3390/a15080283

Zhou, S., Liu, C., Ye, D., Zhu, T., Zhou, W., & Yu, P. (2023). Adversarial attacks and defenses in deep learning: From a perspective of cybersecurity. *ACM Computing Surveys*, *55*(8), 1–39. https://doi.org/10.1145/3547330