MPD-SGR: Robust Spiking Neural Networks with Membrane Potential Distribution-Driven Surrogate Gradient Regularization

Runhao Jiang[1, 2], Chengzhi Jiang[1, 2], Rui Yan[3], Huajin Tang[1, 2, 4*]

[1]College of Computer Science and Technology, Zhejiang University
[2]The State Key Lab of Brain-Machine Intelligence, Zhejiang University
[3]College of Computer Science and Technology, Zhejiang University of Technology
[4]MOE Frontier Science Center for Brain Science and Brain-Machine Integration, Zhejiang University
RhJiang@zju.edu.cn, jiangchengzhi2024@zju.edu.cn, Ryan@zjut.edu.cn, htang@zju.edu.cn

Abstract

The surrogate gradient (SG) method has shown significant promise in enhancing the performance of deep spiking neural networks (SNNs), but it also introduces vulnerabilities to adversarial attacks. Although spike coding strategies and neural dynamics parameters have been extensively studied for their impact on robustness, the critical role of gradient magnitude, which reflects the model's sensitivity to input perturbations, remains underexplored. In SNNs, the gradient magnitude is primarily determined by the interaction between the membrane potential distribution (MPD) and the SG function. In this study, we investigate the relationship between the MPD and SG and their implications for improving the robustness of SNNs. Our theoretical analysis reveals that reducing the proportion of membrane potentials lying within the gradient-available range of the SG function effectively mitigates the sensitivity of SNNs to input perturbations. Building upon this insight, we propose a novel MPD-driven surrogate gradient regularization (MPD-SGR) method, which enhances robustness by explicitly regularizing the MPD based on its interaction with the SG function. Extensive experiments across multiple image classification benchmarks and diverse network architectures confirm that the MPD-SGR method significantly enhances the resilience of SNNs to adversarial perturbations and exhibits strong generalizability across diverse network configurations, SG functions, and spike encoding schemes.

Extended version — https://arxiv.org/abs/2511.12199

Introduction

Robustness is an intrinsic characteristic of the human brain, a trait that traditional artificial neural networks (ANNs) fail to replicate. Unlike ANNs that rely on dense embeddings (Wang et al. 2022, 2024a,b), Spiking neural networks (SNNs) (Maass 1997) emulate the structural and functional properties of the brain by biological neurons that encode information via binary spikes, thereby demonstrating enhanced resilience to noise and perturbations (Marchisio et al. 2020; Sharmin et al. 2019; Hu et al. 2024). Although SNNs currently underperform ANNs in terms of accuracy, their intrinsic robustness advantage makes them particularly suitable for high-security applications like autonomous driving (Zhu et al. 2024) and privacy computing (Kim, Venkatesha, and Panda 2022).

The development of surrogate gradient (SG) methods has significantly advanced the performance of deep SNNs (Tavanaei et al. 2019), yet this progress introduces vulnerabilities to gradient-based adversarial attacks (Liang et al. 2021). Recent studies have shown that, although SNNs inherently exhibit greater robustness than ANNs, such adversarial perturbations can critically compromise their accuracy (Ding et al. 2022). Enhancing the adversarial robustness of SNNs remains a pivotal challenge in the research community. Some techniques originally developed for ANNs, such as adversarial training (Ding et al. 2024a) and Lipschitz regularization (Ding et al. 2022), have been extended to SNNs. In parallel, growing attention has been paid to understanding how structural parameters (e.g., leak factors (Chowdhury, Lee, and Roy 2021; Sharmin et al. 2020; Xu et al. 2024), thresholds (El-Allami et al. 2021)) and neural encoding schemes (Ma, Yan, and Tang 2023; Ding et al. 2024b; Wu et al. 2024) affect the robustness of SNNs. The intrinsic noise-filtering properties of spiking dynamics, along with the stochastic nature of neural coding, are considered critical factors underpinning the robustness of SNNs.

However, existing research (Xu et al. 2024; Liu et al. 2024) has demonstrated that a model's sensitivity to input perturbations (i.e., robustness error) can be reflected in the magnitude of gradients, which is primarily governed by the interaction between the membrane potential distribution (MPD) and the SG function in SNNs. The MPD not only reflects the response characteristics of the neuronal population but also significantly influences the gradient propagation across network layers. Recent studies (Guo et al. 2022a,b, 2023) have further shown that constraining the MPD to remain within the gradient-available region of the SG function with an appropriate proportion facilitates more effective gradient-based optimization. Therefore, it is also crucial to investigate how to constrain the MPD according to the SG function to enhance the robustness of SNNs.

In this paper, we present a theoretical and empirical investigation into the role of neuronal MPD and SG mechanisms in enhancing the adversarial robustness of SNNs. First, through rigorous gradient analysis, we establish a theoretical framework that formally connects robustness error with the SG in SNNs, demonstrating that reducing the SG

*Author for correspondence

magnitude can effectively mitigate robustness error. Since the SG magnitude is primarily influenced by the membrane potential, we further investigate the MPD and establish an explicit connection between the MPD and network parameters. Thus, the MPD can be optimized by learning the network parameters. Lastly, based on the relationship between the MPD and the SG function, we derive an approximate expression for the SG magnitude. Based on these theoretical insights, we design an MPD-driven surrogate gradient regularization (MPD-SGR) method that reduces the proportion of MPD within the gradient-available region of the SG function, thereby enhancing the network's resilience against adversarial attacks (Figure 1). The proposed framework demonstrates remarkable adaptability across diverse neuron parameter configurations, SG function variants, and spike coding methods. Experimental results across multiple image classification datasets, network architectures, and gradient approximation methods demonstrate that our MPD-SGR method effectively improves the robustness of SNNs against various input perturbations.

Related Work

Surrogate Gradient Learning

Surrogate gradient (SG) methods provide approximate gradients for the non-differentiable spiking neurons, thus supporting the backpropagation of errors in both spatial and temporal domains (Wu et al. 2018; Gu et al. 2019; Neftci, Mostafa, and Zenke 2019). Recent studies have found that the relationship between the gradient-available interval of the SG function and the membrane potential dynamics has a significant impact on the backpropagation training of SNNs (Zenke and Vogels 2021), inspiring a series of works to study the alignment between membrane potential and gradient-available interval. Some researchers adopt a fixed SG function and constrain the membrane potential distribution to ensure that the gradient is propagated in an appropriate proportion. InfLoR-SNN (Guo et al. 2022a) designed a membrane potential rectifier to redistribute the membrane potential closer to the spiking threshold. RecDis-SNN (Guo et al. 2022b) introduced three regularization losses to penalize three undesired shifts of the membrane potential distribution. Optimizing the SG function is another appealing approach. Dspike (Li et al. 2021) adaptively changed its shape and captured the direction of finite difference gradients to find the optimal shape and smoothness for gradient estimation. LSG (Lian et al. 2023) adapted the width of the SG function automatically during training based on the membrane potential dynamics. ASGL (Wang et al. 2023) learned the precise gradients of the loss landscape in SNNs adaptively by fusing the learnable relaxation degree into a prototype network with random spike noise.

Adversarial Defense for SNN

SNNs exhibit inherent robustness (Marchisio et al. 2020; Sharmin et al. 2019), a property that is often attributed to their biologically plausible components, such as neural coding and dynamics. Several studies aim to enhance robustness from a neurodynamic perspective. For example, structural parameters, including time windows, leak factors, and thresholds, play a key role in determining the adversarial robustness of SNNs (El-Allami et al. 2021). The noise-filtering effect of membrane potential leakage characteristics has motivated approaches to improve the SNN robustness by evolving leaky factor (Chowdhury, Lee, and Roy 2021; Sharmin et al. 2020). Additionally, leveraging stochasticity or attention mechanisms in neural coding can mitigate noise in the information transmission process. For instance, Poisson coding, which encodes the continuous intensity of an image into binary spikes with inherent randomness, has been shown to exhibit greater robustness than direct coding (Sharmin et al. 2020; Kim et al. 2022). To further utilize the noisy, non-deterministic nature of neural coding, NDL (Ma, Yan, and Tang 2023) incorporates noisy neuronal dynamics into SNNs to investigate their potential robustness benefits, while StoG (Ding et al. 2024b) introduces additional stochastic gating for spiking neurons to enhance model robustness. The FEEL method (Xu et al. 2024) adopts a frequency encoding inspired by selective visual attention mechanisms, suppressing noise in different frequency ranges at different time steps. Moreover, there are also some SNN-specific defensive techniques that draw inspiration from ANNs. A typical representative is adversarial training (AT) (Kundu and Pedram 2021), which enhances robustness by incorporating adversarial examples generated through attacks into the training process. RAT (Ding et al. 2022) proposes a regularized training strategy based on Lipschitz analysis of SNNs.

Preliminaries

Spiking Neural Networks

In the field of deep SNNs, the most commonly used spiking neuron is the iterative leaky integrate-and-fire (LIF) model. The iterative LIF model simulates biological neurons through three key dynamic processes: synaptic integration, membrane potential accumulation, and neuronal firing with resetting, which are mathematically defined as follows:

$$I^{l-1}(t) = W^l \cdot O^{l-1}(t), \tag{1}$$

$$R^l(t-1) = O^l(t-1)r^l(t-1), \tag{2}$$

$$U^l(t) = \tau(U^l(t-1) - R^l(t-1)) + I^{l-1}(t), \tag{3}$$

$$O^l(t) = \mathcal{H}(\overline{U}^l(t) - v_{th}), \tag{4}$$

where l and t denote the layer index and the time step of neural activity, respectively. The postsynaptic current $I^{l-1}(t)$ is the sum of spike signals $O^{l-1}(t)$ weighted by the synaptic weights W^l. $R^l(t-1)$ is the reset term, where the triggering spike $O^l(t-1)$ causes the membrane potential reset and $r^l(t)$ is the subtracted value due to reset. Under the hard reset condition, $r^l(t) = U^l(t)$, whereas under the soft reset condition, $r^l(t) = v_{th}$. $U^l(t)$ denotes the membrane potential state, receiving the postsynaptic current and decaying with a leaky factor τ. When $U^l(t)$ exceeds the threshold v_{th}, a spike is generated according to the Heaviside function $\mathcal{H}(\cdot)$.

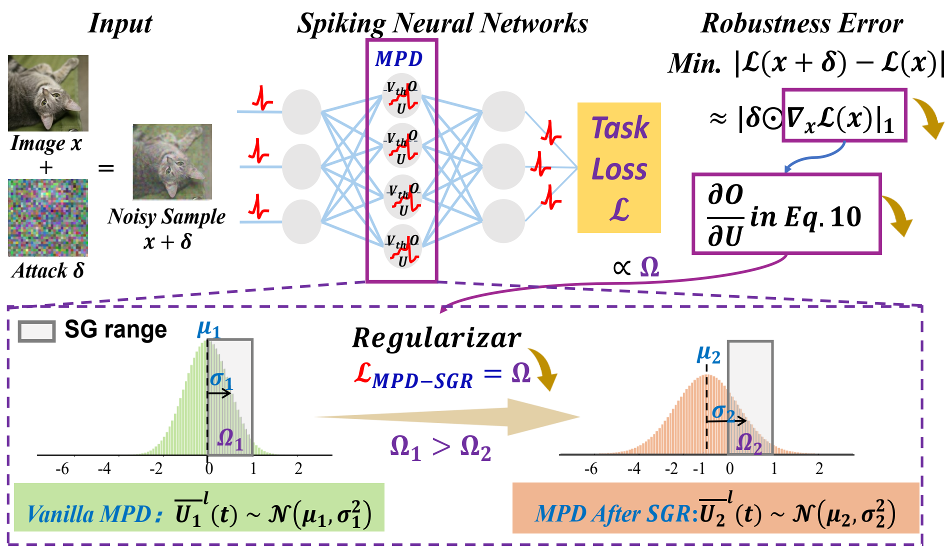

Figure 1: The overall framework of MPD-SGR. The MPD-SGR constrains membrane potential distribution (MPD) of SNNs (mean μ and standard deviation σ) to reduce the overlap Ω between MPD and the gradient-available range of the SG function (gray area). This regularization minimizes output error under adversarial perturbations.

Adversarial Attacks for SNN

Neural networks are notorious for their vulnerability to subtle perturbations in the input data, known as adversarial attacks. Adversarial attacks generate a perturbation δ by maximizing the network's task loss $\mathcal{L}$ and then applying it to input data x, resulting in adversarial examples. Formally, this maximization problem can be expressed as:

$$\delta = \arg\max \mathcal{L}(f(x+\delta; W), y) \quad s.t. \delta \in R(x, \epsilon) \tag{5}$$

where y is the target label, f is the network parameterized by its weights W, and $R(x, \epsilon)$ is the ℓ_p-constrained neighborhood centered on x with radius ϵ, ensuring the perturbation remains imperceptible.

Researchers have observed that adversarial attacks can also be applied to deep SNNs using SG (Bu et al. 2023), where the SG function provides approximate gradients for non-differentiable Heaviside step function. In the SG method, the backpropagation of the loss is derived by unfolding the dynamics of LIF neurons as follows:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{U}^l(t)} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{O}^l(t)} \frac{\partial \boldsymbol{O}^l(t)}{\partial \boldsymbol{U}^l(t)} + \frac{\partial \mathcal{L}}{\partial \boldsymbol{U}^l(t+1)} \frac{\partial \boldsymbol{U}^l(t+1)}{\partial \boldsymbol{U}^l(t)} \tag{6}$$

where $\mathcal{L}$ is the task loss. SG methods approximate the non-differentiable term $\frac{\partial \boldsymbol{O}^l(t)}{\partial \boldsymbol{U}^l(t)}$ with an SG function. Among the various SG functions demonstrated to be effective (Neftci, Mostafa, and Zenke 2019), we employ the triangle SG function defined as:

$$\frac{\partial \boldsymbol{O}^l(t)}{\partial \boldsymbol{U}^l(t)} \approx h\left(\overline{\boldsymbol{U}}^l(t)\right) = \frac{1}{\gamma^2} \max\left(\gamma - \left|\overline{\boldsymbol{U}}^l(t)\right|, 0\right) \tag{7}$$

where $\overline{\boldsymbol{U}}^l(t) = \boldsymbol{U}^l(t) - v_{th}$ and γ is a hyperparameter that controls the smoothness and gradient-available interval of the surrogate function.

Methods

Analysis of robustness error

The effect of a perturbation is quantified as the difference in error values before and after perturbation (i.e., robustness error), $\mathcal{L}(x+\delta) - \mathcal{L}(x)$. This perturbation-induced error difference is theoretically analyzed using local linearity techniques (Qin et al. 2019), yielding the following expression:

$$|\mathcal{L}(x+\delta) - \mathcal{L}(x)| \leqslant |\delta \odot \nabla_x \mathcal{L}(x)|_1 + g(\delta, x) \tag{8}$$

where $g(\delta, x)$ is the residual term and $\nabla_x \mathcal{L}(x)$ is the input gradient. This theorem reveals a relationship between variations in robustness error and the L_1 norm of the input gradient. A natural approach to reduce these variations is to regularize the input gradient. However, this requires double backpropagation to compute its gradient with respect to model parameters, which is computationally expensive and impractical for large-scale SNNs.

To address this, we leverage LIF dynamics and BPTT to reformulate input-gradient optimization as internal network-gradient optimization (Xu et al. 2024). Specifically, the constraint term for SNNs can be rewritten as $\sum_t |\delta_t \odot \nabla_O \mathcal{L}(O_t)|_1$, which can be further derived as follows:

$$\min \sum_t \left|\delta_t \odot \frac{\partial \mathcal{L}}{\partial O_t}\right|_1 = \min \sum_t \left|\frac{1}{L} \sum_{l=1}^{L} (P_1 \cdot P_2 \cdot P_3) \frac{\partial \mathcal{L}}{O_l^T}\right|_1 \tag{9}$$

Equation 9 outlines the key components influencing robustness, including the perturbation term $P_1 = \prod_{k=t}^{T} \delta(t) \odot \tau_l^k$, the model weight term $P_2 = \prod_{q=2}^{l} W_{q-1,q}$ and the SG term $P_3 = \prod_{v=1}^{l} \frac{\partial O_v^t}{\partial U_v^t}$. These constraints offer a principled way to regularize SNNs for robustness, explaining why recent methods such as weight regularization (Ding et al.

2022) and evolving leak factors (Xu et al. 2024) are effective. However, the potential of SG terms for improving robustness has been largely overlooked, which is the motivation of this work. Beyond the gradient-based analysis, we also derive an upper bound on robustness sensitivity from the perspective of neuron firing states (Appendix A).

Analysis of Membrane Potential Distribution

Equation 9 shows that regularizing the SG terms suppresses fluctuations in robustness error. As defined in Equation 7, the SG magnitude depends on the SG function and the membrane potential ($\overline{U}$). Therefore, analyzing the membrane potential distribution (MPD) is a prerequisite for understanding how SG affects SNN robustness. In forward propagation, the MPD is constrained by the threshold-dependent batch normalization (tdBN) (Zheng et al. 2021) and the dynamics of LIF neurons. tdBN first transforms the postsynaptic current I into a Gaussian-distributed variable $\overline{I}$ using the scaling factors (α and V_{th}) and a learnable affine transformation (λ and β). The dynamics of LIF neurons reveal that the MPD is governed by the decay factor (τ), leading to a shift and scaling relative to the normalized postsynaptic current $\overline{I}$. Based on these observations, we propose Theorem 1 to explain the detailed dynamic distribution of membrane potential.

Theorem 1. *In an iterative LIF model with decay factor τ over T timesteps, the tdBN-normalized postsynaptic input follows the distribution $\overline{I} \sim \mathcal{N}(\beta_c, (\lambda_c \alpha V_{th})^2)$. For $t = 1, 2, 3, ..., T$, the membrane potentials is distributed as $\overline{U}_c^l(t) \sim \mathcal{N}(\beta_c D(\tau, t) - S(t), (\lambda_c \alpha V_{th})^2 D(\tau^2, t))$, where c is the channel number of the tdBN layer, $D(\tau, t) = \sum_{i=1}^{t} \tau^{t-i}$ is the cumulative decay function, and $S(t)$ is the cumulative response strength constant.*

Proof. The proof of Theorem 1 is included in Appendix A. □

The relationship between MPD and SG

In SG methods, the SG function determines the gradient-available interval of the membrane potential and provides an approximate gradient. The membrane potential distribution is modeled as a Gaussian distribution according to Theorem 1, where gradient information is primarily carried by the membrane potential within the overlap between this distribution and the gradient-available interval of the SG function. A smaller overlap obstructs gradient propagation, while an excessively large overlap introduces numerous inaccurate approximate gradients, increasing the deviation from the true gradients (See Appendix B for a detailed explanation). To optimize SNNs more effectively, much of the current work focuses on identifying the optimal overlap areas by adjusting the MPD (Guo et al. 2022a,b) or the gradient-available interval (Lian et al. 2023; Wang et al. 2023). Their key objective is to ensure that the proportion of membrane potentials with gradients is appropriate, allowing for stable gradient propagation and reliable training. However, the SG not only plays a pivotal role in network training but also significantly impacts the robustness error induced by perturbations (P_3 term in Equation 9). Therefore, to achieve reliable model output, the SG magnitude needs to be appropriately reduced while avoiding interference with network training, which motivates us to impose thoughtful constraints on the SG magnitude to ensure training and robust performance.

The SG magnitude is determined by the overlap area, so we first derive the expression for the overlap area to represent the SG magnitude. We assume that the SG function satisfies Equation 7, with its gradient-available interval defined as $[-\gamma, \gamma]$. The MPD follows a Gaussian distribution $\overline{U}^l(t) \sim \mathcal{N}(\mu, \sigma^2)$, with the probability density function (PDF) given by $p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$. The overlap area is defined as the integral of the Gaussian distribution over the interval $[-\gamma, \gamma]$:

$$\Omega = \int_{-\gamma}^{\gamma} p(x)dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\gamma}^{\gamma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx \quad (10)$$

To simplify the integral, we perform a change of variables. Let $z = \frac{x-\mu}{\sigma}$, so that $dx = \sigma dz$. Under this substitution, we obtain:

$$\Omega = \int_{\frac{\mu-\gamma}{\sigma}}^{\frac{\mu+\gamma}{\sigma}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz \quad (11)$$

The integral can be evaluated using the cumulative distribution function (CDF) of the standard normal distribution:

$$\Phi(x) = \frac{1}{2}\left[1 + \text{erf}(\frac{x}{\sqrt{2}})\right] \quad (12)$$

where $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ is the error function. Using this CDF, the target overlap Ω can be written as:

$$\Omega = \Phi\left(\frac{\mu+\gamma}{\sigma}\right) - \Phi\left(\frac{\mu-\gamma}{\sigma}\right) \quad (13)$$

When the SG function parameters γ is fixed, the value of Ω is controlled by the mean μ and standard deviation σ of membrane potential. The gradient of Ω with respect to the parameters of the membrane potential distribution is computed as follows:

$$\frac{\partial \Omega}{\partial \mu} = \frac{1}{\sigma}\left[p\left(\frac{\mu^+}{\sigma}\right) - p\left(\frac{\mu^-}{\sigma}\right)\right] \quad (14)$$

$$\frac{\partial \Omega}{\partial \sigma} = \frac{1}{\sigma^2}\left[-\mu^+ p\left(\frac{\mu^+}{\sigma}\right) + \mu^- p\left(\frac{\mu^-}{\sigma}\right)\right] \quad (15)$$

where $\mu^+ = \mu + \gamma$ and $\mu^- = \mu - \gamma$ are the upper and lower bounds of the overlap area, respectively.

Therefore, Equation 13 can serve as a SG regularizer, reducing the SG magnitude by constraining the parameters of the MPD (Figure 1). In practice, except for the last linear output layer, the SG is penalized at every timestamp in every channel for each layer in the training phase. Specifically, $\mathcal{L}_{MPD-SGR}$ for the b-th batch of input is given by:

$$\mathcal{L}^b_{MPD-SGR} = \frac{1}{LCT} \sum_{l,c,t} \Omega_c^{b,l}(t) \quad (16)$$

$$= \frac{1}{LCT} \sum_{l,c,t} \left[\Phi\left(\frac{\mu_c^l(t)+\gamma}{\sigma_c^l(t)}\right) - \Phi\left(\frac{\mu_c^l(t)-\gamma}{\sigma_c^l(t)}\right)\right] \quad (17)$$

Methods	CIFAR-10				CIFAR-100			
	Clean	FGSM	PGD	BIM	Clean	FGSM	PGD	BIM
Vanilla Training								
REG (Ding et al. 2022)	**92.49**	25.18	0.88	0.60	**72.82**	10.14	0.27	0.31
StoG (Ding et al. 2024b)	91.64	16.22	0.28	0.12	72.22	5.92	0.26	0.20
DLIF (Ding et al. 2024a)	92.01	11.52	0.08	0.06	71.38	7.20	0.08	0.08
FEEL (Xu et al. 2024)	90.08	29.17	6.67	5.99	70.06	9.74	2.06	1.92
SR (Liu et al. 2024)	91.04	31.72	8.55	7.28	66.76	16.16	8.00	6.43
Ours	91.63	**47.59**	**20.55**	**16.85**	70.42	**34.51**	**9.03**	**8.41**
Improvement	-0.86	+15.87	+12	+9.57	-2.4	+18.35	+1.03	+1.98
Adversarial Training								
RAT (Ding et al. 2022)	**91.41**	45.00	22.95	20.80	69.43	19.07	9.23	8.41
StoG (Ding et al. 2024b)	90.13	45.75	27.74	26.32	69.24	19.64	9.77	3.23
DLIF (Ding et al. 2024a)	88.94	39.21	27.17	25.98	67.08	19.34	9.96	9.39
FEEL (Xu et al. 2024)	89.00	45.62	29.52	28.39	68.05	19.55	12.11	11.97
SR (Liu et al. 2024)	88.26	44.28	28.63	27.03	61.26	23.10	17.07	16.28
Ours	90.69	**59.27**	**33.38**	**32.61**	**69.56**	**39.45**	**22.23**	**19.45**
Improvement	-0.72	+13.52	+3.86	+4.22	+0.13	+16.35	+5.16	+3.17

Table 1: Compare with state-of-the-art work on adversarial robustness of SNN (VGG11, $T = 8$)

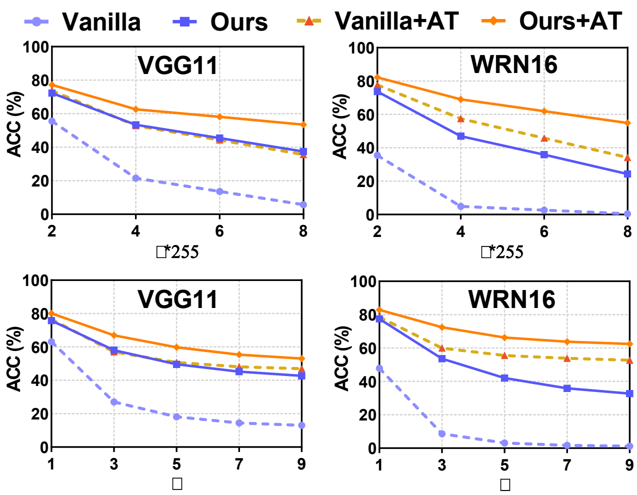

Figure 2: Performance of the white-box PGD attack with increasing perturbation ϵ and iterative step $k = 4$ (Top Panels), increasing iterative step k and $\epsilon = 8/255$ (Bottom Panels).

where L is the number of layers, C is the number of channels, and T is the number of time step. $\Omega_c^l(t)$ is the overlap area of the membrane potential distribution at the channel c of the layer l at time t. Finally, taking classification loss into consideration, the total loss can be written as:

$$\mathcal{L}^b = \mathcal{L}^b_{task} + \eta \mathcal{L}^b_{MPD-SGR} \tag{18}$$

where $\mathcal{L}^b_{task}$ denotes the task loss (e.g., cross-entropy), and η is a coefficient that modulates the strength of the regularization. The training evolution of the MPD and the SG overlap (Ω), as well as an analysis of η are provided in Appendix C.

Experiments

Experimental settings

We conduct experiments to evaluate our proposed method on image classification tasks using the CIFAR-10, CIFAR-100 and Tiny-ImageNet (Le and Yang 2015) datasets. The SNN models are VGG (VGG11) (Simonyan and Zisserman 2014) and WideResNet (WRN16) (Zagoruyko 2016). The LIF neuron parameters and the SG function settings are consistent with previous work (Ding et al. 2022).

We evaluate model robustness under three attack scenarios: white-box, black-box and non-gradient attacks. The adversarial examples are generated by four attack methods using differentiable approximation techniques (BPTT or BPTR (Bu et al. 2023)): FGSM (Goodfellow, Shlens, and Szegedy 2014), PGD (Madry 2017), BIM (Kurakin, Goodfellow, and Bengio 2018) and CW (Carlini and Wagner 2017). The attack perturbation strength is set to $\epsilon = 8/255$, the iterative steps $k = 7$ and step size $\alpha = 0.01$ for PGD and BIM. For adversarial training (AT), models are trained with white-box PGD adversarial examples ($k = 2$, $\epsilon = 2/255$). The bold values in all tables represent the optimal results for each setting. Full details on experimental settings and attack algorithms are provided in Appendix D.

Performance for various attack types

Comparison with state-of-the-art work. We evaluate MPD-SGR against state-of-the-art (SOTA) adversarial defense methods for SNNs including REG (Ding et al. 2022), StoG (Ding et al. 2024b), DLIF (Ding et al. 2024a), SR (Liu et al. 2024) and FEEL (Xu et al. 2024), under both vanilla and adversarial training (AT) settings. As shown in Table 1, MPD-SGR demonstrates superior robustness across various attacks and datasets, outperforming all SOTA methods. Under vanilla training, it significantly surpasses the SOTA baseline (SR) by 15.87%, 12.00%, and 9.57% on CIFAR-10, and by 18.35%, 1.03% (PGD), and 1.98% (BIM) on CIFAR-

Dataset	Model	Clean	FGSM	PGD	BIM	CW
CIFAR10	VGG11, BPTT	**92.45**/90.62	23.86/**42.12**	0.87/**17.18**	0.60/**14.59**	6.26/**18.85**
	VGG11, BPTR		27.92/**47.61**	7.82/**25.52**	6.61/**25.00**	24.38/**38.07**
	VGG11, BPTT, AT	**91.08**/90.34	43.80/**58.71**	20.63/**28.94**	18.63/**24.74**	29.87/**34.38**
	VGG11, BPTR, AT		49.11/**66.17**	35.74/**43.12**	34.59/**41.17**	58.73/**65.05**
	WRN16, BPTT	**93.46**/92.22	18.81/**45.82**	0.02/**8.44**	0.03/**6.59**	3.61/**18.66**
	WRN16, BPTR		15.28/**38.18**	0.17/**8.03**	0.16/**6.65**	11.76/**31.21**
	WRN16, BPTT, AT	91.15/**91.34**	42.31/**63.32**	19.93/**38.11**	18.03/**33.13**	29.60/**43.06**
	WRN16, BPTR, AT		51.75/**70.93**	34.44/**46.91**	33.00/**44.79**	58.90/**70.10**
CIFAR100	VGG11, BPTT	**72.54**/69.56	8.95/**20.16**	0.24/**6.90**	0.20/**5.51**	5.62/**13.54**
	VGG11, BPTR		11.45/**34.49**	4.66/**10.07**	4.44/**9.94**	25.03/**26.99**
	VGG11, BPTT, AT	**68.60**/67.67	20.28/**36.18**	9.97/**18.23**	8.82/**15.70**	15.46/**22.33**
	VGG11, BPTR, AT		29.78/**40.42**	22.14/**26.38**	21.27/**25.88**	40.96/**47.44**
	WRN16, BPTT	**73.86**/72.61	10.99/**22.58**	0.04/**1.97**	0.05/**1.77**	4.93/**10.34**
	WRN16, BPTR		9.74/**21.60**	0.44/**5.52**	0.35/**5.50**	14.81/**24.13**
	WRN16, BPTT, AT	69.29/**69.67**	25.35/**39.47**	12.09/**21.71**	11.30/**18.87**	20.17/**27.55**
	WRN16, BPTR, AT		31.99/**44.23**	22.58/**35.67**	22.04/**35.08**	43.23/**53.21**

Table 2: The classification accuracy (Vanilla/Ours) under white-box attacks across multiple datasets and architectures ($T = 4$).

Methods	Clean	Gaussian Noise	Uniform Noise
REG	72.82	24.73 (-48.09)	49.86 (-23.86)
FEEL	70.06	32.63 (-37.43)	52.47 (-17.59)
SR	66.76	47.47 (-19.29)	61.37 (**-5.39**)
Ours	70.42	**53.01 (-17.41)**	**64.21** (-6.21)

Table 3: Comparative evaluation of defense mechanisms under random perturbation attacks on CIFAR100 datasets with VGG11 ($T = 8$). Values in parentheses denote accuracy degradation relative to clean performance.

100 under FGSM, PGD, and BIM attacks, respectively. With AT, MPD-SGR maintains superior robustness, exceeding the SOTA baseline (FEEL) by 13.52%, 3.86%, and 4.22% on CIFAR-10, and the SOTA baseline (SR) by 16.35%, 5.16%, and 3.17% on CIFAR-100 for FGSM, PGD, and BIM attacks, respectively.

Current adversarial defense mechanisms enhance model robustness against perturbations, yet often incur a degradation in clean accuracy—a trade-off particularly pronounced in AT frameworks and consistent with findings in prior literature (Wu et al. 2024). While SR achieves relatively strong robustness under AT, its clean accuracy declines markedly to 61.26% on CIFAR-100 and 88.26% on CIFAR-10, indicating a significant compromise in nominal performance. In contrast, MPD-SGR achieves a more favorable balance, demonstrating its practical advantage.

White-box attack. We compare the classification accuracy of MPD-SGR with the vanilla method (REG/RAT) under various white-box attacks across VGG11/WRN16 networks and CIFAR-10/100 datasets. As summarized in Table 2, the MPD-SGR method significantly enhances the robustness of the vanilla SNN model against all tested attacks, with consistent gains observed under both BPTT and BPTR methods across all datasets and architectures. Notably, without AT, the vanilla SNN completely fails under strong iterative attacks such as PGD and BIM, with accuracy dropping close to 0%. In contrast, the MPD-SGR method maintains a certain level of performance, achieving an accuracy of approximately 10%.

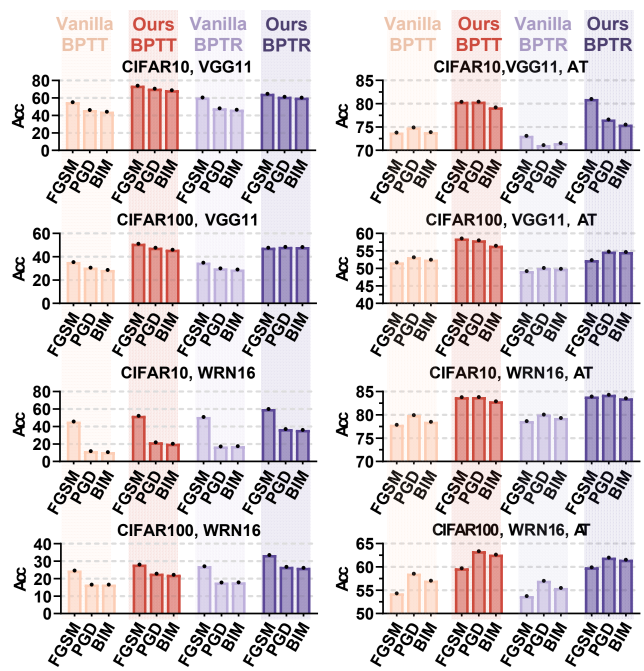

Figure 3: Performance of the proposed MPD-SGR method under different black-box attacks.

We evaluate how model accuracy degrades under increasing PGD attack intensity ϵ and number of iterations k on the CIFAR-10 dataset. The results in Figure 2 show that the MPD-SGR exhibits a slower accuracy decline than the base-

Model+SG	Methods	Clean	FSGM	PGD	BIM
VGG11+Rec	REG	**91.85**	24.00	3.13	2.33
	Ours	91.23	**43.28**	**15.82**	**14.2**
VGG11+Sig	REG	**92.15**	19.42	0.24	0.15
	Ours	89.38	**37.25**	**9.26**	**7.23**
VGG11+Sup	REG	**86.82**	21.39	0.82	0.50
	Ours	84.45	**43.42**	**6.32**	**4.50**
WRN16+Rec	REG	**92.94**	24.13	0.08	0.09
	Ours	92.13	**49.51**	**18.15**	**13.83**
WRN16+Sig	REG	**92.50**	16.80	0.02	0.01
	Ours	91.44	**36.30**	**8.77**	**7.05**
WRN16+Sup	REG	**86.57**	11.54	0.05	0.05
	Ours	86.54	**39.25**	**7.17**	**4.74**

Table 4: Performance (%) of the proposed MPD-SGR method under three different SG functions.

Dataset	Model	Methods	Clean	FGSM	PGD
Tiny-ImageNet	VGG16	DIR	**57.90**	2.04	0.01
	VGG16	DIR+Ours	54.78	**14.33**	**5.72**
	VGG16	POS	**48.14**	6.79	2.68
	VGG16	POS+Ours	47.83	**20.42**	**8.21**
	VGG16	RSC	**47.47**	22.63	13.75
	VGG16	RSC+Ours	46.98	**35.06**	**17.60**

Table 5: Performance (%) of the proposed MPD-SGR method under three different spike coding methods.

line under stronger PGD attacks for both VGG and WRN architectures, regardless of whether AT is used. This demonstrates that our method exhibits strong tolerance to more intense adversarial perturbations. For additional results about white-box attacks, including varying T and different attack settings, please refer to Appendix E.

Black-box attack. We train a substitute SNN model to generate adversarial examples that are transferable to both the vanilla and MPD-SGR models for black-box attacks. This substitute SNN is trained without any regularization on the same dataset, using identical neuron parameters and architecture as the target model. Figure 3 highlights the performance of the MPD-SGR method across various black-box attacks. We observe that black-box attacks are less effective than white-box attacks, and the MPD-SGR method also achieves comparable robustness improvements under black-box attacks as in the white-box setting, suggesting its robustness stems not from gradient obfuscation but from inherent properties of the method. For detailed gradient obfuscation analysis, please refer to the Appendix F.

Non-gradient attack. We further evaluate the impact of MPD-SGR on the robustness of SNNs under non-gradient attacks, specifically two types of random perturbations: Gaussian ($\delta \sim \mathcal{N}(0, \epsilon)$) and Uniform ($\delta \sim \mathcal{U}(-\epsilon, \epsilon)$) noise. All results are reported using a fixed noise intensity of $\epsilon = 0.1$. As shown in Table 3, our method maintains high classification performance under both perturbation types, significantly outperforming the REG baseline, which suffers severe performance degradation (accuracy drops of 48.09% and 23.86%). Although SR exhibits a degree of robustness, its substantially lower clean accuracy (66.76%) fundamentally constrains its performance under noisy conditions, resulting in absolute accuracies below those achieved by our approach. These results indicate that MPD-SGR exhibits strong robustness to non-gradient perturbations, showcasing its versatility across diverse input disturbances.

Ablation study

Performance under different SG functions. We investigate the applicability of our method to other commonly used SG functions, including Rectangular (Rec) (Wu et al. 2019), Sigmoid (Sig) (Roy, Chakraborty, and Roy 2019) and Superspike (Sup) (Zenke and Ganguli 2018). As presented in Table 4, our proposed MPD-SGR method consistently enhances robustness across different SG functions, further confirming the versatility and effectiveness of our method.

Performance under different spike coding. Considering the significant impact of input spike encoding methods on the robustness of SNNs, we also compare the performance of MPD-SGR with that of the vanilla SNN model under different spike coding methods, including Direct Coding (DIR), Poisson Coding (POS) and Randomized Smoothing Coding (RSC) (Wu et al. 2024). The proposed MPD-SGR method exhibits consistently superior performance over the baseline SNN model across all tested spike coding schemes (Table 5), demonstrating the compatibility of our method with various spike coding approaches.

Conclusion

This paper presents a theoretical and empirical investigation into the role of neuronal membrane potential distribution (MPD) and surrogate gradient (SG) mechanisms in enhancing the adversarial robustness of deep spiking neural networks (SNNs). Through rigorous gradient analysis, we establish a theoretical framework that formally connects input perturbation sensitivity to the magnitude characteristics of SG in SNNs. Our analysis reveals that reducing the proportion of membrane potentials within the gradient-available range of the SG function effectively mitigates the sensitivity of SNNs to input perturbations. Based on these theoretical insights, we propose MPD-driven surrogate gradient regularization (MPD-SGR), which systematically constrains the MPD during training to optimize gradient flow, thereby enhancing the network's resilience against adversarial attacks. Extensive experiments across multiple benchmark datasets and SNN architectures show that MPD-SGR maintains comparable clean accuracy while consistently improving robustness against gradient-based attacks and random noise, outperforming existing defense methods. The proposed approach also demonstrates remarkable adaptability across diverse network configurations, SG function variants, and spike coding schemes. These results position MPD-SGR as a promising, generalizable defense strategy for SNNs under diverse perturbation scenarios.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62236007, 32441113, and 62276235). The authors would also like to acknowledge anonymous reviewers and chairs for providing insightful comments to help improve this work.

References

Bu, T.; Ding, J.; Hao, Z.; and Yu, Z. 2023. Rate gradient approximation attack threats deep spiking neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7896–7906.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 IEEE symposium on security and privacy (sp)*, 39–57. IEEE.

Chowdhury, S. S.; Lee, C.; and Roy, K. 2021. Towards understanding the effect of leak in spiking neural networks. *Neurocomputing*, 464: 83–94.

Ding, J.; Bu, T.; Yu, Z.; Huang, T.; and Liu, J. 2022. Snn-rat: Robustness-enhanced spiking neural network through regularized adversarial training. *Advances in Neural Information Processing Systems*, 35: 24780–24793.

Ding, J.; Pan, Z.; Liu, Y.; Yu, Z.; and Huang, T. 2024a. Robust Stable Spiking Neural Networks. In *Forty-first International Conference on Machine Learning*. PMLR.

Ding, J.; Yu, Z.; Huang, T.; and Liu, J. K. 2024b. Enhancing the robustness of spiking neural networks with stochastic gating mechanisms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 492–502.

El-Allami, R.; Marchisio, A.; Shafique, M.; and Alouani, I. 2021. Securing deep spiking neural networks against adversarial attacks through inherent structural parameters. In *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 774–779. IEEE.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Gu, P.; Xiao, R.; Pan, G.; and Tang, H. 2019. STCA: Spatio-temporal credit assignment with delayed feedback in deep spiking neural networks. In *IJCAI*, volume 15, 1366–1372.

Guo, Y.; Chen, Y.; Zhang, L.; Wang, Y.; Liu, X.; Tong, X.; Ou, Y.; Huang, X.; and Ma, Z. 2022a. Reducing information loss for spiking neural networks. In *European Conference on Computer Vision*, 36–52. Springer.

Guo, Y.; Liu, X.; Chen, Y.; Zhang, L.; Peng, W.; Zhang, Y.; Huang, X.; and Ma, Z. 2023. Rmp-loss: Regularizing membrane potential distribution for spiking neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17391–17401.

Guo, Y.; Tong, X.; Chen, Y.; Zhang, L.; Liu, X.; Ma, Z.; and Huang, X. 2022b. Recdis-snn: Rectifying membrane potential distribution for directly training spiking neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 326–335.

Hu, Y.; Zheng, Q.; Li, G.; Tang, H.; and Pan, G. 2024. Toward Large-scale Spiking Neural Networks: A Comprehensive Survey and Future Directions. *arXiv preprint arXiv:2409.02111*.

Kim, Y.; Park, H.; Moitra, A.; Bhattacharjee, A.; Venkatesha, Y.; and Panda, P. 2022. Rate coding or direct coding: Which one is better for accurate, robust, and energy-efficient spiking neural networks? In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 71–75. IEEE.

Kim, Y.; Venkatesha, Y.; and Panda, P. 2022. PrivateSNN: Privacy-Preserving Spiking Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1): 1192–1200.

Kundu, S.; and Pedram, M. 2021. HIRE-SNN: Harnessing the adversarial robustness of energy-efficient deep spiking neural networks via training with crafted input noise. In *ICCV*.

Kurakin, A.; Goodfellow, I. J.; and Bengio, S. 2018. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, 99–112. Chapman and Hall/CRC.

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7): 3.

Li, Y.; Guo, Y.; Zhang, S.; Deng, S.; Hai, Y.; and Gu, S. 2021. Differentiable spike: Rethinking gradient-descent for training spiking neural networks. *Advances in Neural Information Processing Systems*, 34: 23426–23439.

Lian, S.; Shen, J.; Liu, Q.; Wang, Z.; Yan, R.; and Tang, H. 2023. Learnable Surrogate Gradient for Direct Training Spiking Neural Networks. In *IJCAI*, 3002–3010.

Liang, L.; Hu, X.; Deng, L.; Wu, Y.; Li, G.; Ding, Y.; Li, P.; and Xie, Y. 2021. Exploring adversarial attack in spiking neural networks with spike-compatible gradient. *IEEE transactions on neural networks and learning systems*, 34(5): 2569–2583.

Liu, Y.; Bu, T.; Ding, J.; Hao, Z.; Huang, T.; and Yu, Z. 2024. Enhancing Adversarial Robustness in SNNs with Sparse Gradients. In *International Conference on Machine Learning*, 30738–30754. PMLR.

Ma, G.; Yan, R.; and Tang, H. 2023. Exploiting noise as a resource for computation and learning in spiking neural networks. *Patterns*, 4(10).

Maass, W. 1997. Networks of spiking neurons: the third generation of neural network models. *Neural networks*, 10(9): 1659–1671.

Madry, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Marchisio, A.; Nanfa, G.; Khalid, F.; Hanif, M. A.; Martina, M.; and Shafique, M. 2020. Is spiking secure? a comparative study on the security vulnerabilities of spiking and deep neural networks. In *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Qin, C.; Martens, J.; Gowal, S.; Krishnan, D.; Dvijotham, K.; Fawzi, A.; De, S.; Stanforth, R.; and Kohli, P. 2019. Adversarial robustness through local linearization. *Advances in neural information processing systems*, 32.

Roy, D.; Chakraborty, I.; and Roy, K. 2019. Scaling deep spiking neural networks with binary stochastic activations. In *2019 IEEE international conference on cognitive computing (ICCC)*, 50–58. IEEE.

Sharmin, S.; Panda, P.; Sarwar, S. S.; Lee, C.; Ponghiran, W.; and Roy, K. 2019. A comprehensive analysis on adversarial robustness of spiking neural networks. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.

Sharmin, S.; Rathi, N.; Panda, P.; and Roy, K. 2020. Inherent adversarial robustness of deep spiking neural networks: Effects of discrete input encoding and non-linear activations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, 399–414. Springer.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tavanaei, A.; Ghodrati, M.; Kheradpisheh, S. R.; Masquelier, T.; and Maida, A. 2019. Deep learning in spiking neural networks. *Neural networks*, 111: 47–63.

Wang, Q.; Yang, X.; Chen, H.; and Geng, X. 2024a. Vision Transformers as Probabilistic Expansion from Learngene. In *Forty-first International Conference on Machine Learning*.

Wang, Q.; Yang, X.; Feng, F.; Wang, J.; and Geng, X. 2024b. Cluster-Learngene: Inheriting Adaptive Clusters for Vision Transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wang, Q.-F.; Geng, X.; Lin, S.-X.; Xia, S.-Y.; Qi, L.; and Xu, N. 2022. Learngene: From open-world to your learning task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8557–8565.

Wang, Z.; Jiang, R.; Lian, S.; Yan, R.; and Tang, H. 2023. Adaptive smoothing gradient learning for spiking neural networks. In *International Conference on Machine Learning*, 35798–35816. PMLR.

Wu, K.; Yao, M.; Chou, Y.; Qiu, X.; Yang, R.; Xu, B.; and Li, G. 2024. RSC-SNN: Exploring the Trade-off Between Adversarial Robustness and Accuracy in Spiking Neural Networks via Randomized Smoothing Coding. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2748–2756.

Wu, Y.; Deng, L.; Li, G.; Zhu, J.; and Shi, L. 2018. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12: 331.

Wu, Y.; Deng, L.; Li, G.; Zhu, J.; Xie, Y.; and Shi, L. 2019. Direct training for spiking neural networks: Faster, larger, better. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 1311–1318.

Xu, M.; Ma, D.; Tang, H.; Zheng, Q.; and Pan, G. 2024. FEEL-SNN: Robust Spiking Neural Networks with Frequency Encoding and Evolutionary Leak Factor. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zagoruyko, S. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zenke, F.; and Ganguli, S. 2018. Superspike: Supervised learning in multilayer spiking neural networks. *Neural computation*, 30(6): 1514–1541.

Zenke, F.; and Vogels, T. P. 2021. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. *Neural computation*, 33(4): 899–925.

Zheng, H.; Wu, Y.; Deng, L.; Hu, Y.; and Li, G. 2021. Going deeper with directly-trained larger spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, 11062–11070.

Zhu, R.-J.; Wang, Z.; Gilpin, L.; and Eshraghian, J. 2024. Autonomous driving with spiking neural networks. *Advances in Neural Information Processing Systems*, 37: 136782–136804.