Bi-Spectrum Distillation: Addressing Spectral Mismatch in ANN-SNN Knowledge Transfer

Yuxuan Zhang[1], **Yuhang Sun**[1, 2], **Wen Yao**[3], **Yue Deng**[1, 2], **Hongjue Li**[1*]
[1]Beihang University
[2]Beijing Zhongguancun Academy
[3]Defense Innovation Institute, Chinese Academy of Military Science
{zyxuaan, sunyuhang, ydeng, lihongjue}@buaa.edu.cn, wendy0782@126.com

Abstract

Knowledge distillation from Artificial Neural Networks (ANNs) to Spiking Neural Networks (SNNs) is a prominent training paradigm. However, its efficacy is fundamentally limited by a spectral mismatch: SNNs, with their intrinsic low-pass filtering characteristics, struggle to learn high-frequency details from their ANN teachers, creating a bottleneck in knowledge transfer at both the feature and logit levels. To address this, we propose Bi-Spectrum Distillation (BSD), a novel framework that mitigates the mismatch from two complementary perspectives. First, at the feature level, our Spectral Residual Distillation (SRD) enhances the student SNN's features with a parameter-efficient, learnable filter that adaptively compensates for high-frequency information loss, which transforms the student's output to better match the teacher's rich spectral target. Second, at the logits level, our Spectral Semantic Distillation (SSD) enhances fine-grained classification by distilling high-frequency components from teacher-ordered logits. Extensive experiments on CIFAR-10/100, ImageNet, and CIFAR10-DVS demonstrate that BSD achieves new state-of-the-art performance across both CNN and Transformer-based SNNs, validating its effectiveness and broad applicability.

Introduction

Spiking Neural Networks (SNNs) are brain-inspired models prized for their energy efficiency (Maass 1997). Unlike conventional Artificial Neural Networks (ANNs) that rely on power-intensive multiply-accumulate (MAC) operations, SNNs communicate via sparse, binary spikes, using simpler additions (Roy, Jaiswal, and Panda 2019). This event-driven paradigm makes them ideal for neuromorphic hardware (Davies et al. 2018; Deng et al. 2020), promising significant gains in computational efficiency.

Despite their promise, Spiking Neural Networks (SNNs) are notoriously difficult to train due to the non-differentiable nature of their spike events (Eshraghian et al. 2023). Early solutions followed two main paths. Conversion-based methods transferred knowledge by mapping pre-trained ANN weights to SNNs (Hao et al. 2023; Hu et al. 2023; Bu et al. 2023), while learning-based methods enabled direct training via surrogate gradients (SG) (Neftci, Mostafa, and Zenke 2019; Wu et al. 2018). Building upon these foundations, knowledge distillation (KD) (Hinton, Vinyals, and Dean 2015) has emerged as a more advanced strategy. By leveraging a powerful ANN teacher to guide an SNN student, KD has set the current state of the art in both performance and efficiency (Xu et al. 2023; Hong et al. 2023).

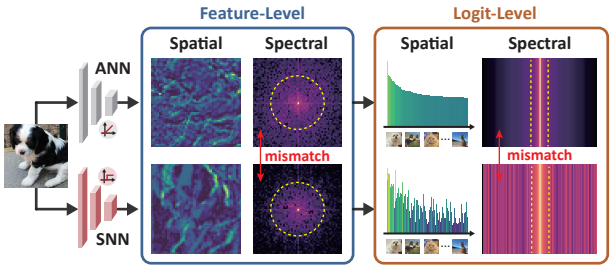

Figure 1: Visualization of the ANN-SNN Spectral Mismatch. At the feature level, the SNN output appears smoother, with a noticeable loss of high-frequency energy compared to the ANN. At the logit level, following the sorting of logits into a meaningful order, the SNN exhibits high-frequency "jitter" absent in the teacher's smooth distribution. These observations highlight a distinct mismatch in high-frequency information between the two networks.

However, the effectiveness of this approach is fundamentally constrained by a challenge we identify as **spectral mismatch**. As illustrated in Figure 1, ANN teachers, using activations like ReLU, can represent a broad spectrum of feature frequencies (Xu et al. 2020). In contrast, student SNNs, commonly built with Leaky Integrate-and-Fire (LIF) neurons, inherently function as low-pass filters (Fang et al. 2025). This discrepancy prevents the SNN student from effectively learning high-frequency details from the teacher's features, creating a critical bottleneck that impedes knowledge transfer.

To address this spectral mismatch, we propose the **Bi-Spectrum Distillation (BSD)** framework, which tackles the problem from two complementary perspectives: At the feature level, **Spectral Residual Distillation (SRD)** introduces a parameter-efficient filter that learns to selectively amplify high-frequency components, directly compensating for the

*Corresponding author.

SNN's intrinsic low-pass nature. At the logit level, **Spectral Semantic Distillation (SSD)** first addresses the unstructured nature of the output by sorting the logits. This creates an ordered representation where frequency analysis becomes meaningful, enabling the distillation of the teacher's high-frequency spectral details to boost the SNN's fine-grained classification performance. Extensive experiments validate the proposed BSD framework, demonstrating that it achieves new state-of-the-art results across diverse static and neuromorphic benchmarks for both CNN and Transformer architectures.

Our contributions are threefold:

- We identify and formulate the spectral mismatch as a core, predictable obstacle in ANN-to-SNN distillation, shifting the perspective from unexplained performance loss to a well-defined frequency-domain problem.
- We propose the Bi-Spectrum Distillation (BSD) framework, a dual-pronged solution that addresses the mismatch at both the feature and logit levels with novel, frequency-aware modules.
- Our method achieves new state-of-the-art results on major benchmarks, including CIFAR-10/100, ImageNet and CIFAR10-DVS across both CNN and Transformer architectures, demonstrating its effectiveness and broad applicability.

Related Work

Knowledge Distillation for SNNs. Knowledge Distillation (KD) has become a key approach for training high-performance SNNs by transferring knowledge from pre-trained ANN teachers (Guo et al. 2024b; Yang et al. 2025). Early methods focused on aligning intermediate features in the spatial domain. For instance, KDSNN (Xu et al. 2023) employed both response-based and feature-based distillation, while LaSNN (Hong et al. 2023) introduced layer-wise attention to bridge the representational gap between ANNs and SNNs. Subsequent approaches, such as SAKD (Qiu et al. 2024) and BKDSNN (Xu et al. 2024b), refined the distillation process by incorporating feature and logit alignment, with BKDSNN address the discrete-continuous mismatch by restoring blurred features from the final layer. Other works explored temporal aspects of SNNs (Dong, Zhao, and Zeng 2024; Yu et al. 2025), decoupling distillation objectives across timesteps or implementing temporal self-distillation for knowledge sharing within the SNN. While these methods have significantly advanced SNN training, they predominantly operate in the spatial domain and implicitly assume that SNNs can replicate the full-spectrum features of the teacher. Our work builds upon this foundation but addresses the underlying spectral mismatch that is not accounted for in existing KD methods. Specifically, we introduce a frequency-aware approach within the feature- and logits-level distillation process to resolve this spectral mismatch.

Frequency Learning in SNNs. Recent work emphasizes the importance of frequency-domain analysis for SNNs. It is well-established that Leaky Integrate-and-Fire (LIF) neurons and their variants act as low-pass filters (Gerstner et al. 2014; Lindner 2013), preferentially propagating low-frequency information while suppressing high-frequency details. This bias has been identified as a significant factor in the performance gap between SNNs and ANNs. To mitigate this, methods like Max-former (Fang et al. 2025) have introduced high-pass operators such as Max-Pooling. SWformer (Fang et al. 2024) and FEEL-SNN (Xu et al. 2024a) use wavelet transforms and frequency encoding to enhance feature learning and robustness. While these methods focus on high-frequency learning at the logit level or modify network architecture, our work directly addresses the spectral mismatch by integrating frequency-aware distillation in both the feature and logit levels.

Preliminary: The LIF Neuron

The Leaky Integrate-and-Fire (LIF) neuron is a powerful abstraction of its biological counterpart, widely adopted as the fundamental computational unit in deep SNNs due to its computational efficiency (Fang et al. 2021; Zhou et al. 2022, 2023a). The dynamics of an LIF neuron's membrane potential, $U(t)$, are formally described by the differential equation:

$$\tau_m \frac{dU(t)}{dt} = -(U(t) - U_{rest}) + R_m I(t), \tag{1}$$

where τ_m is the membrane time constant, U_{rest} is the resting potential, and $I(t)$ is the input synaptic current. When $U(t)$ reaches a threshold V_{th}, the neuron fires a spike and its potential is reset. For computational implementation in deep SNNs, this is typically discretized as:

$$U[n] = \beta U[n-1] + (1-\beta) I[n], \tag{2}$$

where n denotes the discrete time step and $\beta = e^{-\Delta t/\tau_m}$ is the decay factor over a time step Δt. The neuron emits a spike ($S[n] = 1$) when its potential $U[n]$ exceeds a threshold V_{th}, after which the potential is reset.

Method

In this section, we first formalize the spectral mismatch problem inherent in ANN-SNN distillation. We then detail our proposed Bi-Spectrum Distillation (BSD) framework, which is comprised of two complementary modules: Spectral Residual Distillation (SRD) for feature-level compensation and Spectral Semantic Distillation (SSD) for logit-level guidance.

The Spectral Mismatch Problem

A core challenge in ANN-SNN distillation is the spectral mismatch, caused by different frequency responses in their respective activations. We now formalize this problem and its impact on both features and logits.

The LIF Neuron as a Low-Pass Filter. The origin of the spectral mismatch lies in the intrinsic frequency response of the LIF neuron. The neuron's subthreshold membrane dynamics function as a linear filter. Its transfer function,

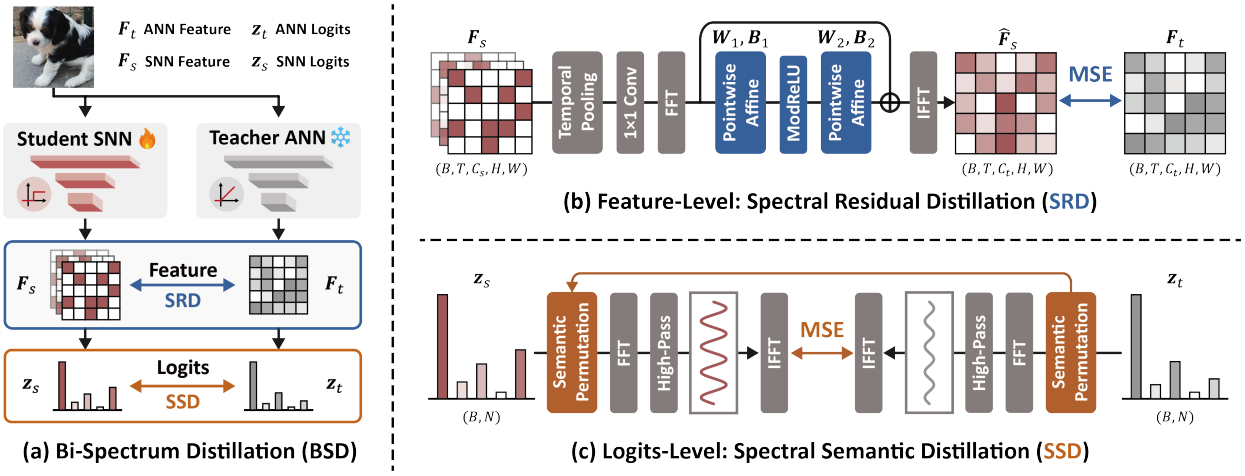

Figure 2: Overview of the Bi-Spectrum Distillation (BSD) framework. **(a)** BSD guides an SNN student with an ANN teacher via two complementary modules: SRD for feature-level distillation and SSD for logit-level distillation. **(b)** SRD operates on features ($\boldsymbol{F}_s$, $\boldsymbol{F}_t$), learning a spectral residual in the frequency domain to bridge the gap between the student and teacher. **(c)** SSD operates on logits ($\boldsymbol{z}_s$, $\boldsymbol{z}_t$), first structuring them via sorting and then distilling high-frequency details to enhance fine-grained learning.

$H_U(\omega)$, maps the input current spectrum $I(\omega)$ to the membrane potential spectrum $U(\omega)$:

$$H_U(\omega) = \frac{U(\omega)}{I(\omega)} = \frac{1-\beta}{1-\beta e^{-j\omega\Delta t}}. \tag{3}$$

Since the magnitude $|H_U(\omega)|$ decreases with frequency ω, this process suppresses high-frequency components of the input signal.

The subsequent spike generation is a non-linear operation, $S[n] = \Theta(U[n] - V_{th})$. We simplify this non-linear stage by assuming the output spectrum is dominated by the linear filter, which allows the output spike spectrum $S(\omega)$ to be approximated as:

$$S(\omega) \sim U(\omega) = |H_U(\omega)|^2 I(\omega). \tag{4}$$

Thus, the LIF neuron acts as a strong low-pass filter. Although standard ANN activation functions like ReLU exhibit some high-frequency suppression, the effect is substantially stronger in LIF neuronss (Fang et al. 2025). This difference in frequency filtering is the primary source of the spectral mismatch. We visually confirm the low-pass filtering property in the supplementary material.

The Spectral Mismatch: From Feature to Logits. The different frequency responses of ANNs and SNNs create a major challenge for feature distillation, a process that minimizes the L2 distance ($\|F_t - F_s\|^2$) between a teacher's features (F_t) and student's (F_s). By Parseval's theorem, this spatial-domain loss can be decomposed into its frequency components:

$$\begin{aligned}\mathcal{L} = \|F_t - F_s\|^2 &= \int \left\|\tilde{F}_t(\omega) - \tilde{F}_s(\omega)\right\|^2 d\omega \\ &= \int_{\Omega_{\text{low}}} \left\|\tilde{F}_t(\omega) - \tilde{F}_s(\omega)\right\|^2 d\omega \\ &\quad + \int_{\Omega_{\text{high}}} \left\|\tilde{F}_t(\omega) - \tilde{F}_s(\omega)\right\|^2 d\omega,\end{aligned} \tag{5}$$

where $\tilde{F}(\omega)$ is the Fourier transform and the spectrum is partitioned into low (Ω_{low}) and high (Ω_{high}) frequencies.

As a natural low-pass filter, an SNN's capacity to generate high-frequency signals is negligible ($\|\tilde{F}_s(\omega)\|^2 \approx 0$). Consequently, the high-frequency loss component becomes irreducible, determined almost entirely by the teacher's signal:

$$\int_{\Omega_{\text{high}}} \|\tilde{F}_t(\omega) - \tilde{F}_s(\omega)\|^2 \, d\omega \approx \int_{\Omega_{\text{high}}} \|\tilde{F}_t(\omega)\|^2 \, d\omega. \tag{6}$$

Misleading gradients from this irreducible term then attempt to force an unattainable high-frequency response from the SNN, hindering the distillation process.

This core problem at the feature level then propagates to the final predictions. Lacking the high-frequency details needed for fine-grained classification, the SNN's logits cannot match the complex structure of the teacher's outputs, creating a secondary, implicit mismatch at the logit level that ultimately lowers the model's accuracy.

Overview of Bi-Spectrum Distillation

To address the spectral mismatch at both the feature and logit levels, we propose **Bi-Spectrum Distillation (BSD)**, a framework that tackles this two-level problem with complementary components (see Figure 2). For the direct feature-level mismatch, our **Spectral Residual Distillation (SRD)**

module adaptively compensates for the SNN's lost high-frequency details, creating a more achievable distillation target. To address the subsequent logit-level mismatch, our **Spectral Semantic Distillation (SSD)** helps the SNN learn the teacher's fine-grained predictions by structuring the logits and then distilling their high-frequency semantic information.

These components guide the training process through a total loss function $\mathcal{L}_{\text{total}}$. This objective combines the standard task loss $\mathcal{L}_{\text{task}}$ and conventional distillation loss $\mathcal{L}_{\text{KD}}$ with our two proposed spectral alignment terms, $\mathcal{L}_{\text{SRD}}$ and $\mathcal{L}_{\text{SSD}}$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \alpha\mathcal{L}_{\text{KD}} + \beta\mathcal{L}_{\text{SRD}} + \gamma\mathcal{L}_{\text{SSD}}. \quad (7)$$

The hyperparameters α, β, and γ balance the influence of these terms. Typically, $\mathcal{L}_{\text{task}}$ is the Cross-Entropy loss, while $\mathcal{L}_{\text{KD}}$ is the standard, temperature-scaled KL divergence loss.

Spectral Residual Distillation for Features

To address the feature-level mismatch, we introduce Spectral Residual Distillation (SRD). The core idea is to stop forcing the SNN student to match an unattainable high-frequency target. Instead, as shown in Figure 2b, we insert a lightweight, learnable module that adaptively helps the student by compensating for its spectral deficiencies. This module learns a spectral residual ($\tilde{\boldsymbol{F}}_{\text{res}}$)—a targeted compensation in the frequency domain—which, when added to the student's features, creates a more achievable target for distillation.

The SRD module transforms student features $\boldsymbol{F}_s$ into enhanced features $\hat{\boldsymbol{F}}_s$ in three steps:

1. **Temporal Pooling and Spectral Projection.** The student's features $\boldsymbol{F}_s$ are first averaged over time and projected into the frequency domain via a 1×1 convolution and a Fast Fourier Transform (FFT), $\mathcal{F}$.

$$\tilde{\boldsymbol{F}}'_s = \mathcal{F}\left(\text{Conv}_{1\times1}\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{F}_s(t)\right)\right). \quad (8)$$

2. **Hierarchical Spectral Filtering.** The spectral residual $\tilde{\boldsymbol{F}}_{\text{res}}$ is computed from $\tilde{\boldsymbol{F}}'_s$ using a block of element-wise complex-valued operations with a modReLU nonlinearity (Trabelsi et al. 2017), denoted as σ.

$$\tilde{\boldsymbol{F}}_{\text{res}} = \boldsymbol{W}_2 \odot \sigma\left(\boldsymbol{W}_1 \odot \tilde{\boldsymbol{F}}'_s + \boldsymbol{B}_1\right) + \boldsymbol{B}_2, \quad (9)$$

   where $\odot$ denotes the Hadamard product. The learnable parameters ($\boldsymbol{W}_k, \boldsymbol{B}_k$) are channel-independent. This highly parameter-efficient design learns a single, global spectral transformation profile. Such a constraint is essential to prevent the module from overfitting and absorbing the learning signal that is intended for the SNN student.

3. **Spectral Residual Fusion and Spatial Reconstruction.** The learned residual is added to the student's spectrum, and an inverse FFT ($\mathcal{F}^{-1}$) returns the enhanced features ($\hat{\boldsymbol{F}}_s$) to the spatial domain.

$$\hat{\boldsymbol{F}}_s = \mathcal{F}^{-1}\left(\tilde{\boldsymbol{F}}'_s + \tilde{\boldsymbol{F}}_{\text{res}}\right). \quad (10)$$

The SRD loss ($\mathcal{L}_{\text{SRD}}$) combines a feature alignment term with a regularization term:

$$\mathcal{L}_{\text{SRD}} = \|\hat{\boldsymbol{F}}_s - \boldsymbol{F}_t\|^2 + \lambda\|\tilde{\boldsymbol{F}}_{\text{res}}\|^2, \quad (11)$$

The two terms in the $\mathcal{L}_{\text{SRD}}$ loss function work in tandem. The first term drives the alignment between the enhanced student features and the teacher's target. The second term, a critical L2 penalty on the residual itself, ensures the SRD module remains lightweight. By encouraging the module to learn only minimal modifications, this regularization forces the SNN student to perform the primary learning. The hyperparameter λ controls the trade-off between the alignment goal and the regularization penalty.

Spectral Semantic Distillation for Logits

While SRD addresses the feature-level mismatch, Spectral Semantic Distillation (SSD) provides complementary high-frequency guidance at the final output layer (Figure 2c). The goal is to apply direct supervision to the logits, but this is challenged by their inherent structure. The class order in a logit vector is an arbitrary, pre-defined arrangement; for instance, placing the "cat" and "dog" logits adjacent to each other provides no meaningful local structure. Consequently, a Fourier transform of such an unordered vector yields no useful information.

Our core insight is to first construct a meaningful structure. By sorting the logits according to the teacher's predictions, we transform them into structured 1D signals where proximity reflects semantic similarity. This process reveals a clear mismatch in the high-frequency details between the teacher and student (visualized in Figure 1), enabling us to directly supervise the learning of these fine-grained patterns.

Let $\boldsymbol{z}_t, \boldsymbol{z}_s \in \mathbb{R}^{B,N}$ be the teacher and student logits, respectively. The SSD process is as follows.

1. **Semantic Permutation.** For each sample in the batch, we find the permutation $\phi = (\phi(1), \phi(2), \ldots, \phi(N))$ that sorts the teacher's logits $\boldsymbol{z}_t$ in descending order, which satisfies the condition:

$$z_{t,\phi(1)} \geq z_{t,\phi(2)} \geq \cdots \geq z_{t,\phi(N)}. \quad (12)$$

   This permutation, which now encodes the teacher's semantic structure, is applied to both the teacher and student logit vectors to create structurally aligned 1D signals, denoted as $\boldsymbol{z}'_t$ and $\boldsymbol{z}'_s$:

$$\begin{aligned} \boldsymbol{z}'_t &= [z_{t,\phi(1)}, z_{t,\phi(2)}, \ldots, z_{t,\phi(N)}], \\ \boldsymbol{z}'_s &= [z_{s,\phi(1)}, z_{s,\phi(2)}, \ldots, z_{s,\phi(N)}]. \end{aligned} \quad (13)$$

2. **High-Pass Filtering.** To focus on fine-grained structural details, we filter these sorted vectors. The operation involves a FFT ($\mathcal{F}$), the application of a binary high-pass mask ($\boldsymbol{M}_{hp} \in \{0,1\}^N$), and an inverse FFT ($\mathcal{F}^{-1}$) to return to the spatial domain. Empirically, the cutoff ratio r_{hp} is configured to preserve only the top 50% of frequencies. This yields the high-frequency components of the logits:

$$\boldsymbol{z}'_{k,\text{hp}} = \mathcal{F}^{-1}(\boldsymbol{M}_{\text{hp}} \odot \mathcal{F}(\boldsymbol{z}'_k)), \text{for } k \in \{t, s\}. \quad (14)$$

Training Types	Method	Architecture	T	CIFAR-10 Top-1 Acc.(%)	CIFAR-100 Top-1 Acc.(%)
ANN-to-SNN	QCFS (Bu et al. 2023)	ResNet-20	8	89.55	55.37
Direct Training	STBP-tdBN (Zheng et al. 2021)	ResNet-19	4	92.92	-
	SEW-ResNet (Fang et al. 2021)	SEW-ResNet-19	4	93.24	70.84
	RecDis (Guo et al. 2022a)	ResNet-19	4	95.55	74.10
	TET (Deng et al. 2022)	ResNet-19	4	94.44	74.47
	GLIF (Yao et al. 2022)	ResNet-19	4	94.85	77.05
	OS (Zhu et al. 2024)	ResNet-19	4	95.20	77.86
	Spikformer (Zhou et al. 2022)	Sformer-4-384	4	95.19	77.86
	+CML (Zhou et al. 2023b)	Sformer-4-384	4	95.93	79.64
	Sformer (Zhou et al. 2023a)	Sformer-4-384	4	95.61	79.09
	+CML (Zhou et al. 2023b)	Sformer-4-384	4	95.81	79.98
	RevSFormer (Zhang and Zhang 2024)	Sformer-4-384	4	95.34	79.04
	QKFormer(Zhou et al. 2024)	Sformer-4-384	4	96.18	81.15
SNN-KD	KDSNN (Xu et al. 2023)	SEW-ResNet-19	4	94.36	74.08
		Sformer-4-384	4	95.88	80.33
	LaSNN (Hong et al. 2023)	SEW-ResNet-19	4	94.21	73.92
		Sformer-4-384	4	95.79	79.99
	SM(Deng et al. 2023)	ResNet-19	4	96.82	81.70
	BKDSNN (Xu et al. 2024b)	SEW-ResNet-19	4	94.64	74.95
		Sformer-4-384	4	96.06	81.26
	TKS (Dong, Zhao, and Zeng 2024)	ResNet-19	4	95.30	76.20
		SEW-ResNet-19	4	96.76	80.67
	SAKD (Qiu et al. 2024)	ResNet-19	4	96.06	80.10
	EnOF (Guo et al. 2024b)	ResNet-19	2	96.19	82.43
	ASNN (Yang et al. 2025)	ResNet-19	2	96.66	81.57
	TLD (Yu et al. 2025)	ResNet-19	4	96.97	**82.47**
	Ours	ResNet-19	4	**97.38**	82.36
		SEW-ResNet-19	4	**97.07**	**82.61**
		Sformer-4-384	4	**96.89**	**83.51**

Table 1: Top-1 accuracy (%) comparison with previous works on CIFAR-10 and CIFAR-100 datasets. T denotes timesteps. Best results are in bold. - indicates results not available (not reported in original article or cannot reproduce). All notations are consistent across experimental conditions for clarity.

The SSD loss ($\mathcal{L}_{\text{SSD}}$) then directly minimizes the discrepancy between these high-frequency components. It is the Mean Squared Error between the filtered logits of the teacher and student:

$$\mathcal{L}_{\text{SSD}} = \|\boldsymbol{z}'_{s,\text{hp}} - \boldsymbol{z}'_{t,\text{hp}}\|^2. \tag{15}$$

This objective specifically guides the SNN to replicate the fine-grained structural details embedded in the high-frequency spectrum of the teacher's sorted logits.

Experiments

We empirically validate our Bi-Spectrum Distillation (BSD) framework on a wide array of benchmarks. Our evaluation employs both CNN and Transformer backbones on the static datasets CIFAR-10/100 (Krizhevsky and Hinton 2009), ImageNet (Deng et al. 2009), and the neuromorphic dataset CIFAR10-DVS (Li et al. 2017), to confirm the broad applicability of our method.

Experimental Setup

The key hyperparameters for our Bi-Spectrum Distillation (BSD) framework are configured as follows. The coefficients for the logit-level distillation, α and γ, are consistently set to 0.5 across all experiments. The feature-level coefficient β is set to 7×10^{-4} for ResNet models on CIFAR and 7×10^{-5} for all other experiments. The distillation temperature τ for $\mathcal{L}_{\text{KD}}$ is 2, and the SRD regularization hyperparameter λ is 0.01.

To ensure fair comparisons, all other training settings (e.g., optimizer, learning rate schedule, and batch size) strictly follow the protocols from the BKDSNN framework or the original papers of the baseline models. Detailed configurations for each specific experiment and ablation study are provided in the supplementary materials.

All models were implemented in PyTorch with the SpikingJelly (Fang et al. 2023) library. Experiments on CIFAR-10, CIFAR-100, and CIFAR10-DVS were conducted using a single NVIDIA GeForce RTX 3090 GPU. For ImageNet, we employed distributed data parallel training on 8 NVIDIA A100 GPUs.

Benchmark Performance

Results on CIFAR-10/100. We first evaluate BSD on the CIFAR-10/100 benchmarks using ResNet-19 (Deng

Method	Architecture	T	Top-1 Acc.(%)
QCFS(2023)	VGG-16	64	72.85
FastSNN(2023)	VGG-16	7	73.00
ECMT(2024)	ViT-L/16	4	83.20
Spiking-ResNet(2021)	ResNet-18	4	62.32
	ResNet-50	4	57.66
SEW-ResNet(2021)	SEW-ResNet-18	4	63.18
	SEW-ResNet-50	4	67.78
Mformer(2024)	Sformer-8-512	4	78.81
E-SpikeFormer(2025)	Sformer-12-512	4	83.20
KDSNN(2023)	SEW-ResNet-18	4	63.42
	SEW-ResNet-50	4	67.72
LaSNN(2023)	SEW-ResNet-18	4	63.31
	SEW-ResNet-50	4	67.81
SM(2023)	ResNet-18	4	64.53
EnOF(2024)	ResNet-18	4	65.31
TKS(2024)	SEW-ResNet-50	4	70.70
SAKD(2024)	ResNet-18	4	65.37
	ResNet-50	4	70.71
BKDSNN(2024b)	SEW-ResNet-18	4	65.60
	SEW-ResNet-50	4	72.32
	Sformer-8-512	4	80.93
ASNN(2025)	ResNet-34	4	70.64
TLD(2025)	ResNet-34	4	71.04
Ours	SEW-ResNet-18	4	**66.92**
	SEW-ResNet-50	4	**73.32**
	Sformer-12-512	4	**85.33**

Table 2: Top-1 accuracy (%) comparison with previous works on ImageNet datasets.

et al. 2022), SEW-ResNet-19 (Fang et al. 2021), and Spiking Transformer (Sformer)-4-384 (Zhou et al. 2024) as student networks. The SNNs are guided by their corresponding high-performing ANN teachers: a ResNet-19 with 96.91%/83.42% accuracy on CIFAR-10/100, and a ViT-S with 96.97%/88.68% accuracy. As shown in Table 1, our method outperforms existing SNN approaches based on BPTT and previously established SNN distillation methods, achieving state-of-the-art performance. Especially, on CIFAR-100, BSD boosts the Sformer accuracy to 83.51%, a significant 2.25% gain over the strong BKDSNN baseline. The performance improvements on these foundational datasets indicate the potential benefits of addressing the spectral mismatch bottleneck.

Results on ImageNet. To assess its scalability, we evaluate BSD on the large-scale ImageNet dataset with SEW-ResNet-18/50 and Sformer-12-512 students. The SEW-ResNet models are guided by their corresponding ResNet teachers (71.5% and 80.4% accuracy, respectively), while the Sformer student is paired with a high-performing CAFormer teacher (85.5% accuracy) (Yu et al. 2023). The Sformer is initialized with pre-trained weights from E-SpikeFormer (Yao et al. 2025), which establishes a strong baseline of 83.20% accuracy. As summarized in Table 2, BSD establishes new state-of-the-art records for all tested architectures. Notably, it boosts the Sformer student's accuracy from a 83.20% to 85.33%. These substantial gains on a complex dataset highlight that addressing the spectral mismatch is critical for challenging, fine-grained recognition tasks.

Method	Architecture	T	Top-1 Acc.(%)
STBP-tdBN(2021)	ResNet-19	10	67.80
RecDis(2022a)	ResNet-19	10	72.42
Real Spike(2022b)	ResNet-19	10	72.85
Ternary Spike(2024a)	ResNet-19	10	78.40
Sformer(2023a)	Sformer-2-256	10	79.90
Sformer+CML(2023b)	Sformer-2-256	10	80.50
KDSNN(2023)*	Sformer-2-256	10	80.60
LaSNN(2023)*	Sformer-2-256	10	80.60
BKDSNN(2024b)*	Sformer-2-256	10	79.80
SAKD(2024)	ResNet-19	4	80.30
ASNN(2025)	ResNet-19	4	80.54
Ours	Sformer-2-256	10	**81.20**

Table 3: Top-1 accuracy (%) comparison with previous works on CIFAR10-DVS datasets. * indicates our reproduced results.

Results on CIFAR10-DVS. Finally, on the neuromorphic CIFAR10-DVS dataset, we evaluate BSD in an SNN-to-SNN distillation setting. In this scenario, a student SNN with a short timestep (T=10) learns from a more powerful teacher of the same architecture using a longer timestep (T=16, achieving 81.50% accuracy). As shown in Table 3, BSD enables the more efficient student to achieve a new SOTA accuracy of 81.20%, performing nearly on par with its less efficient teacher. This result suggests that the spectral mismatch principle is generalizable, highlighting BSD's potential to bridge the "temporal-spectral" gap created by differing network timesteps.

Ablation Study

Efficacy of Distillation Components. To validate the contribution of each component, we conduct an ablation study where we add SRD and SSD to a standard distillation baseline ("Logits KD"). As shown in Table 4, both modules individually improve performance. Notably, on the more complex CIFAR-100 dataset, adding only SRD provides a substantial accuracy gain of +3.06% over the baseline, highlighting the critical impact of correcting the feature-level mismatch. The full BSD framework's top accuracy demonstrates that its two components, SRD and SSD, are highly complementary. This confirms our view of spectral mismatch as a problem with two distinct aspects, requiring solutions at both the feature level and the logit level for optimal performance.

Location of Feature Distillation. We investigated how SRD module placement affects performance by applying it to different network blocks. The results in Table 5 reveal two patterns: applying SRD to a single, deeper block is more effective than a shallower one, and applying it to all blocks simultaneously yields the highest accuracy. This suggests that

Method	SEW-ResNet-19	
	CIFAR-10	CIFAR-100
w/o distillation	95.74	77.29
logits KD	96.33	79.46
+SSD	96.57	79.98
+SRD	96.53	82.52
BSD	**97.07**	**82.61**

Table 4: Ablation study on the components of BSD on CIFAR-10/100. All results are Top-1 accuracy (%). "Logits KD" is a standard distillation baseline. "+SRD" and "+SSD" denote adding each of our components to this baseline. "BSD" is the full framework.

Block Index			ResNet-19	SEW-ResNet-19
1	2	3	Top-1 Acc.(%)	Top-1 Acc.(%)
✓			95.41	96.70
	✓		95.28	96.53
		✓	95.87	96.99
✓	✓		95.32	96.71
✓		✓	95.74	96.77
	✓	✓	95.13	96.69
✓	✓	✓	**97.38**	**97.07**

Table 5: Impact of the SRD module's application point on Top-1 accuracy (%) on CIFAR-10. A checkmark (✓) indicates that SRD is applied at the corresponding network block.

spectral mismatch is a cumulative issue. While compensation at deeper layers—where the mismatch is greatest—is beneficial, our findings indicate that a comprehensive, multi-stage approach is the most effective strategy to counteract this effect.

Analysis of the Spectral Residual Adapter. To verify that SRD's performance gains stem from its frequency-domain design and not merely from added parameters, we conducted a control experiment. We created a spatial-only variant by removing the FFT/iFFT operations from the SRD module. On ImageNet with ResNet-18, this variant's accuracy dropped to 65.47%, notably lower than the 66.92% achieved by our standard SRD. A visualization of the learned weights in Figure 3 explains this performance gap. The standard SRD learns a structured high-pass filter, amplifying the high-frequency components that SNNs typically struggle to represent. In contrast, the weights of the spatial-only adapter show no apparent pattern. This comparison strongly indicates that SRD's success lies in its specialized design—which counteracts the SNN's low-pass nature—rather than in the parameter count alone.

Visualization of Learned Feature Salience. Finally, we use Grad-CAM to qualitatively evaluate how BSD influences the SNN's learned representations. As shown in Figure 4, SNNs trained with conventional distillation methods like KDSNN and LaSNN often produce diffuse attention maps that are less focused than the ANN teacher's. In contrast, the SNN trained with our BSD framework learns a much sharper and more semantically precise attention map that is visually very similar to the teacher's. This provides compelling visual evidence that BSD's benefits extend beyond simple accuracy improvements. By correcting the feature degradation caused by spectral mismatch, our method fundamentally enhances the SNN's ability to identify the most salient image regions, effectively learning where to look in a way that mirrors the teacher's semantic focus.

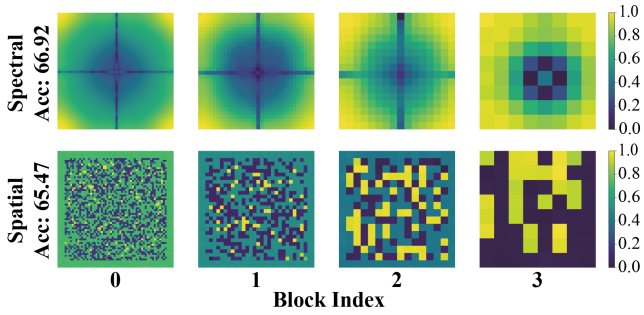

Figure 3: Visualization of the final learned weights (W_2) in the feature adapter module on ImageNet (SEW-ResNet18). **Top**: Our frequency-domain SRD learns a clear high-pass filter pattern. **Bottom**: For comparison, the module operating in the spatial domain learns no apparent pattern.

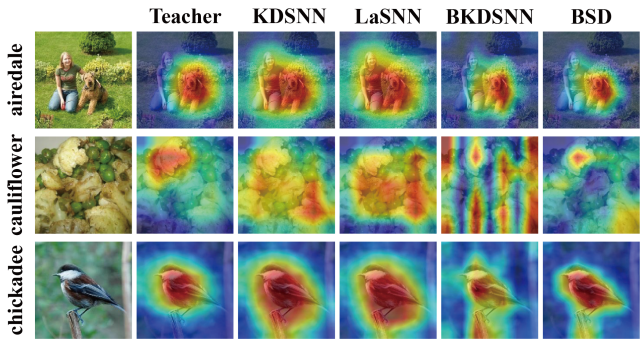

Figure 4: Grad-CAM visualizations comparing feature salience across different KD methods. The heatmaps (blue for low, red for high) indicate regions most influential for the model's classification decision.

Conclusion

In this work, we identify and address the spectral mismatch in ANN-to-SNN distillation, where an SNN's low-pass nature hinders high-frequency information transfer. Our Bi-Spectrum Distillation (BSD) framework tackles this with two complementary modules: SRD compensates for feature-level frequency loss, and SSD distills fine-grained structures from sorted logits. Extensive experiments confirm that BSD achieves state-of-the-art performance across diverse benchmarks and architectures. By reframing distillation as a frequency-domain problem, our work offers an effective solution that narrows the ANN-SNN performance gap and introduces a novel frequency perspective for SNN research.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 62325101, Grant 62031001 and Grant 62405014.

References

Bu, T.; Fang, W.; Ding, J.; Dai, P.; Yu, Z.; and Huang, T. 2023. Optimal ANN-SNN conversion for high-accuracy and ultra-low-latency spiking neural networks. *arXiv preprint arXiv:2303.04347*.

Davies, M.; Srinivasa, N.; Lin, T.-H.; Chinya, G.; Cao, Y.; Choday, S. H.; Dimou, G.; Joshi, P.; Imam, N.; Jain, S.; et al. 2018. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1): 82–99.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Deng, L.; Wang, G.; Li, G.; Li, S.; Liang, L.; Zhu, M.; Wu, Y.; Yang, Z.; Zou, Z.; Pei, J.; et al. 2020. Tianjic: A unified and scalable chip bridging spike-based and continuous neural computation. *IEEE Journal of Solid-State Circuits*, 55(8): 2228–2246.

Deng, S.; Li, Y.; Zhang, S.; and Gu, S. 2022. Temporal efficient training of spiking neural network via gradient reweighting. *arXiv preprint arXiv:2202.11946*.

Deng, S.; Lin, H.; Li, Y.; and Gu, S. 2023. Surrogate module learning: Reduce the gradient error accumulation in training spiking neural networks. In *International Conference on Machine Learning*, 7645–7657. PMLR.

Dong, Y.; Zhao, D.; and Zeng, Y. 2024. Temporal knowledge sharing enable spiking neural network learning from past and future. *IEEE Transactions on Artificial Intelligence*.

Eshraghian, J. K.; Ward, M.; Neftci, E. O.; Wang, X.; Lenz, G.; Dwivedi, G.; Bennamoun, M.; Jeong, D. S.; and Lu, W. D. 2023. Training spiking neural networks using lessons from deep learning. *Proceedings of the IEEE*, 111(9): 1016–1054.

Fang, W.; Chen, Y.; Ding, J.; Yu, Z.; Masquelier, T.; Chen, D.; Huang, L.; Zhou, H.; Li, G.; and Tian, Y. 2023. Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence. *Science Advances*, 9(40): eadi1480.

Fang, W.; Yu, Z.; Chen, Y.; Huang, T.; Masquelier, T.; and Tian, Y. 2021. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34: 21056–21069.

Fang, Y.; Wang, Z.; Zhang, L.; Cao, J.; Chen, H.; and Xu, R. 2024. Spiking wavelet transformer. In *European Conference on Computer Vision*, 19–37. Springer.

Fang, Y.; Zhou, D.; Wang, Z.; Ren, H.; Zeng, Z.; Li, L.; Zhou, S.; and Xu, R. 2025. Spiking Transformers Need High Frequency Information. *arXiv preprint arXiv:2505.18608*.

Gerstner, W.; Kistler, W. M.; Naud, R.; and Paninski, L. 2014. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press.

Guo, Y.; Chen, Y.; Liu, X.; Peng, W.; Zhang, Y.; Huang, X.; and Ma, Z. 2024a. Ternary spike: Learning ternary spikes for spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 12244–12252.

Guo, Y.; Peng, W.; Liu, X.; Chen, Y.; Zhang, Y.; Tong, X.; Jie, Z.; and Ma, Z. 2024b. Enof-snn: Training accurate spiking neural networks via enhancing the output feature. *Advances in Neural Information Processing Systems*, 37: 51708–51726.

Guo, Y.; Tong, X.; Chen, Y.; Zhang, L.; Liu, X.; Ma, Z.; and Huang, X. 2022a. 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 (CVPR)*, 326–335.

Guo, Y.; Zhang, L.; Chen, Y.; Tong, X.; Liu, X.; Wang, Y.; Huang, X.; and Ma, Z. 2022b. Real spike: Learning real-valued spikes for spiking neural networks. In *European Conference on Computer Vision*, 52–68. Springer.

Hao, Z.; Bu, T.; Ding, J.; Huang, T.; and Yu, Z. 2023. Reducing ann-snn conversion error through residual membrane potential. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 11–21.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hong, D.; Shen, J.; Qi, Y.; and Wang, Y. 2023. Lasnn: Layer-wise ann-to-snn distillation for effective and efficient training in deep spiking neural networks. *arXiv preprint arXiv:2304.09101*.

Hu, Y.; Tang, H.; and Pan, G. 2021. Spiking deep residual networks. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8): 5200–5205.

Hu, Y.; Zheng, Q.; Jiang, X.; and Pan, G. 2023. Fast-snn: Fast spiking neural network by converting quantized ann. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12): 14546–14562.

Huang, Z.; Shi, X.; Hao, Z.; Bu, T.; Ding, J.; Yu, Z.; and Huang, T. 2024. Towards high-performance spiking transformers from ann to snn conversion. In *Proceedings of the 32nd ACM international conference on multimedia*, 10688–10697.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*.

Li, H.; Liu, H.; Ji, X.; Li, G.; and Shi, L. 2017. Cifar10-dvs: an event-stream dataset for object classification. *Frontiers in neuroscience*, 11: 309.

Lindner, B. 2013. Low-pass filtering of information in the leaky integrate-and-fire neuron driven by white noise. In *International Conference on Theory and Application in Nonlinear Dynamics (ICAND 2012)*, 249–258. Springer.

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

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.

Qiu, H.; Ning, M.; Song, Z.; Fang, W.; Chen, Y.; Sun, T.; Ma, Z.; Yuan, L.; and Tian, Y. 2024. Self-architectural knowledge distillation for spiking neural networks. *Neural Networks*, 178: 106475.

Roy, K.; Jaiswal, A.; and Panda, P. 2019. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784): 607–617.

Trabelsi, C.; Bilaniuk, O.; Zhang, Y.; Serdyuk, D.; Subramanian, S.; Santos, J. F.; Mehri, S.; Rostamzadeh, N.; Bengio, Y.; and Pal, C. J. 2017. Deep complex networks. *arXiv preprint arXiv:1705.09792*.

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.

Xu, K.; Qin, M.; Sun, F.; Wang, Y.; Chen, Y.-K.; and Ren, F. 2020. Learning in the frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1740–1749.

Xu, M.; Ma, D.; Tang, H.; Zheng, Q.; and Pan, G. 2024a. FEEL-SNN: Robust Spiking Neural Networks with Frequency Encoding and Evolutionary Leak Factor. *Advances in Neural Information Processing Systems*, 37: 91930–91950.

Xu, Q.; Li, Y.; Shen, J.; Liu, J. K.; Tang, H.; and Pan, G. 2023. Constructing Deep Spiking Neural Networks From Artificial Neural Networks With Knowledge Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7886–7895.

Xu, Z.; You, K.; Guo, Q.; Wang, X.; and He, Z. 2024b. BKDSNN: Enhancing the Performance of Learning-Based Spiking Neural Networks Training with Blurred Knowledge Distillation. In *European Conference on Computer Vision*, 106–123. Springer.

Yang, S.; Yu, C.; Liu, L.; Ma, H.; Wang, A.; and Li, E. 2025. Efficient ANN-Guided Distillation: Aligning Rate-based Features of Spiking Neural Networks through Hybrid Block-wise Replacement. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 10025–10035.

Yao, M.; Hu, J.; Hu, T.; Xu, Y.; Zhou, Z.; Tian, Y.; Xu, B.; and Li, G. 2024. Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips. *arXiv preprint arXiv:2404.03663*.

Yao, M.; Qiu, X.; Hu, T.; Hu, J.; Chou, Y.; Tian, K.; Liao, J.; Leng, L.; Xu, B.; and Li, G. 2025. Scaling spike-driven transformer with efficient spike firing approximation training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yao, X.; Li, F.; Mo, Z.; and Cheng, J. 2022. Glif: A unified gated leaky integrate-and-fire neuron for spiking neural networks. *Advances in Neural Information Processing Systems*, 35: 32160–32171.

Yu, C.; Zhao, X.; Liu, L.; Yang, S.; Wang, G.; Li, E.; and Wang, A. 2025. Efficient distillation of deep spiking neural networks for full-range timestep deployment. *arXiv e-prints*, arXiv–2501.

Yu, W.; Si, C.; Zhou, P.; Luo, M.; Zhou, Y.; Feng, J.; Yan, S.; and Wang, X. 2023. Metaformer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2): 896–912.

Zhang, H.; and Zhang, Y. 2024. Memory-efficient reversible spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 16759–16767.

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*, volume 35, 11062–11070.

Zhou, C.; Yu, L.; Zhou, Z.; Ma, Z.; Zhang, H.; Zhou, H.; and Tian, Y. 2023a. Spikingformer: Spike-driven residual learning for transformer-based spiking neural network. *arXiv preprint arXiv:2304.11954*.

Zhou, C.; Zhang, H.; Zhou, Z.; Yu, L.; Huang, L.; Fan, X.; Yuan, L.; Ma, Z.; Zhou, H.; and Tian, Y. 2024. Qkformer: Hierarchical spiking transformer using qk attention. *Advances in Neural Information Processing Systems*, 37: 13074–13098.

Zhou, C.; Zhang, H.; Zhou, Z.; Yu, L.; Ma, Z.; Zhou, H.; Fan, X.; and Tian, Y. 2023b. Enhancing the performance of transformer-based spiking neural networks by SNN-optimized downsampling with precise gradient backpropagation. *arXiv preprint arXiv:2305.05954*.

Zhou, Z.; Zhu, Y.; He, C.; Wang, Y.; Yan, S.; Tian, Y.; and Yuan, L. 2022. Spikformer: When spiking neural network meets transformer. *arXiv preprint arXiv:2209.15425*.

Zhu, Y.; Ding, J.; Huang, T.; Xie, X.; and Yu, Z. 2024. Online stabilization of spiking neural networks. In *The Twelfth International Conference on Learning Representations*.