FGKDSNN: LEARNING FROM FIRING ACTIVITY VIA FIRING-GUIDED KNOWLEDGE DISTILLATION FOR SPIKING NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spiking neural networks (SNNs) provide an energy-efficient alternative to artificial neural networks (ANNs) through sparse spike-based computation, yet a substantial performance gap remains under ultra-low time-steps. Knowledge distillation (KD) from pretrained ANNs has emerged as an effective approach to improve SNN training. Nevertheless, existing SNN distillation methods uniformly transfer feature- and logits-level knowledge, without considering the firing activity of the student SNN. In this work, we reveal that *student firing activity is closely related to knowledge transfer at both feature- and logits-levels*: the firing activity is highly concentrated across feature channels, while higher sample-wise firing ratios are associated with larger differences between teacher and student non-target distributions. Based on the observations above, we propose firing-guided knowledge distillation for SNNs (*FGKDSNN*), consisting of firing-guided feature-based knowledge distillation (*FGFKD*) and firing-guided logits-based knowledge distillation (*FGLKD*). At the feature level, FGFKD leverages channel-wise firing activity to guide feature masking and restoration, strengthening teacher feature transfer to more active student channels. At the logits level, FGLKD reweights non-target distillation according to sample-wise firing activity. Extensive experiments across CNN- and Transformer-based SNNs demonstrate that *FGKDSNN* consistently outperforms existing SNN distillation methods (*e.g.*, +1.04% on ImageNet), achieving state-of-the-art performance on static and neuromorphic image classification, object detection, and semantic segmentation.

1 INTRODUCTION

Spiking neural networks (SNNs) have attracted increasing attention as an energy-efficient alternative to conventional artificial neural networks (ANNs). Unlike ANNs that communicate with continuous activations, SNNs transmit information through discrete spikes and perform sparse synaptic accumulation only when spikes are generated, making them well suited to neuromorphic hardware and event-based vision (Tavanaei et al., 2019; Fang et al., 2021). By combining back-propagation through time (BPTT) with surrogate gradients (Wu et al., 2018), learning-based SNNs have achieved substantial performance improvements. CNN-based architectures such as SEW-ResNet (Fang et al., 2021) enable residual learning in deep SNNs, while recent spiking Transformers further improve the accuracy and scalability of SNNs on large-scale vision tasks (Yao et al., 2025; Qiu et al., 2025; Fang et al., 2025). These developments have extended SNNs from image classification to neuromorphic vision, object detection, and *etc*. Nevertheless, *a clear performance gap remains between directly trained SNNs and their ANN counterparts, particularly under ultra-low time-steps*.

Knowledge distillation (KD) has emerged as an effective approach to bridge the performance gap between ANNs and SNNs by transferring knowledge from a well-trained ANN teacher to a student SNN. Existing SNN distillation methods have explored different forms of knowledge transfer. LaSNN transfers intermediate knowledge through layer-wise attention (Hong et al., 2023). BKDSNN employs masked feature restoration to alleviate the difficulty of directly matching discrete SNN features with continuous ANN features (Xu et al., 2024). TSER improves logit distillation by preserving supervision at individual time-steps (Yu et al., 2025). More recently, BSDSNN (Zhang et al., 2026) considers the spectral mismatch between ANNs and SNNs for both feature- and logit-based distillation. Despite

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

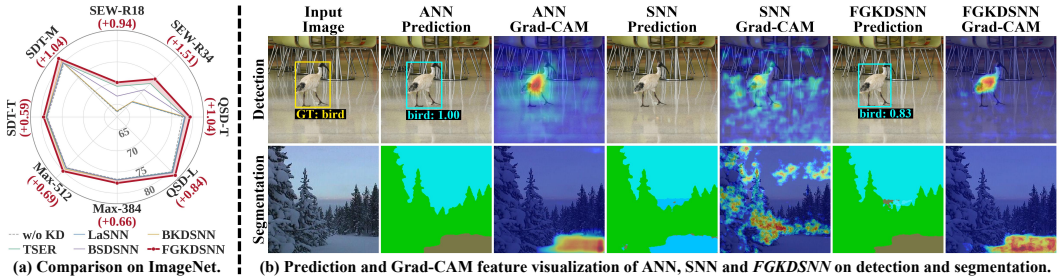


Figure 1: **(a) Comparison of *FGKDSNN* and previous SNN distillation methods on ImageNet** with SEW-R18/R34 (Fang et al., 2021), QSD-T/L (Qiu et al., 2025), Max-384/512 (Fang et al., 2025) and SDT-T/M (Yao et al., 2025). **(b) Prediction and Grad-CAM (Selvaraju et al., 2019) feature visualization of ANN, SNN and *FGKDSNN* on detection and segmentation** with QSD-T.

these advances, existing methods (Xu et al., 2023; Hong et al., 2023; Xu et al., 2024; Yu et al., 2025; Zhang et al., 2026) have not explicitly leveraged student firing activity as an adaptive signal for feature- or logit-level distillation across channels or samples. However, *firing activity is unevenly distributed across channels and varies considerably across samples.*

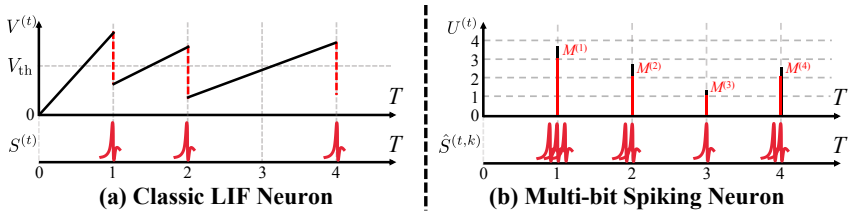
In this paper, we first reveal that *firing activity is closely related to knowledge transfer from ANN to SNN at both the feature and logits levels.* **At the feature level**, firing activity is unevenly distributed across channels, with a small subset accounting for most of the total activity (Figure 4(a)). **At the logits level**, the distance between teacher and student distributions over non-target classes increases with the sample-wise firing ratio (Figure 4(b)). Based on the observations above, we introduce two firing-guided distillation objectives: ① **Firing-Guided Feature-based Knowledge Distillation (*FGFKD*)** leverages channel-wise firing activity to guide masking and restoration, strengthening teacher feature transfer to more active student channels. ② **Firing-Guided Logits-based Knowledge Distillation (*FGLKD*)** exploits sample-wise firing activity to assign stronger non-target supervision to high-firing samples. Combining *FGFKD* with *FGLKD*, we establish **Firing-Guided Knowledge Distillation for SNNs (*FGKDSNN*)**, integrating firing activity into both feature- and logits-based knowledge distillation.

Extensive experiments on widely used and recent state-of-the-art CNN- and Transformer-based SNN architectures (Fang et al., 2021; Qiu et al., 2025; Fang et al., 2025; Yao et al., 2025) demonstrate consistent improvements over existing SNN knowledge distillation methods. As highlighted in Figure 1, *FGKDSNN* achieves state-of-the-art performance on both static (*e.g.*, CIFAR10, CIFAR100 and ImageNet) and neuromorphic (*e.g.*, CIFAR10-DVS) image classification benchmarks. Moreover, *FGKDSNN* further achieves state-of-the-art results on SNN-based object detection and semantic segmentation tasks. Our main contributions are summarized as follows:

- We reveal that firing activity of the student SNN is closely related to knowledge transfer at both feature and logits levels, motivating firing-guided distillation.
- We propose *FGKDSNN*, which incorporates firing activity into both feature- and logits-based distillation. *FGFKD* assigns higher masking probabilities to more active channels to enhance teacher feature transfer, while *FGLKD* reweights non-target distillation by sample-wise firing activity to strengthen supervision on high-firing samples.
- Extensive experiments on both CNN- and Transformer-based SNNs demonstrate the effectiveness and generality of *FGKDSNN*. Our method achieves state-of-the-art performance on static and neuromorphic image classification benchmarks, and further establishes state-of-the-art results on SNN-based object detection and semantic segmentation.

2 Background

Spiking Neural Networks. ANN-to-SNN conversion and learning-based training are two major approaches for obtaining SNNs. Conversion-based methods transfer pre-trained ANNs to their SNN counterparts by replacing continuous activations with spiking neurons (Bu et al., 2023; You et al., 2024; Huang et al., 2024). Although these methods can achieve competitive performance, they require relatively large time-steps (*e.g.*, 32 in SpikeZIP-TF (You et al., 2024)) to approximate ANN activations. In contrast, learning-based methods optimize SNNs via back-propagation through



(a) Classic LIF Neuron **(b) Multi-bit Spiking Neuron**

Figure 2: **Illustration of two kinds of neuron firing activities.**

time (BPTT) with surrogate gradients, enabling efficient inference under ultra-low time-steps (Wu et al., 2018; Neftci et al., 2019). Based on BPTT, SEW-ResNet (Fang et al., 2021) introduces spike-element-wise residual connections and enables effective optimization of deep CNN-based SNNs. More recently, Spiking Transformers have substantially improved the scalability and performance of learning-based SNNs. Spikformer (Zhou et al., 2022) with a novel spiking self-attention block achieves a significant improvement in image classification. Spike-driven Transformer (Yao et al., 2023b) introduces spike-based self-attention for efficient Transformer architectures, while subsequent works (Yao et al., 2024; 2025) further improve their performance through architecture design and training strategies. QSDFormer (Qiu et al., 2025) develops a compact quantized spike-driven Transformer, and MaxFormer (Fang et al., 2025) improves SNN representation by preserving high-frequency information. These developments establish learning-based SNNs as effective backbones for image classification, object detection, and semantic segmentation.

Knowledge Distillation for SNNs. Despite the progress of learning-based SNNs, *an accuracy gap generally remains between SNNs and their ANN counterparts*. Knowledge distillation (KD) transfers knowledge from a pre-trained teacher ANN to a student SNN and has become an effective approach to improve learning-based SNNs. Existing knowledge distillation methods on SNN exploit logits or intermediate features from the teacher ANN. KDSNN (Xu et al., 2023) jointly transfers logit- and feature-level knowledge, while LaSNN (Hong et al., 2023) introduces layer-wise attention-based distillation to align intermediate representations between ANNs and SNNs. BKDSNN (Xu et al., 2024) further applies random feature masking and restoration ahead of the last layer, enabling the student SNN to restore and mimic the corresponding ANN feature. At the logit level, TSER (Yu et al., 2025) performs distillation at individual time-steps and introduces entropy regularization to improve temporal knowledge transfer. More recently, BSDSNN (Zhang et al., 2026) leverages spectral residual distillation for feature transfer and spectral semantic distillation for logit transfer to address the spectral mismatch between ANN and SNN, achieving promising performance. However, *existing SNN distillation methods neglect the firing activity of student SNN*, motivating us to exploit firing activity for both logits- and feature-based distillation.

3 PRELIMINARIES

Learning-based spiking neural networks (SNNs) employ multiple spiking neurons to represent firing activity. The student SNNs in this paper mainly involve two forms: the binary leaky integrate-and-fire (LIF) neuron applied in SEW-ResNet (Fang et al., 2021) and MaxFormer (Fang et al., 2025), and multi-bit spiking neuron adopted by QSDFormer (Qiu et al., 2025) and SDT-v3 (Yao et al., 2025). In this part, we briefly introduce the two spiking neurons as follows.

Classic LIF Neuron. The leaky integrate-and-fire (LIF) neuron (Maass, 1997) is one of the most widely used computational units in SNNs. As depicted in Figure 2(a), under each discrete time-step t, the neuron first integrates the charge carried by the input spikes into the membrane potential, then emits a binary spike once the firing threshold is exceeded, and finally resets the membrane state:

$$H^{(t)} = \beta V^{(t-1)} + (1-\beta) I^{(t)}, \; S^{(t)} = \Theta\big(H^{(t)} - V_{\text{th}}\big), \; V^{(t)} = H^{(t)} - S^{(t)} V_{\text{th}}. \tag{1}$$

where $I^{(t)}$ denotes the input current at time-step t, $H^{(t)}$ is the pre-reset membrane potential, $V^{(t)}$ is the post-reset membrane potential passed to the next time-step, V_{th} is the firing threshold, $S^{(t)}$ denotes the firing spike at time-step t, β is the decay factor, and $\Theta(\cdot)$ denotes the Heaviside step function. Since each neuron produces a binary response at each time-step, its firing activity can be naturally characterized by the occurrence of spikes over time. Note that SEW-ResNet (Fang et al., 2021) and MaxFormer (Fang et al., 2025) adopt LIF neuron.

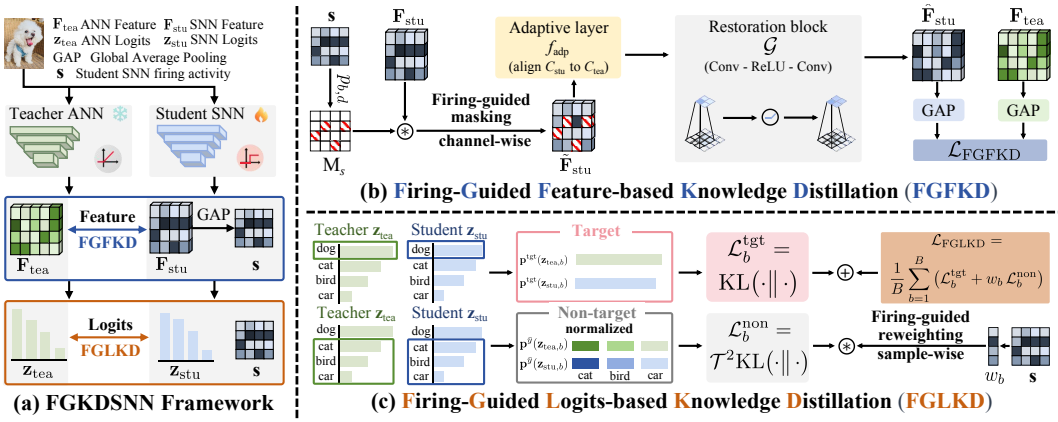


Figure 3: **Overview of the Firing-Guided Knowledge Distillation (*FGKDSNN*) framework. (a)** *FGKDSNN* consists of two complementary modules: *FGFKD* for feature-based distillation and *FGLKD* for logits-based distillation, both guided by the student firing activity $\mathbf{s}$. **(b)** *FGFKD* (Section 4.3) applies channel-wise firing-guided masking to focus feature restoration on active student channels, facilitating more effective teacher feature transfer. **(c)** *FGLKD* (Section 4.4) reweights non-target distillation by sample-wise firing activity, strengthening supervision on high-firing samples.

Multi-bit Spiking Neuron. To enhance spiking representation capability, previous SNNs (Ponghiran & Roy, 2022; Shen et al., 2024) employ multi-bit spiking neurons with integer-valued outputs at each time step. As depicted in Figure 2(b), instead of restricting the neuronal response to $\{0, 1\}$, a multi-bit spiking neuron represents the membrane potential with multiple discrete firing levels. The dynamics of multi-bit spiking neuron are defined as follows:

$$M^{(t)} = \left\lfloor \mathrm{clip}\left(U^{(t)}, 0, D\right) \right\rceil, \quad A^{(t)} = M^{(t)}/D = 1/D \cdot \textstyle\sum_{k=1}^{D} \hat{S}^{(t,k)} \tag{2}$$

where $U^{(t)}$ denotes the membrane potential, D is the maximum firing level, and $M^{(t)}$ represents the integer-valued firing response. $\hat{S}^{(t,k)} \in \{0, 1\}$ denotes the k-th binary spike used to equivalently decompose the multi-bit response at time-step t, such that $\sum_{k=1}^{D} \hat{S}^{(t,k)} = M^{(t)}$. The normalized output $A^{(t)} \in [0, 1]$ can thus be interpreted as the firing rate over D binary spikes. Compared with binary firing, the multi-bit response provides *a finer approximation of neuronal activity while remaining discretized and event-driven.* QSDFormer (Qiu et al., 2025) introduces the information-enhanced LIF (IE-LIF), leveraging multi-bit outputs during training and equivalent binary spikes during inference. Similarly, SDT-v3 (Yao et al., 2025) proposes spike firing approximation (SFA), which uses integer-valued activations during training and equivalent binary spike sequences during inference. In both cases, the normalized multi-level responses in $[0, 1]$ provide a unified characterization of firing activity across different student SNN architectures.

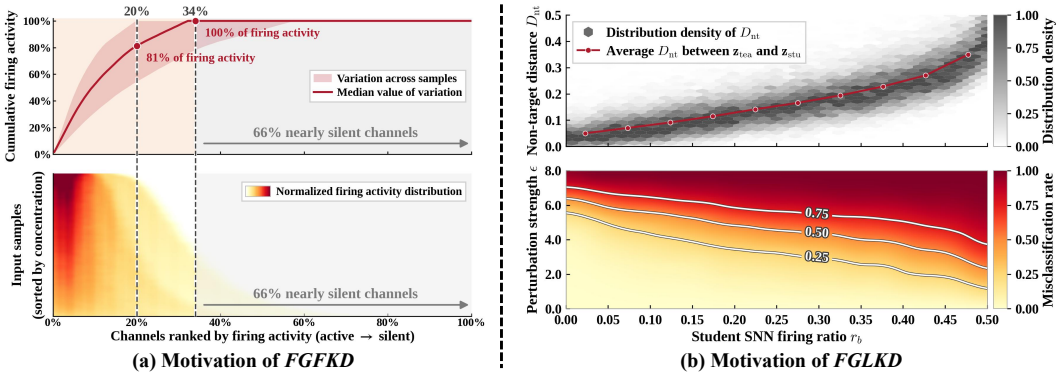
4 Methodology

4.1 Overview

As illustrated in Figure 3, we propose firing-guided knowledge distillation for SNNs (FGKD-SNN), which *leverages student firing activity to guide knowledge transfer from the teacher ANN*. The framework consists of two complementary objectives operating at both the feature and logits levels. At the feature level, FGFKD exploits channel-wise firing activity to guide feature masking and restoration, thereby *strengthening teacher feature transfer to more active student channels*. At the logits level, FGLKD leverages sample-wise firing activity to *adapt the strength of non-target distillation across samples*. We first

Table 1: Notations used in *FGKDSNN*.

Symbol	Definition
B, C, T	batch size, classes, total time-steps
$C_{\mathrm{stu}}, C_{\mathrm{tea}}$	student/teacher feature channels
$\mathbf{F}_{\mathrm{tea}}$	teacher intermediate feature
$\mathbf{F}_{\mathrm{stu}}^{(\tau)}$	student feature at time-step τ
$\mathbf{F}_{\mathrm{stu}}$	temporally averaged student feature
$\mathbf{z}_{\mathrm{tea}}, \mathbf{z}_{\mathrm{stu}}$	teacher and student logits
$\mathbf{s}$	student firing activity
y_b	ground-truth label of sample b
$\mathcal{T}$	distillation temperature
$\sigma(\cdot)$	softmax operation



(a) Motivation of ***FGFKD*** (b) Motivation of ***FGLKD***

Figure 4: **(a) Motivation of *FGFKD*.** *Top:* Cumulative firing activity over channels ranked from active to silent, where the top 20% of channels account for 81% of activity and the top 34% capture nearly all activity. *Bottom:* Normalized firing activity distributions across samples, sorted by channel-wise activity concentration, showing that the remaining 66% of channels are nearly silent. These observations motivate firing-guided feature masking. **(b) Motivation of *FGLKD*.** *Top:* The non-target distance D_{nt} between the ANN and SNN non-target distributions increases with the student firing ratio r_b. *Bottom:* Samples with larger r_b require smaller perturbations ϵ to reach the same misclassification rate. These results motivate stronger non-target distillation for high-firing samples.

define the student firing activity shared by both objectives, and then sequentially introduce *FGFKD* and *FGLKD*, followed by the overall training objective. **The notations are summarized in Table 1.**.

Student firing activity. The teacher ANN produces an intermediate feature $\mathbf{F}_{\mathrm{tea}} \in \mathbb{R}^{B\times C_{\mathrm{tea}}\times H\times W}$ and logits $\mathbf{z}_{\mathrm{tea}} \in \mathbb{R}^{B\times C}$. At each simulation step τ, the student SNN produces a binary spike feature $\mathbf{F}_{\mathrm{stu}}^{(\tau)} \in \{0,1\}^{B\times C_{\mathrm{stu}}\times H\times W}$. We define the temporally averaged student feature $\mathbf{F}_{\mathrm{stu}}$ as follows:

$$\mathbf{F}_{\mathrm{stu}} = \tfrac{1}{T}\textstyle\sum_{\tau=1}^{T} \mathbf{F}_{\mathrm{stu}}^{(\tau)} \in [0,1]^{B\times C_{\mathrm{stu}}\times H\times W} \tag{3}$$

The corresponding student logits $\mathbf{z}_{\mathrm{stu}} \in \mathbb{R}^{B\times C}$ are generated by global average pooling (GAP) followed by the classifier. As shown in Figure 3(a), the temporally averaged feature $\mathbf{F}_{\mathrm{stu}}$ is further pooled over the spatial dimensions $H \times W$ to obtain the channel-wise student firing activity $\mathbf{s} \in [0,1]^{B\times C_{\mathrm{stu}}}$:

$$\mathbf{s} = \mathrm{GAP}(\mathbf{F}_{\mathrm{stu}}) \in [0,1]^{B\times C_{\mathrm{stu}}}, \qquad s_{b,d} \in [0,1], \tag{4}$$

where $s_{b,d}$ denotes the firing ratio of channel d for sample b, averaged over the temporal and spatial dimensions. The firing activity $\mathbf{s}$ is extracted directly from the intermediate student feature and is exploited by *FGFKD* and *FGLKD*.

4.2 Motivation of *FGFKD* and *FGLKD*

In this part, we clarify the motivation of *FGFKD* and *FGLKD*. Note that the visualizations in Figure 4 are obtained on the ImageNet validation set, using the pretrained QSD-T (Qiu et al., 2025) as the student SNN and Vision Transformer Small (Dosovitskiy et al., 2020) as the teacher ANN.

Motivation of *FGFKD*. Masked feature reconstruction learns feature-level knowledge by recovering teacher representations from partially observed student features (Xu et al., 2024). However, *SNN representations are inherently characterized by sparse firing activity* (Tavanaei et al., 2019; Fang et al., 2021; Yao et al., 2023a), while our analysis further reveals that *such activity is highly concentrated across channels*. In Figure 4(a), the most active 20% of channels account for approximately 81% of the total firing activity, while activity nearly saturates within the top 34%, leaving about 66% of channels nearly silent. Prior uniform random masking (Xu et al., 2024) is agnostic to this activity distribution and assigns identical masking probabilities to highly active and nearly silent channels. *This observation motivates conditioning the masking probability on channel-wise firing activity*.

Motivation of *FGLKD*. Logits-based knowledge distillation transfers both target-class information and the relative distribution over non-target classes (Zhao et al., 2022), with the latter encoding inter-class relations beyond the ground-truth category (Yang et al., 2023). For the student SNN, such non-target information may vary with sample-level firing activity. We first define the sample-level

firing ratio $r_b = 1/C_{\text{stu}} \cdot \sum_{d=1}^{C_{\text{stu}}} s_{b,d}$ by averaging the channel-wise firing activity. Following the non-target class normalization in DKD (Zhao et al., 2022), we remove the ground-truth class y_b and normalize the remaining logits as $\mathbf{p}^{\bar{y}}(\mathbf{z}) = \sigma(\mathbf{z}_{\bar{y}}/\mathcal{T})$, then leverage the total variation distance (Wen et al., 2023) to measure the distance between the non-target distributions of the ANN and SNN:

$$D_{\text{nt}} = \tfrac{1}{2} \left\| \mathbf{p}^{\bar{y}}(\mathbf{z}_{\text{tea}}) - \mathbf{p}^{\bar{y}}(\mathbf{z}_{\text{stu}}) \right\|_1 . \tag{5}$$

Accordingly, D_{nt} quantifies the distance between ANN and SNN over non-target classes. As shown in Figure 4(b), D_{nt} increases consistently with the sample-level firing ratio r_b. To further assess the effect of non-target distributional distance on classification, we identify the maximum non-target class $c_b^\star = \arg\max_{c \neq y_b} z_{\text{stu},b,c}$ and perturb its student logit by an additive magnitude ϵ: $\tilde{z}_{\text{stu},b,c_b^\star} = z_{\text{stu},b,c_b^\star} + \epsilon$, while keeping the remaining logits unchanged. The perturbation magnitude ϵ reflects the logit-space margin required to change the prediction, with a smaller value indicating greater classification sensitivity (Elsayed et al., 2018). As shown in the bottom panel of Figure 4(b), smaller ϵ is required to attain the same misclassification rate (*e.g.*, 0.25, 0.50, or 0.75) as r_b increases.

4.3 Firing-Guided Feature-based Knowledge Distillation

Channel-wise firing-guided masking. The feature-level firing pattern in Figure 4(a) indicates that uniform masking (Xu et al., 2024) is suboptimal for sparse SNN features, as many masked channels contain limited firing activity. In Figure 3(b), to enhance the feature transfer from ANN to SNN on high-firing channels, *FGFKD* determines the masking probability according to channel-wise firing activity. For channel d of sample b, the masking probability is defined as follows:

$$p_{b,d} = \text{clip}(s_{b,d}, p_{\min}, p_{\max}) . \tag{6}$$

where $p_{\min}$ and $p_{\max}$ bound the masking probability, and the binary mask $M_{b,d}$ is sampled from:

$$M_{b,d} \sim \text{Bernoulli}(1 - p_{b,d}) . \tag{7}$$

where $M_{b,d} = 1$ preserves the channel and $M_{b,d} = 0$ masks it. The sampled mask is broadcast over the spatial dimensions and applied to the student feature:

$$\tilde{\mathbf{F}}_{\text{stu}} = \mathbf{M}_s \odot \mathbf{F}_{\text{stu}}. \tag{8}$$

Thus, *channels with higher firing activity are assigned higher masking probabilities*.

Alignment and restoration. As depicted in Figure 3(b), we first apply a 1×1 adaptive convolution layer f_{adp} to align the student channel C_{stu} of $\tilde{\mathbf{F}}_{\text{stu}}$ to the teacher channel C_{tea} of $\mathbf{F}_{\text{tea}}$. We then introduce a restoration block $\mathcal{G}$ with two convolutional layers connected by a ReLU layer and apply it to obtain the masked restoration feature $\hat{\mathbf{F}}_{\text{stu}}$ from $\tilde{\mathbf{F}}_{\text{stu}}$ as follows:

$$\hat{\mathbf{F}}_{\text{stu}} = \mathcal{G}\Big(f_{\text{adp}}\Big(\tilde{\mathbf{F}}_{\text{stu}} \Big) \Big) . \tag{9}$$

With the masked restoration feature $\hat{\mathbf{F}}_{\text{stu}}$ and teacher feature $\mathbf{F}_{\text{tea}}$, the *FGFKD* loss $\mathcal{L}_{\text{FGFKD}}$ is:

$$\mathcal{L}_{\text{FGFKD}} = \tfrac{1}{B} \sum_{b=1}^{B} \left\| \text{GAP}\Big(\hat{\mathbf{F}}_{\text{stu},b} \Big) - \text{GAP}(\mathbf{F}_{\text{tea},b}) \right\|_2^2 . \tag{10}$$

4.4 Firing-Guided Logits-based Knowledge Distillation

Target and non-target distillation. Following the target and non-target decomposition of logits-based KD (Zhao et al., 2022; Yang et al., 2023), we first separate the target and non-target knowledge distillation for SNN. For sample b with logits $\mathbf{z} \in \mathbb{R}^C$, the target distribution is defined as follows:

$$\mathbf{p}^{\text{tgt}}(\mathbf{z}) = [\sigma(\mathbf{z})_{y_b},\, 1 - \sigma(\mathbf{z})_{y_b}] . \tag{11}$$

Equation (11) yields the target-class knowledge distillation loss as follows:

$$\mathcal{L}_b^{\text{tgt}} = \text{KL}\big(\mathbf{p}^{\text{tgt}}(\mathbf{z}_{\text{tea},b}) \,\big\|\, \mathbf{p}^{\text{tgt}}(\mathbf{z}_{\text{stu},b}) \big) . \tag{12}$$

As depicted in Figure 3, for non-target classes, the ground-truth class is removed and the remaining logits are normalized with temperature $\mathcal{T}$:

$$p_c^{\bar{y}}(\mathbf{z}) = \tfrac{\exp(z_c/\mathcal{T})}{\sum_{c' \neq y_b} \exp(z_{c'}/\mathcal{T})}, \quad c \neq y_b. \tag{13}$$

Then the corresponding non-target knowledge distillation loss is defined as follows:

$$\mathcal{L}_b^{\text{non}} = \mathcal{T}^2 \, \text{KL}\big(\mathbf{p}^{\bar{y}}(\mathbf{z}_{\text{tea},b}) \,\big\|\, \mathbf{p}^{\bar{y}}(\mathbf{z}_{\text{stu},b}) \big) . \tag{14}$$

Table 2: **Top-1 accuracy (%) comparison with KD methods on ImageNet.** † indicates the reproduced results. [·] denotes the parameter size. Best results are in **bold** and shaded in green, while the runner-up results are shaded in gray.

Method \ Model	SEW			QSDFormer			MaxFormer			SDT-v3	
	T	R-18 [11.69M]	R-34 [21.79M]	T	QSD-T [1.80M]	QSD-L [6.80M]	T	Max-384 [16.23M]	Max-512 [28.65M]	T	SDT-T [10.00M]	SDT-M [19.00M]
w/o KD	4	63.18	67.04	4	77.50	80.30	4	76.72†	78.92†	4	78.50	79.80
Teacher ANN	–	71.69	75.38	–	81.23	83.78	–	81.23	83.78	–	81.23	83.78
LaSNN†	4	63.33	66.98	4	77.97	80.41	4	76.84	79.07	4	78.74	80.02
BKDSNN†	4	63.43	67.21	4	78.24	80.59	4	76.96	79.40	4	78.82	80.52
TSER†	4	69.24	73.16	4	78.15	80.52	4	76.89	79.11	4	78.79	80.08
BSDSNN†	4	66.92	71.08	4	78.19	80.65	4	76.92	78.99	4	78.63	80.19
FGKDSNN	4	**70.18 (+0.94)**	**74.67 (+1.51)**	4	**79.28 (+1.04)**	**81.49 (+0.84)**	4	**77.62 (+0.66)**	**80.09 (+0.69)**	4	**79.41 (+0.59)**	**81.56 (+1.04)**

Sample-wise firing-guided reweighting. As depicted in Figure 4(b), the increased non-target mismatch and perturbation sensitivity at higher r_b further motivate firing-guided reweighting of $\mathcal{L}_b^{\text{non}}$. Accordingly, we derive the sample-wise weight w_b from sample-wise firing ratio r_b as follows:

$$w_b = \text{clip}(r_b, w_{\min}, w_{\max}) \,. \tag{15}$$

With the sample-wise weight w_b above, the *FGLKD* loss $\mathcal{L}_{\text{FGLKD}}$ is defined as follows:

$$\mathcal{L}_{\text{FGLKD}} = \tfrac{1}{B} \textstyle\sum_{b=1}^{B} \left(\mathcal{L}_b^{\text{tgt}} + w_b \, \mathcal{L}_b^{\text{non}} \right) . \tag{16}$$

Consequently, *the non-target term is strengthened with increasing firing activity, whereas the target component remains uniformly distilled across samples*.

4.5 Training Objective

Finally, we integrate the proposed $\mathcal{L}_{\text{FGFKD}}$ and $\mathcal{L}_{\text{FGLKD}}$ with the task loss $\mathcal{L}_{\text{task}}$ to obtain the overall training objective $\mathcal{L}_{\text{total}}$ as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \alpha \, \mathcal{L}_{\text{FGFKD}} + \beta \, \mathcal{L}_{\text{FGLKD}}. \tag{17}$$

where α and β balance the feature- and logits-based distillation losses, respectively.

5 Experiments

5.1 Experimental Setups

Student SNN Models and Distillation Baselines. We evaluate *FGKDSNN* on both CNN- and Transformer-based SNNs. For CNN-based models, we adopt SEW-ResNet (Fang et al., 2021). For Transformer-based models, we consider QSDFormer (Qiu et al., 2025), MaxFormer (Fang et al., 2025) and SDT-v3 (Yao et al., 2025). We then compare *FGKDSNN* with representative SNN distillation methods, including LaSNN (Hong et al., 2023), BKDSNN (Xu et al., 2024), TSER (Yu et al., 2025) and BSDSNN (Zhang et al., 2026).

Image Classification. We evaluate image classification tasks on three static datasets (*i.e.*, CIFAR10 (Krizhevsky et al., 2009), CIFAR100 (Krizhevsky et al., 2009) and ImageNet (Deng et al., 2009)) and one neuromorphic dataset (CIFAR10-DVS (Hongmin et al., 2017)). For CIFAR10 and CIFAR100, ResNet-34 (He et al., 2016) and Vision Transformer Small (ViT-Small) (Dosovitskiy et al., 2020) are adopted as the teacher ANNs for CNN- and Transformer-based student SNNs, respectively. For ImageNet, we leverage pretrained ResNet teachers for SEW-ResNet and pretrained ViT teachers for Transformer-based SNNs. Following BKDSNN (Xu et al., 2024), $\mathcal{L}_{\text{FGFKD}}$ is applied *only to the intermediate feature ahead of the final classification layer*. For CIFAR10-DVS, we follow the high-to-low time-step distillation setting of BKDSNN, where SNNs with $T = 16$ serve as teachers for the corresponding student SNNs with lower time-steps (*i.e.*, $T = 8$).

Object Detection. We evaluate object detection on COCO (Lin et al., 2015) using Mask R-CNN (He et al., 2017), with SDT-v3 (Yao et al., 2025) and QSDFormer (Qiu et al., 2025) as student SNN backbones and ConvFormer-S16 (Yu et al., 2023) as the teacher ANN. For *FGKDSNN*, *FGFKD* is applied to the five multi-scale features produced by the feature pyramid network (FPN) (Lin et al., 2017). *FGLKD* is performed on the region-of-interest (RoI) classification logits using shared RoIs for the teacher and student, involving only positive RoIs and the 80 foreground classes.

Table 3: **Top-1 accuracy (%) comparison of KD methods on CIFAR10, CIFAR100, and CIFAR10-DVS.** CF, DVS and $[\cdot]$ denote CIFAR, CIFAR10-DVS and parameter size, respectively.

Method \ Model	SEW-R19						MaxFormer					
	CF10 [12.63M]		CF100 [12.66M]		DVS [12.63M]		CF10 [6.57M]		CF100 [6.60M]		DVS [1.45M]	
	T	Acc.	T	Acc.	T	Acc.	T	Acc.	T	Acc.	T	Acc.
w/o KD	4	93.24	4	70.84	8	78.9	4	97.04	4	82.65	8	81.1
Teacher	–	98.21	–	86.19	16	82.0	–	98.76	–	88.14	16	84.2
BKDSNN†	4	97.14	4	81.40	8	80.2	4	97.26	4	84.43	8	82.2
TSER†	4	96.72	4	81.28	8	79.6	4	96.97	4	83.14	8	81.8
BSDSNN†	4	97.07	4	82.61	8	80.9	4	97.21	4	83.78	8	81.9
FGKDSNN	4	**97.42 (+0.28)**	4	**83.26 (+0.65)**	8	**81.4 (+0.5)**	4	**97.98 (+0.72)**	4	**86.56 (+2.13)**	8	**83.2 (+1.0)**

Table 4: **Object detection results on COCO 2017.** † indicates the reproduced results. We report mAP@50 (%).

Method \ Model	SDT-v3			QSDFormer		
	T	SDT-T [21.2M]	SDT-M [38.7M]	T	QSD-T [16.9M]	QSD-L [34.9M]
w/o KD	4	46.5†	49.4†	4	48.1	54.4†
Teacher ANN	–	67.7	67.7	–	67.7	67.7
BKDSNN†	4	46.9	49.8	4	49.1	55.3
TSER†	4	46.6	49.7	4	48.3	54.7
BSDSNN†	4	47.1	49.7	4	49.0	55.1
FGKDSNN	4	**47.9 (+0.8)**	**50.7 (+0.9)**	4	**50.5 (+1.4)**	**56.2 (+0.9)**

Table 5: **Semantic segmentation results on ADE20K.** † indicates the reproduced results. We report mIoU (%).

Method \ Model	SDT-v3			QSDFormer		
	T	SDT-T [11.5M]	SDT-M [20.4M]	T	QSD-T [3.3M]	QSD-L [9.6M]
w/o KD	4	39.9†	41.3†	4	38.5	40.5†
Teacher ANN	–	48.6	48.6	–	48.6	48.6
BKDSNN†	4	40.5	41.6	4	38.9	41.2
TSER†	4	40.1	41.5	4	38.8	40.9
BSDSNN†	4	40.4	41.8	4	39.1	41.6
FGKDSNN	4	**41.8 (+1.3)**	**42.5 (+0.7)**	4	**40.1 (+1.0)**	**43.1 (+1.5)**

Semantic Segmentation. We evaluate semantic segmentation on ADE20K (Zhou et al., 2017) with 512×512 training crops, using SDT-v3 (Yao et al., 2025) and QSDFormer (Qiu et al., 2025) as student SNN backbones and ConvFormer-S16 (Yu et al., 2023) as the teacher ANN. *FGFKD* is applied to the four multi-scale pyramid features ahead of the segmentation head, while *FGLKD* is performed pixel-wise on the segmentation logits at the native spatial resolution.

Implementation Details. For all tasks, we set $p_{\min}$=0.25 and $p_{\max}$=0.75 for $\mathcal{L}_{\mathrm{FGFKD}}$, $w_{\min}$=0.25, $w_{\max}$=1.0, and $\mathcal{T}$=1 for $\mathcal{L}_{\mathrm{FGLKD}}$. For image classification, α=2×10^{-3} for $\mathcal{L}_{\mathrm{FGFKD}}$ and β=0.5 for $\mathcal{L}_{\mathrm{FGLKD}}$, while for object detection and semantic segmentation, α=2×10^{-1} for $\mathcal{L}_{\mathrm{FGFKD}}$ and β=0.1 for $\mathcal{L}_{\mathrm{FGLKD}}$. The detailed training setups are specified in Section A1.

5.2 Benchmark Performance

Results on ImageNet. As tabulated in Table 2, *FGKDSNN* consistently achieves the best performance on ImageNet across CNN- and Transformer-based SNNs. Specifically, compared with the previous KD methods, *FGKDSNN* improves Top-1 accuracy by 0.94%/1.51% on SEW-R18/R34, 1.04%/0.84% on QSD-T/L, 0.66%/0.69% on Max-384/512, and 0.59%/1.04% on SDT-T/M, respectively. The consistent improvements across diverse SNN architectures on large-scale ImageNet highlight the effectiveness of adapting knowledge transfer according to student firing activity.

Results on CIFAR10 and CIFAR100. The results in Table 3 further demonstrate the effectiveness of *FGKDSNN* on both CNN- and Transformer-based SNNs. On CIFAR10 and CIFAR100, *FGKDSNN* surpasses the best prior KD results by 0.28%/0.65% on SEW-R19 and 0.72%/2.13% on MaxFormer, respectively. The consistent gains across different datasets and architectures further demonstrate the effectiveness and generality of firing-guided knowledge transfer.

Results on CIFAR10-DVS. On CIFAR10-DVS dataset, *FGKDSNN* achieves state-of-the-art performance under the high-to-low time-step distillation setting. Specifically, *FGKDSNN* achieves 81.4% with SEW-R19 and 83.2% with MaxFormer at $T = 8$, exceeding the previous SOTA results by 0.5% and 1.0%, respectively. The results extend the effectiveness of *FGKDSNN* from static inputs to event-driven inputs, demonstrating its applicability to temporally encoded SNN representations.

Results on Object Detection. As tabulated in Table 4, *FGKDSNN* consistently improves SDT-v3 and QSDFormer on COCO 2017. Notably, *FGKDSNN* with QSD-L obtains 56.2% mAP@50, achieving the best performance among the compared SNN distillation methods. These results demonstrate the effectiveness of *FGKDSNN* beyond image classification.

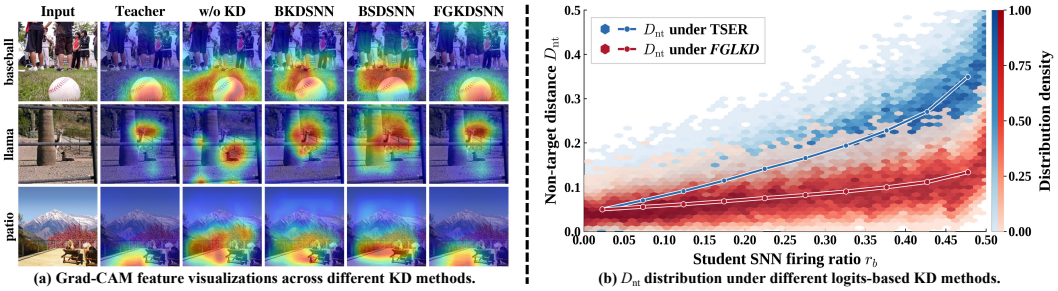
Results on Semantic Segmentation. As reported in Table 5, *FGKDSNN* consistently improves the performance of SDT-v3 and QSDFormer on ADE20K. Specifically, *FGKDSNN* with QSD-L

Table 6: Ablation of firing guidance in $\mathcal{L}_{\text{FGFKD}}$ and $\mathcal{L}_{\text{FGLKD}}$ on CF10 and CF100.

Loss	Firing-Guided	SEW-R19 CF10	SEW-R19 CF100	MaxFormer CF10	MaxFormer CF100
$\mathcal{L}_{\text{FGFKD}}$	✗ (Uniform)	96.18	81.43	97.02	84.19
	✗ (Shuffled)	96.04	81.52	96.96	84.13
	✓	**96.94**	**82.02**	**97.59**	**85.17**
$\mathcal{L}_{\text{FGLKD}}$	✗ (Fixed)	95.88	80.23	96.93	83.98
	✗ (Shuffled)	95.94	80.36	96.85	84.04
	✓	**96.43**	**81.09**	**97.43**	**84.96**

Table 7: Component ablation on CIFAR10 (CF10) and CIFAR100 (CF100).

$\mathcal{L}_{\text{FGFKD}}$	$\mathcal{L}_{\text{FGLKD}}$	SEW-R19 CF10	SEW-R19 CF100	MaxFormer CF10	MaxFormer CF100
✗	✗	93.24	70.84	97.04	82.65
✓	✗	96.94	82.02	97.59	85.17
✗	✓	96.43	81.09	97.43	84.96
✓	✓	**97.42**	**83.26**	**97.98**	**86.56**



(a) Grad-CAM feature visualizations across different KD methods.

(b) D_{nt} distribution under different logits-based KD methods.

Figure 5: **(a) Grad-CAM feature visualizations across different KD methods. (b) D_{nt} distribution under different logits-based KD methods**, including TSER (Yu et al., 2025) and our *FGLKD*.

achieves the best result among the compared SNN models. These experimental results further verify the generality of *FGKDSNN* across complex downstream tasks.

5.3 Ablation Study

Ablation on channel-wise firing-guided masking in *FGFKD*. As tabulated in Table 6, *Uniform* follows the random channel-wise masking of BKDSNN (Xu et al., 2024), while *Shuffled* permutes the firing-guided masking probabilities across channels. *Our channel-wise firing-guided masking consistently achieves the best accuracy on both SEW-R19 and MaxFormer.* The Grad-CAM results in Figure 5(a) further demonstrate that *FGFKD* yields feature responses more consistent with the teacher ANN, highlighting the importance of preserving the correspondence between channel-wise firing activity and feature transfer.

Ablation on sample-wise firing-guided reweighting in *FGLKD*. As reported in Table 6, *Fixed* removes firing guidance and instead uses a sample-independent non-target weight, whereas *Shuffled* permutes the firing-guided weights across samples. *FGLKD* consistently achieves the best performance. Moreover, Figure 5(b) shows that *FGLKD* substantially reduces the increase of D_{nt} with r_b, corroborating the effectiveness of sample-wise firing-guided reweighting.

Ablation on loss components. As shown in Table 7, applying $\mathcal{L}_{\text{FGFKD}}$ alone improves the CF10/CF100 accuracy from $93.24\%/70.84\%$ to $96.94\%/82.02\%$ on SEW-R19, while $\mathcal{L}_{\text{FGLKD}}$ achieves $96.43\%/81.09\%$. Similar improvements are observed on MaxFormer. Combining both losses further increases the accuracy to $97.42\%/83.26\%$ on SEW-R19 and $97.98\%/86.56\%$ on MaxFormer, confirming the effectiveness of both firing-guided feature- and logits-based distillation.

6 Conclusion

In this work, we reveal that student firing activity is closely related to knowledge transfer from ANN to SNN. Based on this observation, we propose Firing-Guided Knowledge Distillation for SNNs (*FGKDSNN*), which leverages channel-wise firing activity for feature distillation and sample-wise firing activity for logits distillation. Extensive experiments show that *FGKDSNN* consistently outperforms existing distillation methods across CNN- and Transformer-based SNNs on image classification, object detection, and semantic segmentation. We anticipate extending firing-guided distillation to more diverse SNN architectures, which holds the potential to further improve the performance of SNN.

AI USE STATEMENT

Large Language Models (LLMs) were used solely for language polishing, including sentence rephrasing, grammar checking, and improving readability. All research ideas, methodological design, code implementation, experiments, and analyses were independently developed and conducted by the authors. The authors take full responsibility for the content of the manuscript.

REFERENCES

Tong Bu, Wei Fang, Jianhao Ding, PengLin Dai, Zhaofei Yu, and Tiejun Huang. Optimal ann-snn conversion for high-accuracy and ultra-low-latency spiking neural networks. *arXiv preprint arXiv:2303.04347*, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Gamaleldin Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. Large margin deep networks for classification. *Advances in neural information processing systems*, 31, 2018.

Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in neural information processing systems*, 34:21056–21069, 2021.

Yuetong Fang, Deming Zhou, Ziqing Wang, Hongwei Ren, ZeCui Zeng, Lusong Li, Shibo Zhou, and Renjing Xu. Spiking transformers need high frequency information. *arXiv preprint arXiv:2505.18608*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Di Hong, Jiangrong Shen, Yu Qi, and Yueming Wang. Lasnn: Layer-wise ann-to-snn distillation for effective and efficient training in deep spiking neural networks. *arXiv preprint arXiv:2304.09101*, 2023.

Li Hongmin, Liu Hanchao, Ji Xiangyang, Li Guoqi, and Shi Luping. Cifar10-dvs: An event-stream dataset for object classification. *Frontiers in Neuroscience*, 11, 2017.

Zihan Huang, Xinyu Shi, Zecheng Hao, Tong Bu, Jianhao Ding, Zhaofei Yu, and Tiejun Huang. Towards high-performance spiking transformers from ann to snn conversion. In *Proceedings of the 32nd ACM international conference on multimedia*, pp. 10688–10697, 2024.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. URL https://arxiv.org/abs/1405.0312.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 936–944. Ieee, 2017.

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

Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. 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, 2019.

Wachirawit Ponghiran and Kaushik Roy. Spiking neural networks with improved inherent recurrence dynamics for sequential learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8001–8008, 2022.

Xuerui Qiu, Jieyuan Zhang, Wenjie Wei, Honglin Cao, Junsheng Guo, Rui-Jie Zhu, Yimeng Shan, Yang Yang, Malu Zhang, and Haizhou Li. Quantized spike-driven transformer. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=5J9B7Sb8rO.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019. ISSN 1573-1405. doi: 10.1007/s11263-019-01228-7. URL http://dx.doi.org/10.1007/s11263-019-01228-7.

Guobin Shen, Dongcheng Zhao, Tenglong Li, Jindong Li, and Yi Zeng. Are conventional snns really efficient? a perspective from network quantization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27528–27537. IEEE, 2024.

Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural networks*, 111:47–63, 2019.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. F-divergence minimization for sequence-level knowledge distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10817–10834, 2023.

Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12:331, 2018.

Qi Xu, Yaxin Li, Jiangrong Shen, Jian K Liu, Huajin Tang, and Gang Pan. 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*, pp. 7886–7895, 2023.

Zekai Xu, Kang You, Qinghai Guo, Xiang Wang, and Zhezhi He. Bkdsnn: Enhancing the performance of learning-based spiking neural networks training with blurred knowledge distillation. In *European Conference on Computer Vision*, pp. 106–123. Springer, 2024.

Zhendong Yang, Ailing Zeng, Zhe Li, Tianke Zhang, Chun Yuan, and Yu Li. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17139–17148. IEEE, 2023.

Man Yao, Jiakui Hu, Guangshe Zhao, Yaoyuan Wang, Ziyang Zhang, Bo Xu, and Guoqi Li. Inherent redundancy in spiking neural networks. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16878–16888. IEEE, 2023a.

Man Yao, Jiakui Hu, Zhaokun Zhou, Li Yuan, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer, 2023b. URL https://arxiv.org/abs/2307.01694.

Man Yao, Jiakui Hu, Tianxiang Hu, Yifan Xu, Zhaokun Zhou, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips, 2024. URL https://arxiv.org/abs/2404.03663.

Man Yao, Xuerui Qiu, Tianxiang Hu, Jiakui Hu, Yuhong Chou, Keyu Tian, Jianxing Liao, Luziwei Leng, Bo Xu, and Guoqi Li. Scaling spike-driven transformer with efficient spike firing approximation training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(4):2973–2990, 2025. doi: 10.1109/TPAMI.2025.3530246.

Kang You, Zekai Xu, Chen Nie, Zhijie Deng, Qinghai Guo, Xiang Wang, and Zhezhi He. Spikezip-tf: Conversion is all you need for transformer-based snn. In *Proceedings of Forty-First International conference on Machine Learning (ICML)*, 2024.

Kairong Yu, Chengting Yu, Tianqing Zhang, Xiaochen Zhao, Shu Yang, Hongwei Wang, Qiang Zhang, and Qi Xu. Temporal separation with entropy regularization for knowledge distillation in spiking neural networks. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8806–8816. IEEE, 2025.

Weihao Yu, Chenyang Si, Pan Zhou, Mi Luo, Yichen Zhou, Jiashi Feng, Shuicheng Yan, and Xinchao Wang. Metaformer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2):896–912, 2023.

Yuxuan Zhang, Yuhang Sun, Wen Yao, Yue Deng, and Hongjue Li. Bi-spectrum distillation: Addressing spectral mismatch in ann-snn knowledge transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11953–11962, June 2022.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng Yan, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer, 2022.

Table A1: **Training setups of *FGKDSNN* on ImageNet.**

config	CNN-based SNN	Transformer-based SNN
time-steps T	4	4
optimizer	SGD	LAMB
learning rate	0.1	6×10^{-4}
momentum / drop path	0.9 / –	– / 0.1
weight decay	0	0.05
warmup epochs	0	15
training epochs	320	300
lr scheduler	cosine	cosine
batch size per GPU	32	64
input size	224×224	224×224
$p_{\min}, p_{\max}$	0.25, 0.75	0.25, 0.75
$w_{\min}, w_{\max}$	0.25, 1.0	0.25, 1.0
α, β	2×10^{-3}, 0.5	2×10^{-3}, 0.5
temperature $\mathcal{T}$	1.0	1.0

Table A2: **Training setups of *FGKDSNN* on CIFAR10, CIFAR100 and CIFAR10-DVS.**

config	CNN-based SNN	Transformer-based SNN
time-steps T	4 / 8	4 / 8
optimizer	AdamW	AdamW
learning rate	1.5×10^{-3}	1.5×10^{-3}
momentum / drop path	– / –	– / –
weight decay	0.06	0.06
warmup epochs	20	20
training epochs	400	400
lr scheduler	cosine	cosine
batch size per GPU	64	64
input size	32×32	32×32
$p_{\min}, p_{\max}$	0.25, 0.75	0.25, 0.75
$w_{\min}, w_{\max}$	0.25, 1.0	0.25, 1.0
α, β	2×10^{-3}, 0.5	2×10^{-3}, 0.5
temperature $\mathcal{T}$	1.0	1.0

Table A3: **Training setups of *FGKDSNN* for object detection on COCO.**

config	SDT-v3	QSDFormer
time-steps T	4	4
optimizer	AdamW	AdamW
learning rate	2×10^{-4}	2×10^{-4}
momentum / drop path	– / 0	– / 0
weight decay	0.05	0.05
warmup iterations	1,000	1,000
training epochs	12	15
lr scheduler	multi-step	multi-step
batch size per GPU	4	8 (Tiny) / 4 (Large)
input size	multi-scale	multi-scale
$p_{\min}, p_{\max}$	0.25, 0.75	0.25, 0.75
$w_{\min}, w_{\max}$	0.25, 1.0	0.25, 1.0
α, β	0.2, 0.1	0.2, 0.1
temperature $\mathcal{T}$	1.0	1.0

Table A4: **Training setups of *FGKDSNN* for semantic segmentation on ADE20K.**

config	SDT-v3	QSDFormer
time-steps T	4	4
optimizer	AdamW	AdamW
learning rate	1×10^{-3}	1×10^{-3}
momentum / drop path	– / 0	– / 0
weight decay	0.005	0.005
warmup iterations	1,500	1,500
training iterations	160,000	160,000
lr scheduler	polynomial	polynomial
batch size per GPU	16	16
input size	512×512	512×512
$p_{\min}, p_{\max}$	0.25, 0.75	0.25, 0.75
$w_{\min}, w_{\max}$	0.25, 1.0	0.25, 1.0
α, β	0.2, 0.1	0.2, 0.1
temperature $\mathcal{T}$	1.0	1.0

Appendix

A1 Implementation Details

Image Classification. As tabulated in Table A1 and Table A2, we summarize the detailed implementation details on image classification tasks. On ImageNet, we train both CNN- and Transformer-based students for T=4 with input size of 224×224. For CNN-based SNNs, we leverage SGD with an initial learning rate of 0.1, momentum of 0.9, and zero weight decay, and train the models for 320 epochs without warmup using a per-GPU batch size of 32. For Transformer-based SNNs, we adopt LAMB with a base learning rate of 6×10^{-4}, drop path rate of 0.1, and weight decay of 0.05, and train for 300 epochs with 15 warmup epochs using a per-GPU batch size of 64. A cosine learning-rate schedule is adopted for both architectures. On CIFAR-10 and CIFAR-100, we set T=4, while T=8 is used for CIFAR10-DVS. For these three datasets, we adopt the same training recipe for CNN- and Transformer-based students: AdamW with a learning rate of 1.5×10^{-3} and weight decay of 0.06, 400 training epochs with 20 warmup epochs, a cosine learning-rate schedule, a per-GPU batch size of 64, and 32×32 inputs. All ImageNet experiments are conducted on eight NVIDIA A800 80GB GPUs, while the CIFAR-10, CIFAR-100, and CIFAR10-DVS experiments are conducted on a single NVIDIA GeForce RTX 4090 GPU. Across all image-classification experiments, we set $p_{\min}$=0.25, $p_{\max}$=0.75, $w_{\min}$=0.25, $w_{\max}$=1.0, α=2×10^{-3}, β=0.5, and $\mathcal{T}$=1.0.

Object Detection. On COCO, we adopt SDT-v3 and QSDFormer as student SNN backbones within Mask R-CNN, using T=4 time steps. Both students are optimized with AdamW using an initial learning rate of 2×10^{-4}, weight decay of 0.05, and a drop path rate of zero. We train SDT-v3 for 12 epochs and QSDFormer for 15 epochs. Both use linear warmup for the first 1,000 iterations, starting from 0.001 times the base learning rate, followed by a multi-step learning-rate schedule that decays the learning rate by a factor of 0.1 at epochs 8 and 11. The per-GPU batch size is 4 for SDT-v3, 8 for QSDFormer-Tiny, and 4 for QSDFormer-Large, with a global batch size of 8 for all models. We leverage multi-scale resizing, random horizontal flipping, and random cropping during training. At inference, we resize the input images to a maximum size of 1333×800 while preserving the aspect

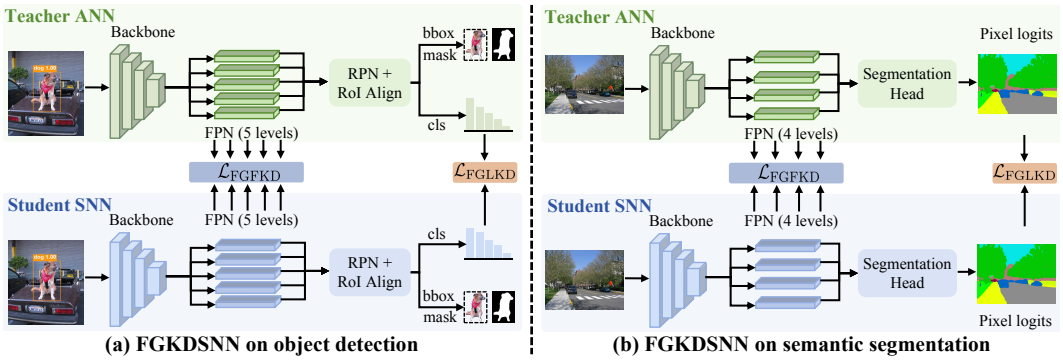


Figure A1: **Overview of the Firing-Guided Knowledge Distillation (*FGKDSNN*) framework on object detection and semantic segmentation.**

ratio. As illustrated in Figure A1(a), *FGFKD* is applied to the five FPN feature levels of the teacher and student before the detection heads. *FGLKD* is applied to the RoI classification logits using shared RoIs, where only positive RoIs and the 80 foreground classes are involved in distillation; the RPN, bounding-box regression, and mask outputs remain supervised only by their original task losses. For *FGKDSNN*, we set $p_{\min}$=0.25, $p_{\max}$=0.75, $w_{\min}$=0.25, $w_{\max}$=1.0, α=0.2, β=0.1, and $\mathcal{T}$=1.0.

Semantic Segmentation. On ADE20K, we adopt SDT-v3 and QSDFormer as student SNN backbones with T=4 time steps and 512×512 training crops. Both architectures are trained for 160,000 iterations using AdamW with an initial learning rate of 1×10^{-3}, weight decay of 0.005, and a drop path rate of zero. The segmentation head uses a learning-rate multiplier of 2. We apply linear warmup over the first 1,500 iterations, starting from 10^{-6} times the base learning rate, followed by polynomial learning-rate decay with a power of 1.0 and a minimum learning rate of zero. We use two GPUs with a per-GPU batch size of 16, yielding a global batch size of 32. Data augmentation includes random resizing with a scale ratio in $[0.5, 2.0]$, random cropping, random horizontal flipping, and photometric distortion. As illustrated in Figure A1(b), *FGFKD* is applied to the four multi-scale pyramid features of the teacher and student before the segmentation head. *FGLKD* is further performed pixel-wise on the 150-class segmentation logits at their native output resolution, where only valid pixels participate in distillation and pixels with the ignore label 255 are excluded. The original pixel-wise segmentation loss is retained as the task objective. We use the same *FGKDSNN* settings as for object detection: $p_{\min}$=0.25, $p_{\max}$=0.75, $w_{\min}$=0.25, $w_{\max}$=1.0, α=0.2, β=0.1, and $\mathcal{T}$=1.0.

A2 Visualization Results

As shown in Figures A2 and A3, we further visualize the predictions and intermediate feature responses of the teacher ANN, the original student SNN, and the student trained with *FGKDSNN* on object detection and semantic segmentation. Compared with the original SNN, *FGKDSNN* produces feature responses that are noticeably more consistent with those of the teacher ANN, with stronger activation on task-relevant regions and reduced responses in irrelevant areas. This improved feature alignment is also reflected in the final predictions, where *FGKDSNN* yields more accurate object localization and segmentation results. These visualizations further demonstrate that firing-guided knowledge distillation effectively transfers informative representations from the teacher ANN to the student SNN and generalizes well to dense prediction tasks.



Figure A2: **Visualization results of ANN, SNN and *FGKDSNN* on object detection.**

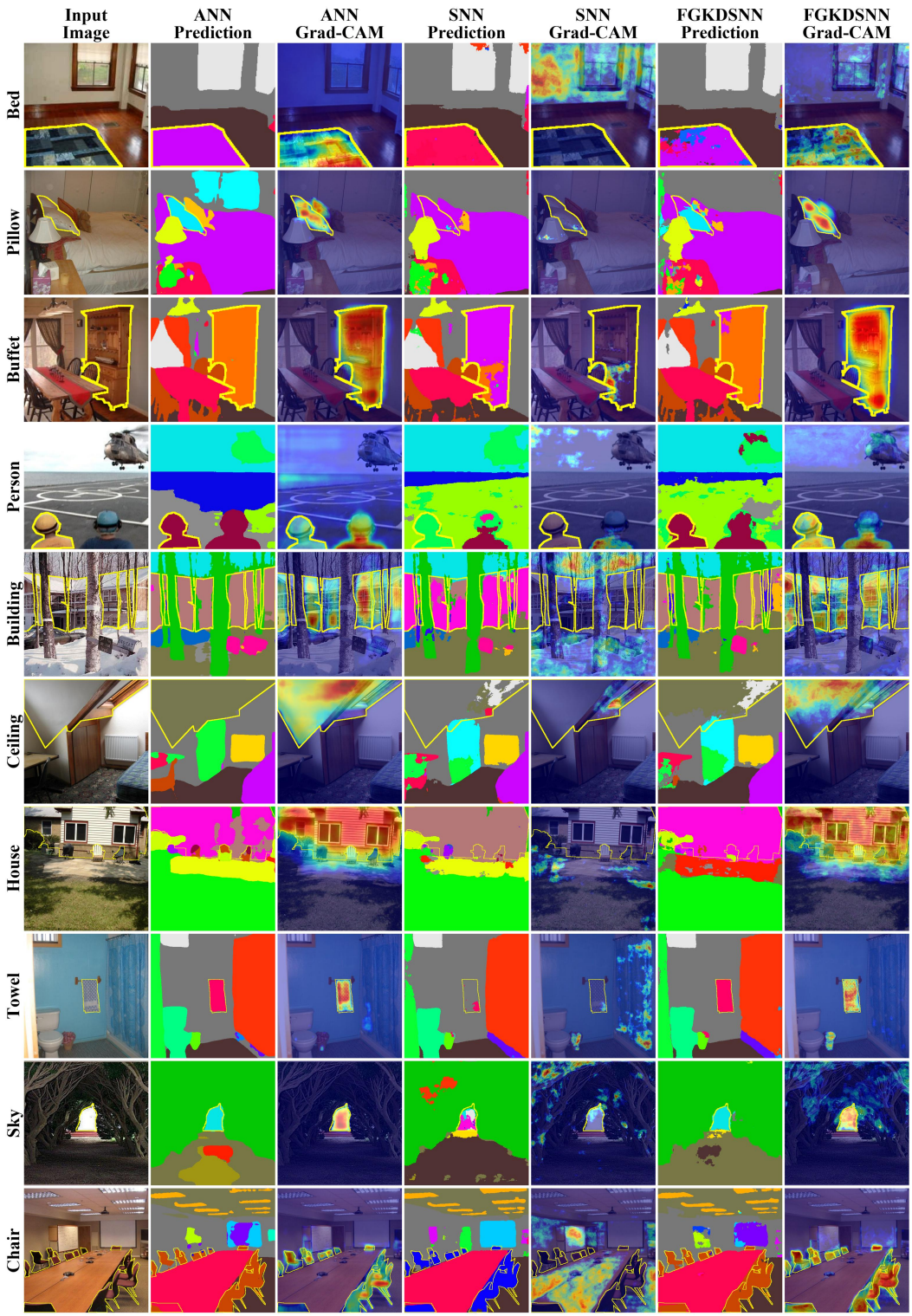


Figure A3: **Visualization results of ANN, SNN and *FGKDSNN* on semantic segmentation.**