STARS: SPIKE TAIL-AWARE RELATIONAL SYNTHESIS FOR ANN-TO-SNN DATA-FREE KNOWLEDGE DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

SNNs offer a route to efficient inference, but their performance still trails that of ANNs. ANN-to-SNN knowledge distillation helps narrow this gap, yet the original training data are often unavailable. Existing data-free knowledge distillation (DFKD) methods synthesize surrogate data using teacher-side priors, especially BN statistics, whose first two moments do not determine even a neuron's pre-reset threshold-crossing behavior. We propose Spike Tail-Aware Relational Synthesis (STARS), an add-on for BN-guided ANN-to-SNN DFKD that augments synthesis with Relational Consistency Alignment, matching cross-sample teacher–student relations, and Tail-Aware Regularization, matching soft exceedance probabilities over teacher-derived feature thresholds. The original teacher priors remain active during synthesis. Experiments on CIFAR-10, CIFAR-100, and Tiny-ImageNet across multiple ANN–SNN pairs show consistent gains over three DFKD baselines, reaching absolute accuracy gains of 4.6% on CIFAR-10 and 6.7% on CIFAR-100. Paired multi-seed and resource measurements further characterize reliability and training cost in the evaluated settings.

1 INTRODUCTION

Spiking neural networks (SNNs) have been widely regarded as the third generation of neural networks because they compute with discrete spikes and explicitly model temporal neural dynamics (Maass, 1997; Neftci et al., 2019; Roy et al., 2019; Wu et al., 2018; Zheng et al., 2021). Their event-driven communication paradigm offers an attractive route to energy-efficient and low-latency inference, especially for edge intelligence and neuromorphic hardware (Roy et al., 2019; Davies et al., 2018; Fang et al., 2021b; Deng & Gu, 2021; Qian et al., 2025). Recent progress in direct training and surrogate-gradient learning has further improved the practicality of modern SNNs (Neftci et al., 2019; Li et al., 2021; Yao et al., 2022; Zhou et al., 2023; Wu et al., 2022; Yang et al., 2022; Wei et al., 2023; Guo et al., 2022). However, directly trained SNNs still often suffer from limited optimization stability and suboptimal task accuracy in practical applications, which has motivated ANN-to-SNN knowledge distillation (Hinton et al., 2015; Romero et al., 2014; Zagoruyko & Komodakis, 2016; Park et al., 2019; Tian et al., 2019) as a way to transfer richer supervision from strong ANN teachers while preserving efficient SNN inference (Guo et al., 2023; Xu et al., 2024; Ye et al., 2025; Yu et al., 2025; Yang et al., 2025).

In many practical settings, however, the original training data are unavailable at distillation time because of privacy, copyright, transmission, or storage constraints (Yoo et al., 2019; Fang et al., 2019; Binici et al., 2022). This setting is especially relevant in edge deployment scenarios for SNNs, where pretrained models may remain accessible while the original data can no longer be retained locally or transmitted across devices. Data-free knowledge distillation (DFKD) addresses this scenario by synthesizing surrogate samples from a pretrained teacher and then using them for transfer (Micaelli & Storkey, 2019; Nayak et al., 2019; Chen et al., 2019; Yin et al., 2020; Fang et al., 2021a; Tran et al., 2024). As a result, DFKD provides an attractive route for ANN-to-SNN transfer when real data are inaccessible.

Previous work has shown that the quality of synthetic data depends critically on the priors used during synthesis. Early methods explored zero-shot and generator-based formulations (Micaelli &

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.

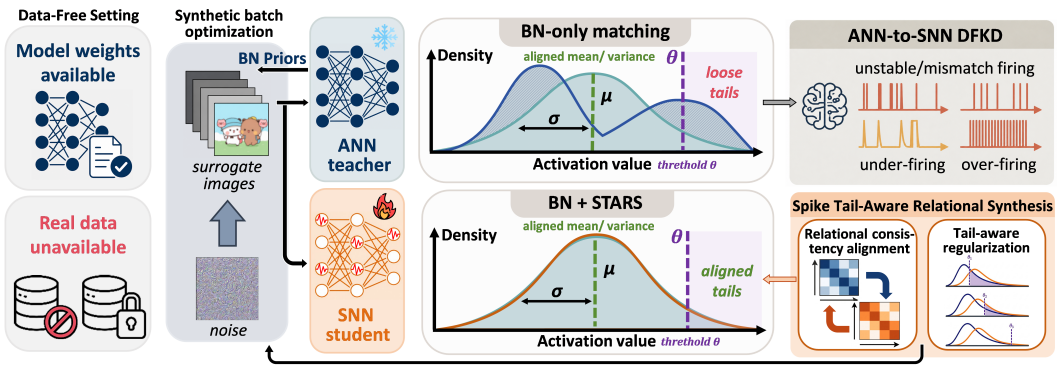

Figure 1: Overview of STARS. Teacher-side BN matching constrains low-order statistics but leaves threshold-relevant tails undetermined. STARS adds relational consistency alignment and tail-aware regularization during synthesis to provide additional teacher-anchored supervision for the SNN student.

Storkey, 2019; Chen et al., 2019), while later works demonstrated the importance of internal teacher statistics, especially Batch Normalization (BN) statistics, for reconstructing more faithful surrogate data (Yin et al., 2020). Subsequent methods further improved data synthesis with stronger diversity, contrastive, and optimization priors (Fang et al., 2021a; Tran et al., 2024; Hu et al., 2023a;b; Wei et al., 2024b;a; Hu et al., 2026). Teacher-side BN priors remain important, but synthesis can also use student feedback: CMI and NAYER use teacher–student disagreement during synthesis (Fang et al., 2021a; Tran et al., 2024). These methods already combine teacher priors with student feedback.

However, existing synthesis objectives need not capture the feature responses relevant to an SNN student. Student feedback at the prediction level does not explicitly match the feature-tail information associated with integration, thresholding, and reset. Samples satisfying ANN BN constraints may therefore still elicit different threshold-sensitive responses. We focus on the residual ambiguity in BN moments even when synthesis uses student feedback. We examine it through a local first-crossing analysis and add complementary constraints on cross-sample relations and feature-tail probabilities.

To address this mismatch, we propose **S**pike **T**ail-**A**ware **R**elational **S**ynthesis (**STARS**), a framework for BN-guided ANN-to-SNN DFKD with corresponding teacher–student stages. As shown in Figure 1, STARS combines *Relational Consistency Alignment* (RCA), which preserves cross-sample relational consistency between teacher and student, with *Tail-Aware Regularization* (TAR), which regularizes soft exceedance behavior over teacher-derived feature thresholds motivated by local LIF threshold sensitivity. Together, they complement BN matching and make the synthesized data more informative for SNN training.

Our main contributions are:

- We study data-free distillation from a pretrained ANN teacher to a separately initialized SNN student without ANN-to-SNN weight conversion, and identify a BN-moment ambiguity relevant to the student's threshold dynamics.
- We propose STARS, an add-on for BN-guided synthesis that combines RCA and TAR to preserve cross-sample relations and constrain feature-tail statistics at corresponding teacher–student stages.
- We plug STARS into multiple conventional DFKD methods and evaluate it on CIFAR-10, CIFAR-100, and Tiny-ImageNet, where STARS consistently improves the evaluated CNN teacher–student pairs.

2 Preliminary and Related Work

Spiking neuron dynamics. Spiking neural networks (SNNs) replace continuous activations with binary spikes and perform computation through membrane integration, threshold-triggered firing and reset (Neftci et al., 2019). Following the widely used discrete-time formulation of the leaky

integrate-and-fire (LIF) neuron (Gerstner et al., 2014), we denote by $X[t]$, $H[t]$, $S[t]$, and $V[t]$ the input current, the sub-threshold membrane potential, the output spike, and the post-reset membrane potential at time step t, respectively. The neuron dynamics is written as

$$H[t]=V[t-1]+\frac{1}{\tau}(X[t]-(V[t-1]-V_{\mathrm{r}})),\ S[t]=\Theta(H[t]-V_{\mathrm{th}}),\ V[t]=H[t](1-S[t])+V_{\mathrm{r}}S[t], \quad (1)$$

where τ is the membrane time constant, V_{th} is the firing threshold, V_{r} is the reset potential, and $\Theta(\cdot)$ is the Heaviside step function. As shown in Eq. (1), spiking neuron dynamics differ fundamentally from those of ANNs, as their outputs depend on temporally evolving spiking states shaped by membrane integration, thresholding, and reset, rather than static activation mappings.

Knowledge distillation for SNNs. Knowledge distillation (KD) (Hinton et al., 2015; Romero et al., 2014; Zagoruyko & Komodakis, 2016; Park et al., 2019; Tian et al., 2019) has demonstrated strong effectiveness for SNN training (Guo et al., 2023; Xu et al., 2024; Yu et al., 2025; Yang et al., 2025), where knowledge from a powerful teacher model is transferred to an SNN student to improve its performance. For a mini-batch $\mathcal{B}$, a student model f_S is typically trained to mimic a teacher model f_T by matching softened predictions:

$$\mathcal{L}_{\mathrm{KD}} = \frac{1}{|\mathcal{B}|}\sum_{x\in\mathcal{B}}\left[\mathrm{KL}(\sigma(f_T(x)/\gamma)\mid\sigma(f_S(x)/\gamma))\right], \quad (2)$$

where $\sigma(\cdot)$ is the softmax and γ is the temperature. Since existing SNNs still lag behind their ANN counterparts in accuracy, more studies have explored distilling knowledge from ANN teachers to SNNs, aiming to improve SNN performance while preserving their energy-efficiency advantage. Xu et al. (2024) introduces blurred feature distillation to better mimic ANN representations, Yu et al. (2025) performs temporally separated logit distillation with entropy regularization, and Yang et al. (2025) aligns rate-based SNN features with ANN features via hybrid block-wise replacement.

However, many practical scenarios do not permit access to the original training data, due to privacy, storage, or deployment constraints. This challenge is particularly relevant for SNNs, which are often deployed in resource-constrained and edge settings, motivating the study of transferring knowledge to students without access to the original training data (Roy et al., 2019; Yin et al., 2020).

Data-free knowledge distillation (DFKD). To address the absence of original training data caused by privacy, storage, and deployment constraints, researchers have developed DFKD, which synthesizes surrogate data from a pretrained teacher to achieve knowledge transfer without access to the real training set. Chen et al. (2019) synthesizes data by exploiting teacher outputs and class-discriminative priors, Yin et al. (2020) reconstructs inputs by matching the internal batch-normalization (BN) statistics stored in the teacher, Fang et al. (2021a) improves sample diversity and transferability through contrastive model inversion on top of teacher-side priors, and Tran et al. (2024) further enhances synthesis quality within the BN-guided DFKD framework (Hu et al., 2023a;b; Wei et al., 2024b;a; Hu et al., 2026). Teacher BN statistics are an important prior, but do not identify the training distribution. AdaDFKD already studies adaptive inter-sample relationships (Li et al., 2024); our contribution is their joint use with feature-tail probes for ANN-to-SNN synthesis, not relational matching in isolation. PrivateSNN uses synthetic data for ANN-to-SNN conversion (Kim et al., 2022), whereas we distill into a separately initialized SNN without transferring ANN weights.

Specifically, for the l-th BN layer of the teacher network, let $\hat{\mu}_l$ and $\hat{\sigma}_l^2$ be the running mean and variance estimated from the real training data. For a synthetic mini-batch $\mathcal{B}=\{x_i\}_{i=1}^{B}$, let $\mu_l(\mathcal{B})$ and $\sigma_l^2(\mathcal{B})$ denote the empirical mean and variance of the corresponding pre-BN activations. A standard BN-matching regularization is defined as

$$\mathcal{L}_{\mathrm{BN}}(\mathcal{B}) = \sum_l \Big(\|\mu_l(\mathcal{B})-\hat{\mu}_l\|_2^2 + \|\sigma_l^2(\mathcal{B})-\hat{\sigma}_l^2\|_2^2\Big). \quad (3)$$

The synthetic batch is then optimized by combining BN matching with teacher-oriented objectives such as class confidence and image regularization:

$$\mathcal{B}^\star = \arg\min_{\mathcal{B}}\Big[\mathcal{L}_{\mathrm{cls}}(\mathcal{B};f_T)+\lambda_{\mathrm{BN}}\mathcal{L}_{\mathrm{BN}}(\mathcal{B})+\lambda_{\mathrm{reg}}\mathcal{R}(\mathcal{B})\Big]. \quad (4)$$

Here $\mathcal{R}$ collects the base method's other synthesis terms and may include student-dependent adversarial feedback. BN statistics remain a teacher-side anchor. In ANN-to-SNN distillation, however, the

statistics in Eq. (3) come from continuous ANN activations, whereas the same batch induces student responses shaped by the LIF dynamics in Eq. (1). This discrepancy motivates a natural question:

Are BN moments sufficient to determine the responses that matter for a SNN student?

3 Analysis: BN Priors are Under-Constrained for SNNs

In this section, we answer the above question with the following analysis. We consider the most common direct coding setting for SNNs, where a static image is presented identically at each time step. Under this setting, we analyze using a BN-constrained ANN layer together with its corresponding SNN layer. This local analysis isolates a threshold-related ambiguity; it does not assume constant currents at every layer of a deep SNN. Let $a^l = W^l z^{l-1} + b^l$ denote the pre-BN activation of the l-th ANN teacher layer. For simplicity, we omit the layer index and write $a = wz + b$. Corresponding to the BN statistics $\mu_l(\mathcal{B})$ and $\sigma_l^2(\mathcal{B})$ in Eq. (3), we write their distribution-level counterparts under the induced synthetic distribution P as

$$\mu_A(P) = \mathbb{E}_{a\sim P}[a], \qquad \sigma_A^2(P) = \mathrm{Var}_{a\sim P}(a). \tag{5}$$

BN matching alone cannot distinguish distributions with the same (μ_A, σ_A^2). To show why this can matter to an SNN, consider a constant input current a, $V_\mathrm{r} = 0$, and the membrane trajectory *up to its first spike*. Before any reset, Eq. (1) gives

$$H_\mathrm{pre}[t] = \Big(1-\frac{1}{\tau}\Big)H_\mathrm{pre}[t-1] + \frac{1}{\tau}a, \quad H_\mathrm{pre}[0] = 0 \quad \Rightarrow \quad H_\mathrm{pre}[t] = \beta_t a, \quad \beta_t := 1-\Big(1-\frac{1}{\tau}\Big)^t. \tag{6}$$

For $\tau > 1$ and $V_\mathrm{th} > 0$, β_t increases with t. Thus the event that a first threshold crossing has occurred *by* time t is

$$\{T_\mathrm{first} \le t\} = \{\max_{s\le t} H_\mathrm{pre}[s] \ge V_\mathrm{th}\} = \{a \ge \theta_t\}, \qquad \theta_t := V_\mathrm{th}/\beta_t, \tag{7}$$

For an activation distribution P, let $C_t(P)$ denote the probability of this first crossing by time t. Writing $\bar{F}_P(\theta) := \mathbb{P}_{a\sim P}(a \ge \theta)$, we obtain

$$C_t(P) = \mathbb{P}_{a\sim P}(T_\mathrm{first} \le t) = \bar{F}_P(\theta_t). \tag{8}$$

Now consider two activation distributions P and Q with identical BN moments,

$$\mu_A(P) = \mu_A(Q), \qquad \sigma_A^2(P) = \sigma_A^2(Q). \tag{9}$$

Their first-crossing probabilities can nevertheless differ:

$$C_t(P) - C_t(Q) = \bar{F}_P(\theta_t) - \bar{F}_Q(\theta_t). \tag{10}$$

Thus BN moments do not determine even this local, pre-reset threshold response. Reset after the first crossing cannot remove that counterexample, but Eqs. (6)–(10) do not characterize subsequent spikes or the firing rate of a deep SNN. At deeper layers, upstream spikes can change the inputs reaching each pre-LIF feature; the effectiveness of constraining those features is evaluated empirically. **This motivates matching threshold-relevant tail information in addition to BN moments.** A constructive example and scope discussion appear in Appendix B.

4 Methodology

Problem Statement. The local analysis in Sec. 3 shows that equal ANN-side BN moments need not imply equal first-crossing probabilities for an SNN neuron. This establishes an SNN-relevant ambiguity within BN-matched synthesis without assuming a closed-form description of deep, reset-aware dynamics. We address it by retaining the complete base synthesis objective and adding relational and tail constraints on pre-LIF features.

Motivation. We seek an add-on for BN-guided synthesis with compatible teacher–student stages. The goal is to retain the original BN prior while making the synthesized batches more informative for the SNN student by constraining additional threshold-relevant information that BN matching alone

does not determine. Meanwhile, since the LIF dynamics in Eq. (1) involve thresholding and reset, directly regularizing post-spike outputs is less suitable for optimization. Accordingly, we impose the additional constraint on *pre-activation* features, consistent with the analysis in Sec. 3, which characterizes the response in terms of the pre-threshold activation a and the effective thresholds θ_t.

The Proposed Framework: STARS. In this paper, we design ***S**pike **T**ail-**A**ware **R**elational **S**ynthesis* (STARS), a plug-and-play regularization framework for ANN-to-SNN DFKD. STARS is built on top of the original BN-guided synthesis objective and introduces two complementary components: *Relational Consistency Alignment* (RCA), which preserves global semantic structure by aligning cross-sample relational consistency between teacher and student, and *Tail-Aware Regularization* (TAR), which regularizes the threshold-relevant tail behavior that BN matching alone does not determine to reduce local threshold-induced ambiguity. In this way, STARS preserves the original teacher-side prior while making the synthesized batches more informative for the SNN student.

Our STARS is imposed on *pre-activation* features, *e.g.,* before ReLU in the ANN teacher and before LIF in the SNN student. This choice is motivated by the LIF dynamics in Eq. (1): due to thresholding and reset, post-spike responses are less amenable to direct optimization. In contrast, pre-activation features provide a smoother and more comparable space for regularization. This design is also consistent with the analysis in Sec. 3, where the student response is characterized by the pre-threshold activation and the effective thresholds.

For each selected layer $l \in \mathcal{S}$, let $F_T^l \in \mathbb{R}^{B\times C\times N}$ denote the teacher pre-activation feature and $F_S^l \in \mathbb{R}^{T\times B\times C\times N}$ denote the corresponding student feature produced by the current synthetic batch $\mathcal{B}$, where B is the batch size, C is the channel dimension, N is the number of spatial locations after flattening, and T is the number of time steps. Stage correspondence and uniform layer weights are specified in Appendix D. We obtain global features by averaging the teacher feature over spatial locations and the student feature over both time steps and spatial locations:

$$z_T^l = \frac{1}{N}\sum_{n=1}^{N} F_{T,n}^l \in \mathbb{R}^{B\times C}, \qquad z_S^l = \frac{1}{TN}\sum_{t=1}^{T}\sum_{n=1}^{N} F_{S,t,n}^l \in \mathbb{R}^{B\times C}. \tag{11}$$

where $F_{T,n}^l$ denotes the n-th spatial slice of F_T^l, and $F_{S,t,n}^l$ denotes the slice of F_S^l at time step t and spatial index n. The resulting z_T^l and z_S^l form a unified batch-wise channel representation, where rows index samples and columns index channels, corresponding to the global pre-activation features of the ANN teacher and the SNN student, respectively.

Relational Consistency Alignment (RCA). The first component of STARS addresses a different ambiguity from the threshold example in Sec. 3: BN moments also leave cross-sample relations unconstrained. Consequently, BN-matched batches can induce different relational structures in the ANN teacher and SNN student. At deeper layers, such differences can change the features received downstream.

To mitigate this effect, we introduce a global *Relational Consistency Alignment* (RCA) objective that aligns the pairwise relations among samples within each batch between teacher and student. By anchoring the batch-level relational structure seen by the student to that induced by the teacher, RCA regularizes the global organization of synthetic samples and promotes more consistent teacher-student alignment throughout the network.

Specifically, we first apply row-wise ℓ_2 normalization to the global pre-activation features z_T^l and z_S^l in Eq. (11), yielding $\tilde{z}_T^l, \tilde{z}_S^l \in \mathbb{R}^{B\times C}$. We then construct cosine relation matrices

$$G_T^l = \tilde{z}_T^l(\tilde{z}_T^l)^\mathsf{T}, \qquad G_S^l = \tilde{z}_S^l(\tilde{z}_S^l)^\mathsf{T}, \tag{12}$$

where each entry measures the cosine similarity between a pair of samples in the batch. Since the diagonal entries are trivially equal to one, we match only the off-diagonal relations,

$$\ell_{\mathrm{rca}}^l = \frac{1}{|\mathcal{P}|}\sum_{(i,j)\in\mathcal{P}} \left(G_{S,ij}^l - G_{T,ij}^l\right)^2, \qquad \mathcal{P} := \{(i,j) \mid 1 \le i < j \le B\}, \tag{13}$$

and aggregate them over layers as

$$\mathcal{L}_{\mathrm{RCA}} = \frac{1}{\sum_{l\in\mathcal{S}} w_l}\sum_{l\in\mathcal{S}} w_l\, \ell_{\mathrm{rca}}^l. \tag{14}$$

Algorithm 1 STARS-enhanced surrogate synthesis for ANN-to-SNN DFKD.

Require: Teacher f_T with BN statistics $\{\hat{\mu}_l, \hat{\sigma}_l^2\}$, SNN student f_S, selected layers $\mathcal{S}$, synthesis steps K, simulation steps T, threshold count M, weights $\lambda_{\mathrm{BN}}, \lambda_{\mathrm{reg}}, \lambda_{\mathrm{rca}}, \lambda_{\mathrm{tar}}$
Ensure: Final synthetic batch $\mathcal{B}^\star$ for downstream ANN-to-SNN distillation

Init: Randomly initialize the synthetic batch $\mathcal{B}$
# Synthesis stage (optimize surrogate data under teacher priors and STARS constraints)
for $k = 1, \ldots, K$ **do**
  Forward $\mathcal{B}$ through f_T and compute $\mathcal{L}_{\mathrm{cls}}(\mathcal{B}; f_T)$ together with the BN regularizer $\mathcal{L}_{\mathrm{BN}}(\mathcal{B})$ # Eq. 3
  Forward $\mathcal{B}$ through student f_S for T steps and collect pre-LIF student features $\{F_S^l\}_{l\in\mathcal{S}}$
  Collect the corresponding teacher features $\{F_T^l\}_{l\in\mathcal{S}}$
  # Layerwise STARS regularization at pre-LIF features
  for each $l \in \mathcal{S}$ **do**
    Time-average F_S^l and apply global average pooling to obtain z_T^l, z_S^l # Eq. 11
    RCA: Row-normalize z_T^l and z_S^l, form G_T^l and G_S^l, and compute ℓ_{rca}^l # Eq. 13
    Construct teacher-anchored thresholds $\{\theta_m^l\}_{m=1}^M$ from z_T^l
    TAR: Compute soft exceedance probabilities $p_T^{l,m}$ and $p_S^{l,m}$, then compute ℓ_{tar}^l # Eq. 16
  end for
  Aggregate layer losses into $\mathcal{L}_{\mathrm{RCA}}$ and $\mathcal{L}_{\mathrm{TAR}}$ # Eq. 14
  $\mathcal{L}_{\mathrm{DFKD}} \leftarrow \mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{BN}}\mathcal{L}_{\mathrm{BN}} + \lambda_{\mathrm{reg}}\mathcal{R}$ # Eq. 4; $\mathcal{L}_{\mathrm{total}} \leftarrow \mathcal{L}_{\mathrm{DFKD}} + \lambda_{\mathrm{rca}}\mathcal{L}_{\mathrm{RCA}} + \lambda_{\mathrm{tar}}\mathcal{L}_{\mathrm{TAR}}$ # Eq. 17
  Update $\mathcal{B} \leftarrow \mathcal{B} - \eta\nabla_{\mathcal{B}}\mathcal{L}_{\mathrm{total}}$ # update synthetic batch only
end for
return $\mathcal{B}^\star \leftarrow \mathcal{B}$

Here, $\mathcal{S}$ denotes the set of selected layers, $\mathcal{P}$ indexes all unordered sample pairs, and w_l controls the contribution of each layer. Unlike BN matching, which only constrains marginal activation statistics, RCA preserves the relative arrangement of samples in feature space. As a result, teacher and student are encouraged to agree not only on *what* activations are produced on average, but also on *how* synthetic samples are organized with respect to one another.

Tail-Aware Regularization (TAR). While RCA regularizes relations across samples, it does not address the local threshold sensitivity illustrated in Sec. 3. TAR regularizes *tail probabilities across multiple teacher-anchored thresholds*, providing a smooth proxy for pre-threshold feature behavior.

For each selected layer, we construct a threshold set $\{\theta_m^l\}_{m=1}^M$ from the teacher feature distribution, for example from fixed values or teacher quantiles. When normalization is used, both teacher and student features are normalized using teacher-derived statistics before thresholding, so that the thresholds are defined in a stable teacher-centric coordinate system. We then compute the soft exceedance probabilities

$$p_T^{l,m} = \frac{1}{BC}\sum_{b,c}\sigma\left(\frac{z_{T,bc}^l - \theta_m^l}{\delta}\right), \qquad p_S^{l,m} = \frac{1}{BC}\sum_{b,c}\sigma\left(\frac{z_{S,bc}^l - \theta_m^l}{\delta}\right), \tag{15}$$

where $\sigma(\cdot)$ is the sigmoid and $\delta > 0$ controls the softness of the threshold transition. Each quantity above can be interpreted as a differentiable estimate of the probability mass lying above threshold θ_m^l. The layer-wise regularization term ℓ_{tar}^l and the final tail-aware regularization objective $\mathcal{L}_{\mathrm{TAR}}$ are

$$\ell_{\mathrm{tar}}^l = \frac{1}{M}\sum_{m=1}^{M}\left(p_S^{l,m} - p_T^{l,m}\right)^2, \quad \mathcal{L}_{\mathrm{TAR}} = \frac{1}{\sum_{l\in\mathcal{S}} w_l}\sum_{l\in\mathcal{S}} w_l\,\ell_{\mathrm{tar}}^l. \tag{16}$$

The pre-reset result motivates tail information beyond BN moments; it does not derive the pooled TAR objective. TAR matches sigmoid-smoothed survival probabilities at finite teacher-derived probes, rather than physical membrane thresholds. Appendix B.4 bounds the corresponding hard-tail gap at those probes under an explicit near-threshold mass term. Pooling discards spatial and temporal order, so matching these features does not establish equal deep-network spike dynamics.

Overall objective. Combining the complete base DFKD objective with the two STARS modules above, we synthesize the surrogate batch by solving

$$\mathcal{B}^\star = \arg\min_{\mathcal{B}}\Big[\mathcal{L}_{\mathrm{DFKD}}(\mathcal{B}; f_T, f_S) + \lambda_{\mathrm{rca}}\cdot\mathcal{L}_{\mathrm{RCA}}(\mathcal{B}; f_T, f_S) + \lambda_{\mathrm{tar}}\cdot\mathcal{L}_{\mathrm{TAR}}(\mathcal{B}; f_T, f_S)\Big]. \tag{17}$$

Here, $\mathcal{L}_{\mathrm{DFKD}}$ retains all base synthesis terms, including teacher priors and student feedback when present; RCA constrains sample relations and TAR constrains feature tails. The teacher provides

Table 1: Summary results on CIFAR-10 and CIFAR-100. Models prefixed with "S-" denote spiking neural networks (SNNs), while unprefixed models are artificial neural networks (ANNs) by default. "–" denotes not reported. Teacher accuracy is reported in the order of CIFAR-10 / CIFAR-100.

Type	Method	Teacher	Teacher acc.	Student	T	CIFAR-10	CIFAR-100
KD	LaSNN	VGG-16	– / –	S-VGG-16	100	91.22	61.52
	Joint A-SNN	VGG-16	– / –	S-VGG-16	1	93.79	74.24
		VGG-16	– / –	S-VGG-16	4	–	–
	KDSNN	ResNet-19	95.60 / 79.78	S-ResNet-19	4	94.36	74.08
		ResNet-20	91.77 / 68.40	S-ResNet-20	4	89.03	60.18
	BKDSNN	ResNet-19	95.60 / 79.78	S-ResNet-19	4	94.64	74.95
		ResNet-20	91.77 / 68.40	S-ResNet-20	4	89.29	60.92
DFKD	DeepInversion	VGG-16	94.32 / 75.39	S-VGG-16	4	66.88	35.68
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	73.84	41.71
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	78.05	42.25
	DeepInversion + **STARS**	VGG-16	94.32 / 75.39	S-VGG-16	4	**70.12**	**39.72**
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	**76.93**	**46.18**
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	**80.97**	**46.73**
	CMI	VGG-16	94.32 / 75.39	S-VGG-16	4	80.20	55.91
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	83.13	58.02
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	85.44	60.72
	CMI + **STARS**	VGG-16	94.32 / 75.39	S-VGG-16	4	**83.67**	**58.96**
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	**85.85**	**60.28**
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	**88.02**	**63.49**
	NAYER	VGG-16	94.32 / 75.39	S-VGG-16	4	88.30	68.09
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	90.54	71.89
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	92.02	73.95
	NAYER + **STARS**	VGG-16	94.32 / 75.39	S-VGG-16	4	**92.89**	**74.79**
		ResNet-34	95.70 / 77.94	S-ResNet-18	4	**94.08**	**76.97**
		ResNet-19	96.30 / 80.48	S-ResNet-19	4	**95.73**	**78.42**

the target for both additions, while the current student is held fixed within each synthesis inner loop and updated only during subsequent distillation. Architecture-specific feature correspondences and this optimization separation are detailed in Appendix D. After obtaining $\mathcal{B}^\star$, we distill the student according to Eq. (2); the expanded objective is provided in Appendix I.

5 Experiments

Experimental setup. We evaluate STARS on CIFAR-10 and CIFAR-100 (Krizhevsky, 2009) using VGG- (Simonyan & Zisserman, 2014) and ResNet-based (He et al., 2016) ANN teacher / SNN student pairs, and report top-1 accuracy (%). Accuracy gains denote absolute differences, $\Delta A = A_{\text{STARS}} - A_{\text{base}}$, with both accuracies expressed in %. We further test on Tiny-ImageNet (Le & Yang, 2015), a larger benchmark with 200 classes and 64×64 images. All SNN students use direct coding and are evaluated with $T = 4$ simulation steps, while reported KD baselines follow their original settings. For fair comparison, DeepInversion and CMI are implemented under the efficient DFKD training protocol of FastDFKD (Fang et al., 2022). STARS is plugged into each BN-guided synthesis step by adding RCA and TAR to the original BN-matching objective, while keeping the baseline teacher-side priors and distillation schedules unchanged. We set $\lambda_{\text{rca}} = \lambda_{\text{tar}} = 1$ and fix the TAR softness parameter to $\delta = 0.5$ unless otherwise specified. Further details are provided in Appendix A.

Baselines and plug-and-play protocol. We compare against ANN-to-SNN KD methods (Joint A-SNN (Guo et al., 2023), BKDSNN (Xu et al., 2024), LaSNN (Hong et al., 2025), KDSNN (Zhang et al., 2023)) and BN-guided DFKD baselines (DeepInversion (Yin et al., 2020), CMI (Fang et al.,

2021a), NAYER (Tran et al., 2024)). For each DFKD method $\mathcal{M}$, the plug-and-play variant $\mathcal{M}$ **+ STARS** augments its synthesis objective with RCA and TAR while leaving teacher-side priors unchanged, isolating STARS's contribution.

5.1 Experimental Results

Results on CIFAR-10 & CIFAR-100. Table 1 shows that STARS improves all three DFKD baselines across the evaluated model pairs on CIFAR-10 and CIFAR-100. Gains are most pronounced on NAYER, with STARS yielding 3.5–4.6% on CIFAR-10 and 4.5–6.7% on CIFAR-100. NAYER **+ STARS** reaches 95.73/78.42 under ResNet-19→S-ResNet-19. On DeepInversion and CMI, the improvements are approximately 2–4%. A discussion of the variation across baselines and benchmarks appears in Appendix H. Under ResNet-19→S-ResNet-19, NAYER **+ STARS** reaches 95.73 versus 96.30 for the ANN teacher on CIFAR-10 and 78.42 versus 80.48 on CIFAR-100. These comparisons show that the data-free student approaches the teacher in the evaluated pair; they do not establish equivalence to training with real data across architectures or tasks. NAYER **+ STARS** also exceeds several reported data-available KD entries in Table 1. Those entries use different teachers, architectures, or simulation lengths, so they provide context rather than a controlled comparison of access to real data.

Table 2: Tiny-ImageNet DFKD results under the ResNet-34 → S-ResNet-18 setting. Tiny-ImageNet contains 200 classes with 64×64 images. Δ denotes the Top-1 Acc. improvement of each STARS-enhanced variant over its corresponding DFKD baseline.

Type	Method	Teacher Model	Teacher Acc.	Student Model	T	Top-1 Acc.	Δ
DFKD	CMI	ResNet-34	67.83	S-ResNet-18	4	57.51	–
	CMI **+ STARS**	ResNet-34	67.83	S-ResNet-18	4	**60.84**	**+3.33**
	NAYER	ResNet-34	67.83	S-ResNet-18	4	61.21	–
	NAYER **+ STARS**	ResNet-34	67.83	S-ResNet-18	4	**64.18**	**+2.97**

Results on Tiny-ImageNet. On Tiny-ImageNet, improvements remain consistent, demonstrating generalization beyond CIFAR scales. Specifically, under the same ResNet-34 → S-ResNet-18 setting, **STARS** improves from 57.51 to 60.84 on CMI, and from 61.21 to 64.18 on NAYER, corresponding to gains of 3.33% and 2.97%. Relative to the fixed teacher's accuracy, these recover 32.3% and 44.9% of the respective baseline-to-teacher gaps; the calculation and interpretation are in Appendix H.

Component ablation of STARS. Table 3 isolates the effects of the two components in STARS. On CIFAR-100 with ResNet-19 → S-ResNet-19, we examine whether RCA and TAR are individually useful and whether their combination yields complementary gains. Both components contribute positively, while the full STARS objective yields the strongest result. In particular, adding RCA alone improves the NAYER baseline from 73.95 to 76.18, whereas adding TAR alone gives a smaller gain to 74.36. Combining both modules reaches 78.42: TAR adds an absolute accuracy gain of 2.24% on top of RCA, compared with 0.41% alone, supporting their joint use. A cross-architecture ablation under ResNet-34 → S-ResNet-18 is reported in Appendix Table 10; the layer-selection study appears in Appendix Table 7. Further interpretation is in Appendix I.1.

Table 3: Component ablation of STARS on CIFAR-100 under the ResNet-19 → S-ResNet-19 setting.

Variant	NAYER	+RCA	+TAR	STARS
Acc (%)	73.95	76.18	74.36	**78.42**

Reliability and cost. We repeat the three CIFAR-100 NAYER pairs with paired seeds to assess run-to-run variability. Mean gains are 6.72%, 5.04%, and 4.50% for VGG-16→S-VGG-16, ResNet-34→S-ResNet-18, and ResNet-19→S-ResNet-19, respectively; all nine paired differences are positive. The complete per-seed results and fixed-checkpoint protocol are in Appendix Table 8. In a separate CIFAR-100 VGG-16→S-VGG-16 profile, adding STARS increases total measured epoch time by 0.56–0.98% across the three DFKD bases; Appendix Table 9 reports stage times, GPU memory, and the profiling setup. These figures describe the tested configuration rather than the cost of every architecture or batch size.

Tail-quantile ablation for TAR. Table 4 tests the tail quantile set used in TAR on CIFAR-100 under ResNet-19 $\rightarrow$ S-ResNet-19. Since TAR matches teacher and student soft exceedance probabilities over teacher-anchored thresholds, this experiment tests how much threshold-wise tail coverage is needed for effective ANN-to-SNN transfer. As the number of matched quantiles increases from $M = 2$ to $M = 5$, the accuracy improves monotonically from 76.58 to 77.46, 78.07, and finally 78.42. The trend supports the tested broader probe sets, but does not establish that these quantiles are optimal or that increasing probe count always helps. This ablation changes both probe count and covered range, so their individual effects are not isolated.

Table 4: TAR tail-quantile ablation on CIFAR-100 (ResNet-19 $\rightarrow$ S-ResNet-19). M: number of thresholds.

Tail quantiles	M	Acc. (%)
$(0.90, 0.95)$	2	76.58
$(0.80, 0.90, 0.95)$	3	77.46
$(0.70, 0.80, 0.90, 0.95)$	4	78.07
$(0.60, 0.70, 0.80, 0.90, 0.95)$	5	**78.42**

Qualitative feature view. Figure 2 shows t-SNE (van der Maaten & Hinton, 2008) projections of penultimate-layer SNN features from the VGG-16$\rightarrow$S-VGG-16 NAYER setting. The STARS projections suggest more compact class grouping on CIFAR-10 and CIFAR-100; this visualization is qualitative and does not isolate RCA from TAR. The NAYER-only views are more diffuse in several regions, whereas STARS produces tighter class groupings. A panel-by-panel interpretation appears in Appendix I.

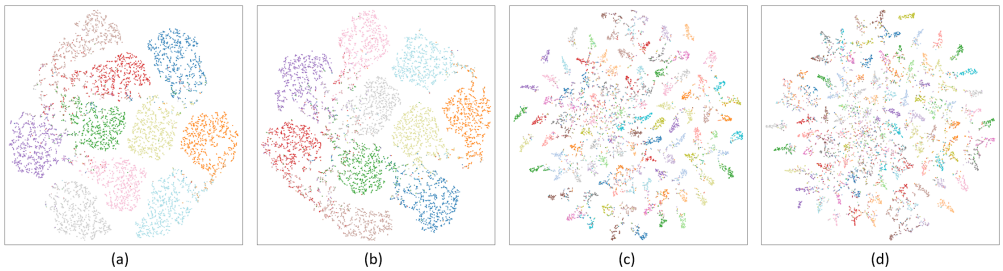

Figure 2: t-SNE of S-VGG-16 penultimate features under NAYER-based DFKD ($T = 4$). **(a,b)** CIFAR-10 with/without STARS; **(c,d)** CIFAR-100 with/without STARS. Colors denote classes.

Simulation length and neuron models. With VGG-16$\rightarrow$S-VGG-16 on CIFAR-100, STARS reaches 72.48, 74.79, and 75.25% at $T = 2, 4, 8$, respectively (Appendix Table 5). Increasing T from two to four adds 2.31%, compared with 0.46% from four to eight. At $T = 4$, IF, LIF, and PLIF yield 74.88, 74.79, and 75.21%, respectively (Appendix Table 6). The observed neuron-model spread is 0.42% at this simulation length. These results cover different temporal resolutions and neuron implementations; without paired baselines at each setting, they do not establish gains for every variant. In a separate ANN–ANN NAYER experiment on CIFAR-100, adding RCA improves ResNet-19 accuracy from 78.26 to 79.11% (+0.85%; Appendix H). This supports RCA's use beyond SNNs but does not evaluate the full RCA+TAR objective with an ANN student.

6 Conclusion

We presented STARS, an add-on for BN-guided ANN-to-SNN data-free knowledge distillation that addresses the mismatch between ANN-oriented BN-guided synthesis and SNN student dynamics. A local first-crossing analysis shows that matching first- and second-order ANN statistics leaves threshold-relevant tail information under-constrained. STARS augments standard synthesis with Relational Consistency Alignment (RCA), which preserves cross-sample relational structure between teacher and student, and Tail-Aware Regularization (TAR), which constrains threshold-relevant tail probabilities through soft exceedance. Experiments on CIFAR-10, CIFAR-100, and Tiny-ImageNet show that STARS consistently improves three DFKD baselines across the evaluated ANN–SNN pairs, with NAYER+STARS gaining up to 4.6% on CIFAR-10 and 6.7% on CIFAR-100. These results support complementing BN matching with relational and tail constraints for the tested SNN-oriented distillation settings. Further mathematical analysis, paired-seed reliability measurements, and discussion of the practical scope and limitations are detailed in the appendix.

AI USE STATEMENT

Generative AI tools were used in a limited capacity to assist with language polishing, literature and method-related research, and minor manuscript organization and formatting. The research problem, methodological design, experiments, analyses, and scientific conclusions were developed and verified by the authors. No experimental data or results were generated by AI. The authors take full responsibility for the final content of this work, including any AI-assisted text.

ETHICS STATEMENT

This work uses existing image-classification benchmarks and does not involve new human-participant data collection. Potential concerns regarding dataset bias, access to pretrained models, and downstream use are discussed in Appendix K.

REPRODUCIBILITY STATEMENT

The main paper describes the STARS objectives and synthesis procedure. Appendix A provides experimental settings, Appendix B details the theoretical analysis, and Appendix D specifies feature correspondence and synthesis implementation. Paired-seed evaluation and resource profiling are described in Appendices F and G, respectively.

REFERENCES

Kuluhan Binici, Shivam Aggarwal, Nam Trung Pham, Karianto Leman, and Tulika Mitra. Robust and resource-efficient data-free knowledge distillation by generative pseudo replay. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 6089–6096, 2022.

Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3514–3522, 2019.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018. doi: 10.1109/MM.2018.112130359.

Shikuang Deng and Shi Gu. Optimal conversion of conventional artificial neural networks to spiking neural networks. *arXiv preprint arXiv:2103.00476*, 2021.

Gongfan Fang, Jie Song, Chengchao Shen, Xinchao Wang, Da Chen, and Mingli Song. Data-free adversarial distillation. *arXiv preprint arXiv:1912.11006*, 2019.

Gongfan Fang, Jie Song, Xinchao Wang, Chengchao Shen, Xingen Wang, and Mingli Song. Contrastive model inversion for data-free knowledge distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 2374–2380, 2021a.

Gongfan Fang, Kanya Mo, Xinchao Wang, Jie Song, Shitao Bei, Haofei Zhang, and Mingli Song. Up to 100x faster data-free knowledge distillation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 6597–6604, 2022.

Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2661–2671, 2021b.

Wulfram Gerstner, Werner M Kistler, Richard Naud, and Liam Paninski. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

Yufei Guo, Xinyi Tong, Yuanpei Chen, Liwen Zhang, Xiaode Liu, Zhe Ma, and Xuhui Huang. 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*, pp. 326–335, 2022.

Yufei Guo, Weihang Peng, Yuanpei Chen, Liwen Zhang, Xiaode Liu, Xuhui Huang, and Zhe Ma. Joint A-SNN: Joint training of artificial and spiking neural networks via self-distillation and weight factorization. *Pattern Recognition*, 142:109639, 2023.

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.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Di Hong, Yu Qi, and Yueming Wang. LaSNN: Layer-wise ANN-to-SNN distillation for effective and efficient training in deep spiking neural networks. *Neurocomputing*, 656:131351, 2025. doi: 10.1016/j.neucom.2025.131351.

Zixuan Hu, Li Shen, Zhenyi Wang, Tongliang Liu, Chun Yuan, and Dacheng Tao. Architecture, dataset and model-scale agnostic data-free meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7736–7745, 2023a.

Zixuan Hu, Li Shen, Zhenyi Wang, Baoyuan Wu, Chun Yuan, and Dacheng Tao. Learning to learn from apis: Black-box data-free meta-learning. In *International Conference on Machine Learning*, pp. 13610–13627. PMLR, 2023b.

Zixuan Hu, Yongxian Wei, Li Shen, Zhenyi Wang, Baoyuan Wu, Chun Yuan, and Dacheng Tao. Task-distributionally robust data-free meta-learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(1):436–447, 2026. doi: 10.1109/TPAMI.2025.3609625.

Youngeun Kim, Yeshwanth Venkatesha, and Priyadarshini Panda. PrivateSNN: Privacy-preserving spiking neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36 (1):1192–1200, June 2022. ISSN 2159-5399. doi: 10.1609/aaai.v36i1.20005. URL `http://dx.doi.org/10.1609/aaai.v36i1.20005`.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. Technical report, Stanford University, 2015. URL `https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf`. CS231n course project report.

Jingru Li, Sheng Zhou, Liangcheng Li, Haishuai Wang, Jiajun Bu, and Zhi Yu. AdaDFKD: Exploring adaptive inter-sample relationship in data-free knowledge distillation. *Neural Networks*, 177: 106386, September 2024. ISSN 0893-6080. doi: 10.1016/j.neunet.2024.106386. URL `http://dx.doi.org/10.1016/j.neunet.2024.106386`.

Yuhang Li, Yufei Guo, Shanghang Zhang, Shikuang Deng, Yongqing Hai, and Shi Gu. Differentiable spike: Rethinking gradient-descent for training spiking neural networks. *Advances in neural information processing systems*, 34:23426–23439, 2021.

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

Paul Micaelli and Amos J Storkey. Zero-shot knowledge transfer via adversarial belief matching. *Advances in Neural Information Processing Systems*, 32, 2019.

Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In *International conference on machine learning*, pp. 4743–4751. PMLR, 2019.

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. doi: 10.1109/MSP.2019.2931595.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3967–3976, 2019.

Yuanbin Qian, Shuhan Ye, Chong Wang, Xiaojie Cai, Jiangbo Qian, and Jiafei Wu. Ucf-crime-dvs: A novel event-based dataset for video anomaly detection with spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 6577–6585, 2025.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. URL https://arxiv.org/abs/1412.6550.

Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784):607–617, 2019.

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

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

Minh-Tuan Tran, Trung Le, Xuan-May Le, Mehrtash Harandi, Quan Hung Tran, and Dinh Phung. Nayer: Noisy layer data generation for efficient and effective data-free knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23860–23869, 2024.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL https://jmlr.org/papers/v9/vandermaaten08a.html.

Wenjie Wei, Malu Zhang, Hong Qu, Ammar Belatreche, Jian Zhang, and Hong Chen. Temporal-coded spiking neural networks with dynamic firing threshold: Learning with event-driven backpropagation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10552–10562, 2023.

Yongxian Wei, Zixuan Hu, Li Shen, Zhenyi Wang, Yu Li, Chun Yuan, and Dacheng Tao. Task groupings regularization: Data-free meta-learning with heterogeneous pre-trained models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 52573–52587, 2024a.

Yongxian Wei, Zixuan Hu, Zhenyi Wang, Li Shen, Chun Yuan, and Dacheng Tao. Free: Faster and better data-free meta-learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 23273–23282, 2024b.

Jibin Wu, Chenglin Xu, Xiao Han, Daquan Zhou, Malu Zhang, Haizhou Li, and Kay Chen Tan. Progressive tandem learning for pattern recognition with deep spiking neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7824–7840, 2022. doi: 10.1109/TPAMI.2021.3114196.

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.

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.

Qu Yang, Jibin Wu, Malu Zhang, Yansong Chua, Xinchao Wang, and Haizhou Li. Training spiking neural networks with local tandem learning. *Advances in Neural Information Processing Systems*, 35:12662–12676, 2022.

Shu Yang, Chengting Yu, Lei Liu, Hanzhi Ma, Aili Wang, and Erping Li. Efficient ANN-guided distillation: Aligning rate-based features of spiking neural networks through hybrid block-wise replacement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10025–10035, 2025.

Xingting Yao, Fanrong Li, Zitao Mo, and Jian Cheng. Glif: A unified gated leaky integrate-and-fire neuron for spiking neural networks. *Advances in Neural Information Processing Systems*, 35: 32160–32171, 2022.

Shuhan Ye, Yuanbin Qian, Chong Wang, Sunqi Lin, Jiazhen Xu, Jiangbo Qian, and Yuqi Li. Cross knowledge distillation between artificial and spiking neural networks. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2025.

Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8715–8724, 2020.

Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. *Advances in Neural Information Processing Systems*, 32, 2019.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8806–8816, 2025.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Fengzhao Zhang, Chengting Yu, Hanzhi Ma, Zheming Gu, and Er-ping Li. Knowledge distillation for spiking neural network. In *2023 5th International Conference on Robotics, Intelligent Control and Artificial Intelligence (RICAI)*, pp. 1015–1020. IEEE, 2023.

Hanle Zheng, Yujie Wu, Lei Deng, Yifan Hu, and Guoqi Li. Going deeper with directly-trained larger spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11062–11070, 2021.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng YAN, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *The Eleventh International Conference on Learning Representations*, 2023.

APPENDIX

A DETAILED EXPERIMENTAL SETTINGS

All experiments are conducted on NVIDIA RTX Pro 6000 GPUs. This appendix summarizes the detailed experimental settings used in our DFKD experiments. Unless otherwise specified, STARS is added on top of the base synthesis objective with $\lambda_{\mathrm{rca}} = \lambda_{\mathrm{tar}} = 1$, `threshold_mode=quantile`, and `threshold_values=`$\{$`0.6,0.7,0.8,0.9,0.95`$\}$.

DeepInversion. For DeepInversion on CIFAR-10 and CIFAR-100, we evaluate across three ANN teacher / SNN student pairs: VGG-16 $\rightarrow$ S-VGG-16, ResNet-34 $\rightarrow$ S-ResNet-18, and ResNet-19 $\rightarrow$ S-ResNet-19, all with $T = 4$ simulation steps. The following training hyperparameters are shared across all three pairs. The student is trained for 200 epochs with no warmup. We use 400 KD steps, 400 EP steps, and 1000 generator steps, with student batch size 256 and synthesis batch size 256. The student learning rate is 0.1 and the generator learning rate is 0.1. The DeepInversion loss weights are set to adversarial loss 1.0, BN regularization 10.0, and one-hot loss 1.0, and the KD temperature is 20. The random seed is fixed to 0. On top of this baseline, STARS applies RCA and TAR with `threshold_values=`$\{$`0.6,0.7,0.8,0.9,0.95`$\}$, shared across all three pairs.

CMI. For CMI on CIFAR-10 and CIFAR-100, we likewise evaluate across the same three ANN teacher / SNN student pairs: VGG-16 $\rightarrow$ S-VGG-16, ResNet-34 $\rightarrow$ S-ResNet-18, and ResNet-19 $\rightarrow$ S-ResNet-19, all with $T = 4$. The following training hyperparameters are shared across all three pairs. The student is trained for 200 epochs without warmup, and the inversion / distillation schedule uses 400 KD steps, 400 EP steps, and 200 generator steps. Both the student batch size and the synthesis batch size are 256. The student learning rate is 0.1 and the generator learning rate is 10^{-3}. The CMI objective uses loss weights 0.5 for adversarial loss, 1.0 for BN regularization, 1.0 for one-hot loss, 0.8 for contrastive regularization, and 0.2 for teacher-side contrastive temperature scaling, with KD temperature 20 and seed 0. The STARS configuration is identical to that used in DeepInversion above and is likewise shared across all three pairs. For Tiny-ImageNet, CMI also follows this same training and synthesis configuration under the ResNet-34 $\rightarrow$ S-ResNet-18 setting with $T = 4$.

NAYER. For NAYER on CIFAR-10 and CIFAR-100, we evaluate across three ANN teacher / SNN student pairs: VGG-16 $\rightarrow$ S-VGG-16, ResNet-34 $\rightarrow$ S-ResNet-18, and ResNet-19 $\rightarrow$ S-ResNet-19, all with $T = 4$ simulation steps. The student is trained for 320 epochs with 20 warmup epochs. The student learning rate is 0.2 and the generator learning rate is 4×10^{-3}. The NAYER loss weights are adversarial loss 1.33, BN regularization 20.0, and one-hot loss 0.75. We use 1000 KD steps, 1000 EP steps, and 75 generator steps per synthesis round, with generator life 10, generator loops 4, and GWP loops 10. The random seed is fixed to 0.

For Tiny-ImageNet, we use the ResNet-34 $\rightarrow$ S-ResNet-18 setting with $T = 4$ and input resolution 64×64. The teacher top-1 accuracy is 67.83. The student is trained for 320 epochs with 20 warmup epochs. We use 1000 KD steps, 1000 EP steps, and 60 generator steps, together with generator life 10, generator loops 4, and GWP loops 20. The student batch size is 256 and the synthesis batch size is 200. The student learning rate is 0.1 and the generator learning rate is 4×10^{-3}. The NAYER loss weights are adversarial loss 1.33, BN regularization 10.0, and one-hot loss 0.5, with seed 0.

B Scope of the BN-Moment Ambiguity Analysis

This appendix expands the simplified single-layer direct-coding analysis in Sec. 3. We retain the distinction between the distribution of a local constant input and the time-varying currents generated inside a deep network. The purpose is to show, step by step, what BN moments leave unspecified and how this motivates additional tail constraints, without identifying a pre-reset response with the firing rate after reset.

B.1 From membrane integration to a first-crossing event

Set $V_{\mathrm{r}} = 0$, $H_{\mathrm{pre}}[0] = 0$, $X[t] = a$, $\tau > 1$, and $V_{\mathrm{th}} > 0$. Before the first spike, $S[t-1] = 0$ and $V[t-1] = H_{\mathrm{pre}}[t-1]$. Substituting these relations into Eq. (1) gives

$$H_{\mathrm{pre}}[t] = H_{\mathrm{pre}}[t-1] + \frac{a - H_{\mathrm{pre}}[t-1]}{\tau} = \alpha H_{\mathrm{pre}}[t-1] + \frac{a}{\tau}, \qquad \alpha = 1 - \frac{1}{\tau} \in (0,1). \tag{18}$$

Iterating the recursion from zero yields the geometric sum

$$H_{\mathrm{pre}}[t] = \frac{a}{\tau} \sum_{k=0}^{t-1} \alpha^k = \frac{a}{\tau} \frac{1-\alpha^t}{1-\alpha} = (1-\alpha^t)a = \beta_t a. \tag{19}$$

Here β_t quantifies the fraction of the constant input accumulated by time t. For positive a, this unreset trajectory increases with time; for nonpositive a, it cannot reach the positive threshold. Consequently, in both cases the event of a first crossing by t is exactly $\{a \geq V_{\mathrm{th}}/\beta_t\}$. The unreset trajectory may be continued algebraically after a crossing to evaluate this event, but that continuation is not the physical membrane trajectory after a spike.

Define $I_t(a) = \mathbf{1}\{T_{\mathrm{first}} \leq t\}$ and $\theta_t = V_{\mathrm{th}}/\beta_t$. The survival probability uses an inclusive threshold:

$$C_t(P) = \mathbb{E}_{a\sim P}[I_t(a)] = \mathbb{P}_{a\sim P}(a \geq \theta_t) = 1 - F_P(\theta_t^-). \tag{20}$$

The left limit $F_P(\theta_t^-) = \mathbb{P}(a < \theta_t)$ matters when the activation distribution has an atom at the threshold. For a continuous distribution, it equals $F_P(\theta_t)$.

B.2 WHY TWO MOMENTS DO NOT DETERMINE THE RESPONSE

Step 1: The relevant event is a tail functional. Equation (20) expresses the probability of a first crossing as probability mass above an effective threshold. One can also summarize the cumulative crossing profile over a horizon by

$$J_T(P) := \frac{1}{T}\sum_{t=1}^{T} C_t(P) = \mathbb{E}_{a\sim P}\left[\frac{1}{T}\sum_{t=1}^{T} I_t(a)\right] = \frac{1}{T}\sum_{t=1}^{T} \bar{F}_P(\theta_t). \tag{21}$$

This is a cumulative first-crossing score, not a firing rate. If the first spike occurs at time $k \le T$, its contribution is $(T-k+1)/T$, irrespective of the number of subsequent spikes. Earlier first crossings contribute more, while inputs that never cross within the horizon contribute zero.

Step 2: BN matching constrains only the first two moments. At the local layer under consideration, the BN term compares $\mu_A(P) = \mathbb{E}_P[a]$ and $\sigma_A^2(P) = \mathrm{Var}_P(a)$ with the stored teacher statistics. These two quantities do not specify the entire activation distribution. In particular, the BN term alone does not directly constrain $\bar{F}_P(\theta_t)$, cross-sample relations, or higher-order distributional shape. Other parts of a complete DFKD objective can impose additional constraints; the insufficiency claim concerns the moments alone.

Step 3: Construct equal moments with unequal tails. The local direct-coding analysis in Sec. 3 concerns the membrane trajectory before its first reset. Fix a time t, set $\theta = \theta_t > 0$, and let $d > 0$. Consider activation distributions of the form $a = \theta + dX$ with

$$P: \quad \mathbb{P}(X=-1) = \mathbb{P}(X=1) = \tfrac{1}{2}, \qquad Q: \quad \mathbb{P}(X=-1/\sqrt{3}) = \tfrac{3}{4}, \quad \mathbb{P}(X=\sqrt{3}) = \tfrac{1}{4}. \tag{22}$$

For both distributions, $\mathbb{E}[a] = \theta$ and $\mathrm{Var}(a) = d^2$. Yet their probabilities of crossing the pre-reset effective threshold by time t are

$$C_t(P) = \mathbb{P}_P(a \ge \theta) = \tfrac{1}{2}, \qquad C_t(Q) = \mathbb{P}_Q(a \ge \theta) = \tfrac{1}{4}. \tag{23}$$

Thus equal first and second moments do not identify even one SNN-relevant first-crossing probability. The example is an insufficiency result for BN moments alone, rather than a claim that every BN-matched synthetic batch fails: other teacher priors and the joint constraints across layers can reduce the ambiguity in practice.

The moment calculation is explicit: under P, $\mathbb{E}[X] = 0$ and $\mathbb{E}[X^2] = 1$; under Q, $\mathbb{E}[X] = -3/(4\sqrt{3}) + \sqrt{3}/4 = 0$ and $\mathbb{E}[X^2] = (3/4)(1/3) + (1/4)3 = 1$. The shift and scale therefore give the same mean θ and variance d^2 while retaining different upper-tail masses. Unlike adding an arbitrary point mass to a reference distribution, this construction preserves both moments exactly.

Step 4: Consequence for the crossing profile. For equal-moment distributions, $C_t(P) - C_t(Q) = \bar{F}_P(\theta_t) - \bar{F}_Q(\theta_t)$. The example proves that the profile can differ at a selected time. For the horizon score,

$$J_T(P) - J_T(Q) = \frac{1}{T}\sum_{t=1}^{T}\big[\bar{F}_P(\theta_t) - \bar{F}_Q(\theta_t)\big]. \tag{24}$$

A difference at one threshold alone need not make this sum nonzero, because differences at other thresholds can cancel. A common sign across thresholds, with at least one strict inequality, is sufficient, as discussed next.

B.3 OPERATING-POINT INTERPRETATION AND SCOPE

Lower or higher first-crossing propensity. If $\bar{F}_{P_s}(\theta_t) \le \bar{F}_{P_r}(\theta_t)$ at every effective threshold, with strict inequality for at least one, then $J_T(P_s) < J_T(P_r)$. Reversing the inequalities gives $J_T(P_s) > J_T(P_r)$. These conditions concern the tested thresholds, not stochastic dominance over the entire real line. They describe a lower or higher cumulative first-crossing propensity relative to a reference distribution: less or more mass reaches the threshold within the evaluated horizon. They do not, by themselves, establish sustained under-firing, over-firing, or saturation once resets and time-varying currents are included.

What the argument does and does not establish. The event $\{T_{\text{first}} \le t\}$ is cumulative. Its probability $C_t(P)$ is not the probability of an output spike *at* time t, nor is an average of the $C_t(P)$ values the firing rate of a reset-aware neuron. Once the first spike occurs, the post-reset membrane trajectory must be followed explicitly to count later spikes. The counterexample remains valid because reset takes place only after the first crossing. At deeper layers, upstream spiking changes the input currents, so the constant-current closed form is not a deep-network theorem. STARS instead evaluates the teacher pre-ReLU and student pre-LIF features produced by the actual networks; the benefit of applying its constraints across these layers is empirical.

Interpretation for TAR. A smaller $C_t(P)$ means that a first crossing is less likely to have occurred by t under the simplified neuron model; a larger value means it is more likely. Neither statement alone identifies the later firing rate or a network-wide operating regime. TAR uses teacher-derived, layer-specific feature quantiles as smooth tail probes. These quantiles need not equal a neuron's physical membrane threshold or be an optimal proxy for it. Their role is to constrain tail information that the first two moments leave unspecified; the choice of pooled layer-wise rather than channel-wise quantiles trades granularity for more observations per probe in a limited synthetic batch.

Implication for heterogeneous distillation. Matching BN moments defines an equivalence class with respect to a particular teacher-side statistic, but that equivalence need not hold for a student's threshold-sensitive response. TAR adds finite tail probes to reduce this ambiguity, while RCA constrains cross-sample geometry. Neither a finite threshold set nor moment matching identifies the full distribution. The analysis motivates the additional information being matched; improvements in classification, the benefit of combining RCA and TAR, and behavior in deep SNNs are established by the reported experiments rather than by the local calculation alone.

B.4 What finite soft tail probes control

The local counterexample motivates tail information but does not uniquely specify TAR. To state its direct mathematical effect, fix a realized batch and layer. Let X_T and X_S be uniform draws from the corresponding pooled feature entries, in the coordinates used by TAR. Treat the realized teacher-derived probes $\theta_1, \dots, \theta_M$ as fixed. Write

$$s_r(u) = \mathbb{P}(X_r \ge u), \qquad p_r(u) = \mathbb{E}[\sigma((X_r - u)/\delta)], \qquad r \in \{T, S\}. \tag{25}$$

Thus Eq. (15) is exactly a sigmoid-smoothed survival functional of the pooled empirical distribution. Equivalently, $p_r(u) = \mathbb{P}(X_r + \delta L \ge u)$ for an independent standard logistic variable L. This is an interpretation of the objective, not injected noise in the implementation.

Proposition 1 (Hard-tail discrepancy at a selected probe)**.** *Let $\ell = M^{-1}\sum_{j=1}^{M}(p_S(\theta_j) - p_T(\theta_j))^2$ be the layer TAR loss. For any selected probe θ_m and any $\varepsilon > 0$,*

$$|s_S(\theta_m) - s_T(\theta_m)| \le \sqrt{M\ell} + \sum_{r\in\{T,S\}} \mathbb{P}(|X_r - \theta_m| \le \varepsilon) + 2e^{-\varepsilon/\delta}. \tag{26}$$

Proof. Outside $|x - u| \le \varepsilon$, the difference between $\mathbf{1}\{x \ge u\}$ and $\sigma((x-u)/\delta)$ is at most $e^{-\varepsilon/\delta}$; inside, it is at most one. Taking expectations bounds $|s_r(u) - p_r(u)|$ by the near-threshold probability plus $e^{-\varepsilon/\delta}$. Apply the triangle inequality for teacher and student at $u = \theta_m$, and use $|p_S(\theta_m) - p_T(\theta_m)| \le \sqrt{M\ell}$. □

The bound separates the optimized soft discrepancy from smoothing error and feature mass close to a probe. A small TAR loss alone does not make all three terms small; the bound may be loose at the chosen softness. It applies to the empirical pooled features at selected probes, not to unseen data without further assumptions. For atom-free distributions, letting $\delta \to 0$ recovers the hard survival functional; an atom exactly at the probe instead receives half weight in the sigmoid limit.

This conditional result does not identify tail gaps at unprobed thresholds, justify the particular quantile set as optimal, or recover information discarded by spatial pooling and temporal averaging. Only when the variables and probes are the local currents and effective thresholds of Eq. (8) can the hard tails be identified with first-crossing probabilities. Those identifications are not assumed for deep pooled TAR features. The connection is therefore a precise feature-level surrogate interpretation, not a theorem guaranteeing matching deep SNN firing rates or improved classification.

C Additional Robustness Studies

Impact of time steps T. Appendix Table 5 reports STARS accuracy at $T = 2, 4, 8$. It varies the student simulation length but does not include a paired NAYER baseline for each T.

Impact of neuron models. Appendix Table 6 reports STARS accuracy with IF, LIF, and PLIF neurons. These results show the method can be evaluated with each neuron choice; they do not establish a gain over a separately tuned NAYER baseline for each choice.

Table 5: Impact of simulation time steps T on CIFAR-100 under the VGG-16 $\rightarrow$ S-VGG-16 setting. We report Top-1 Acc. (%) of NAYER **+ STARS** with different SNN student simulation steps.

Teacher	Student	T	Top-1 Acc. (%)
VGG-16	S-VGG-16	2	72.48
VGG-16	S-VGG-16	4	74.79
VGG-16	S-VGG-16	8	75.25

Table 6: Impact of different spiking neuron models on CIFAR-100 under the VGG-16 $\rightarrow$ S-VGG-16 setting with $T = 4$. We report Top-1 Acc. (%) of NAYER **+ STARS** when replacing the student neuron model.

Teacher	Student	Neuron model	Top-1 Acc. (%)
VGG-16	S-VGG-16	IF	74.88
VGG-16	S-VGG-16	LIF	74.79
VGG-16	S-VGG-16	PLIF	75.21

D Stage Correspondence and Student-Aware Synthesis

We apply RCA and TAR once at the end of each semantic stage, rather than after every BN layer or residual block. For each convolutional or residual stage, the selected teacher feature is the output of its final BN before ReLU, paired with the student feature before the corresponding LIF. We also select the final global-pooled representation, but exclude the input stem and classifier logits. Spatial feature maps are globally averaged; student features are additionally averaged over the T simulation steps before computing RCA and TAR.

For VGG-16 $\rightarrow$ S-VGG-16, the selected set contains the final features of stages 1–5 and the global-pooled feature ($|\mathcal{S}| = 6$). For ResNet-34 $\rightarrow$ S-ResNet-18, we align the final-block features of stages `layer1`–`layer4` despite the different numbers of residual blocks, and add the global-pooled feature ($|\mathcal{S}| = 5$). For ResNet-19 $\rightarrow$ S-ResNet-19, we use the final-block features of `layer1`–`layer3` and the global-pooled feature ($|\mathcal{S}| = 4$). In each case, $w_l = 1/|\mathcal{S}|$ for all selected positions.

During each synthesis inner loop, the ANN teacher and current SNN student weights are fixed. The complete base DFKD objective, including teacher class confidence, BN statistics, and its original regularizers, remains active. RCA matches student relations to teacher relations on the same batch, and TAR uses teacher-derived normalization and layer-specific quantiles. Gradients pass through the fixed networks to update the synthetic batch; the student weights are updated later in the distillation stage. Thus surrogate batches are optimized for the specified teacher–student transfer, without any claim that a frozen batch would transfer to arbitrary unseen students.

E Layer-Selection Ablation

Table 7 evaluates a cumulative selection on CIFAR-100 with NAYER and ResNet-19 $\rightarrow$ S-ResNet-19. Positions 1–3 are the final pre-activation features of `layer1`–`layer3`; position 4 is the global-pooled feature. The all-position configuration obtains the highest observed accuracy. This ablation tests the chosen cumulative sequence; it does not isolate every possible layer subset.

Table 7: Cumulative selected positions for RCA and TAR on CIFAR-100. "None" is the NAYER baseline.

Selected positions	None	1	1,2	1,2,3	1,2,3,4
Top-1 accuracy (%)	73.95	76.88	77.51	78.03	78.42

F Paired Multi-Seed Evaluation

For each CIFAR-100 NAYER comparison in Table 8, seeds 0, 2026, and 42 use the same teacher checkpoint, architecture, synthesis budget, training schedule, and non-seed hyperparameters within the baseline/STARS pair. We report top-1 accuracy and the sample standard deviation across three runs. The seed-0 values match the main results. All three paired seed differences are positive for each model pair; these data quantify run-to-run variability within the tested configurations, rather than establishing a population-wide performance guarantee.

Table 8: CIFAR-100 paired random-seed results (top-1 accuracy, %). The final column is mean $\pm$ sample standard deviation.

Method	Teacher $\rightarrow$ student	Seed 0	Seed 2026	Seed 42	Mean $\pm$ SD
NAYER	VGG-16 $\rightarrow$ S-VGG-16	68.09	67.74	68.26	68.03 ± 0.27
NAYER+STARS	VGG-16 $\rightarrow$ S-VGG-16	74.79	74.55	74.92	74.75 ± 0.19
NAYER	ResNet-34 $\rightarrow$ S-ResNet-18	71.89	72.14	71.71	71.91 ± 0.22
NAYER+STARS	ResNet-34 $\rightarrow$ S-ResNet-18	76.97	77.10	76.79	76.95 ± 0.16
NAYER	ResNet-19 $\rightarrow$ S-ResNet-19	73.95	73.68	74.09	73.91 ± 0.21
NAYER+STARS	ResNet-19 $\rightarrow$ S-ResNet-19	78.42	78.21	78.58	78.40 ± 0.19

The corresponding differences between means are 6.72%, 5.04%, and 4.50%. Since these are paired seeds, their differences should be read alongside the per-seed values rather than inferred from the marginal standard deviations alone.

G Training-Time Resource Profile

Table 9 profiles multiple complete CIFAR-100 VGG-16 $\rightarrow$ S-VGG-16 epochs on one NVIDIA RTX Pro 6000, using $T = 4$, FP32, and student and synthesis batch sizes of 256. Stage-boundary CUDA synchronization is used for timing. The values are average seconds per epoch; peak GPU memory is the maximum allocated memory observed across the profiled epochs. DeepInversion uses 1000 synthesis and 400 KD steps, CMI uses 200 and 400, and NAYER uses 75 synthesis steps across four generator loops and 1000 KD steps. We report these measurements to compare training costs under the specified architecture, hardware, and optimization settings; they should not be interpreted as universal runtime estimates across different configurations.

Table 9: Measured resource use on CIFAR-100 VGG-16 $\rightarrow$ S-VGG-16. Relative overhead is computed against the corresponding base method's total epoch time. DeepInv. denotes DeepInversion; Synth. is synthesis time.

Method	Synth. (s)	Student (s)	Valid. (s)	Total (s)	Peak (GiB)	Overhead	Top-1 (%)
DeepInv.	62.34	85.14	1.23	148.71	7.04	–	35.68
DeepInv.+STARS	63.63	85.31	1.22	150.16	7.19	+0.98%	39.72
CMI	26.61	122.12	1.21	149.94	6.23	–	55.91
CMI+STARS	27.20	122.36	1.22	150.78	6.40	+0.56%	58.96
NAYER	37.50	127.85	1.20	166.55	7.46	–	68.09
NAYER+STARS	38.32	128.07	1.20	167.59	7.65	+0.62%	74.79

The corresponding synthesis-stage increases are 2.07%, 2.22%, and 2.19%; total epoch increases are 0.98%, 0.56%, and 0.62%. RCA computes a batch relation matrix with approximate $O(\sum_{l \in \mathcal{S}} B^2 C_l)$

work, while extracting and storing student activations depends on T and the selected layers. RCA and TAR are absent from the deployed student's inference graph, so they add no inference modules or simulation steps. We did not measure inference energy; an unchanged graph does not by itself establish identical energy consumption.

H Additional Transfer Results and Interpretation

Cross-architecture component ablation. Table 10 repeats the component ablation on CIFAR-100 with a ResNet-34 ANN teacher and an S-ResNet-18 student. The teacher, student, base NAYER objective, and training schedule are held fixed within the comparison. Both components improve the base method, and their combination has the largest observed gain. The larger standalone RCA gain than TAR gain is an empirical observation in this pair; the experiment alone does not establish the cause of that difference.

Table 10: CIFAR-100 cross-architecture component ablation. Gains are absolute accuracy differences (%) over NAYER.

Variant	Top-1 accuracy (%)	Gain (%)
NAYER	71.89	–
NAYER+RCA	74.78	+2.89
NAYER+TAR	72.55	+0.66
NAYER+RCA+TAR (STARS)	76.97	+5.08

Tiny-ImageNet teacher-gap interpretation. For the same ResNet-34 teacher (67.83% top-1), CMI improves from 57.51% to 60.84% and NAYER from 61.21% to 64.18%. Defining recovered gap as $(A_{\mathrm{STARS}} - A_{\mathrm{base}})/(A_{\mathrm{teacher}} - A_{\mathrm{base}})$ gives $3.33/10.32 = 32.3\%$ for CMI and $2.97/6.62 = 44.9\%$ for NAYER. This normalizes each gain by its own starting margin to the fixed teacher; neither student reaches teacher accuracy.

Variation across baselines and benchmarks. The gains are not monotonic in class count: synthesis difficulty, baseline priors, and the remaining teacher–student margin all vary together. DeepInversion relies strongly on teacher BN and confidence priors, CMI already constrains some relations through contrastive synthesis, and NAYER provides a stronger generative basis. These factors offer plausible interpretations of the observed gain pattern, but the present experiments do not isolate their individual causal effects.

RCA beyond ANN-to-SNN transfer. In a separate CIFAR-100 ResNet-19 ANN-teacher/ANN-student NAYER experiment, adding RCA alone changes top-1 accuracy from 78.26% to 79.11% (+0.85%). This is evidence that RCA is usable in an ANN–ANN setting; it does not test the full RCA+TAR framework or replace the main ANN–SNN evidence.

I Extended Method and Experimental Discussion

The synthesis and distillation stages remain separate. With the optimized batch fixed, the student's distillation loss is the original KD objective evaluated on that batch:

$$\mathcal{L}_{\mathrm{KD}}(\mathcal{B}^\star) = \frac{1}{|\mathcal{B}^\star|} \sum_{x \in \mathcal{B}^\star} \mathrm{KL}(\sigma(f_T(x)/\gamma) \mid \sigma(f_S(x)/\gamma)) \,. \tag{27}$$

Here σ is softmax and γ is the KD temperature, as in Eq. (2); the sigmoid in TAR serves a separate role as a smooth threshold function.

Pre-activation features and teacher-anchored tail probes. The selected ANN features precede ReLU, and the corresponding SNN features precede LIF thresholding. Post-spike outputs are binary and reflect thresholding and reset; pre-activation features provide a continuous space in which to construct the two penalties. Spatial pooling and temporal averaging produce the batch-wise channel

representations in Eq. (11). RCA row-normalizes those representations and compares off-diagonal cosine relations. TAR instead uses their feature values with teacher-derived thresholds and, when used, teacher-derived normalization. These operations retain different aspects of the representation: relative sample geometry for RCA and pooled threshold-wise tail mass for TAR. Temporal averaging does not preserve every spike time or the full membrane trajectory, so TAR should not be interpreted as matching a time-resolved firing process.

Comparison with the teacher and data-available KD. For ResNet-19 $\rightarrow$ S-ResNet-19, NAYER+STARS obtains 95.73% on CIFAR-10 against the teacher's 96.30%, leaving a 0.57% accuracy gap; on CIFAR-100 it obtains 78.42% against 80.48%, leaving 2.06%. Several data-available KD entries in Table 1 have lower reported accuracies. However, teacher checkpoints, architectures, and simulation lengths vary across those entries. The table therefore demonstrates strong results in the evaluated data-free setting, without establishing that synthetic data can generally substitute for real-data training under matched conditions. On Tiny-ImageNet, the gains of 3.33% and 2.97% extend the observation to the tested 200-class, 64×64 benchmark; they do not isolate which aspect of dataset scale drives the gain.

Reading the t-SNE visualization. Figure 2 compares S-VGG-16 penultimate-layer features under NAYER-based DFKD with a VGG-16 teacher and $T = 4$. Panels (a) and (b) show CIFAR-10 with STARS and with NAYER only, respectively; panels (c) and (d) give the corresponding CIFAR-100 views. Colors identify classes. The STARS projections display visually tighter class groupings and less entanglement in several regions, consistent with the accuracy improvements. This is a qualitative view of the combined framework. A two-dimensional t-SNE projection does not prove linear separability in the original feature space, quantify every class margin, or attribute the change separately to RCA or TAR.

Simulation length and neuron choice. On CIFAR-100, NAYER+STARS with VGG-16 $\rightarrow$ S-VGG-16 reaches 72.48%, 74.79%, and 75.25% at $T = 2, 4, 8$, respectively (Table 5). The change from two to four steps is 2.31%, and from four to eight it is 0.46% in these runs. At $T = 4$, the reported accuracies are 74.88% with IF, 74.79% with LIF, and 75.21% with PLIF (Table 6). We observe higher accuracy with longer simulation horizons and similar accuracies across the tested neuron models. Without paired base-method results at each setting, they do not establish an improvement over NAYER for every simulation length or neuron choice.

Prospects and deployment boundary. The data-free setting is relevant when a pretrained teacher remains accessible but its original training data cannot be stored or transmitted. Potential applications include on-device sensing and edge robotics, provided the teacher, data permissions, and target task are suitable. These are possible uses rather than evaluated deployments. Extending the framework to LayerNorm-based transformer SNNs requires a compatible synthesis prior in place of the present BN-guided setting. Similarly, structured-output tasks require their own objectives and evaluation. Access to a teacher can itself expose information about training data; avoiding direct use of the original dataset is not a privacy guarantee. The present experiments do not measure hardware energy consumption or validate a deployment in a medical or robotic system.

I.1 Joint interpretation and practical scope

Interpretation of the two constraints. The analysis in Sec. 3 establishes that BN moments alone do not identify a direct-coded neuron's first threshold-crossing probability. It does not derive the firing rate of a deep, reset-aware SNN. TAR uses teacher-derived quantiles as practical tail probes, while RCA constrains cross-sample relations absent from BN moments. Both operate on features of the actual networks; their deep-network benefit is supported by experiments rather than implied by the local counterexample. Appendix B gives the construction and its boundary.

Evidence for joint use. The full objective is best in both component ablations: it improves NAYER by 4.47% for ResNet-19$\rightarrow$S-ResNet-19 (Table 3) and 5.08% for ResNet-34$\rightarrow$S-ResNet-18 (Appendix Table 10). RCA alone contributes 2.23% and 2.89% in these pairs, versus 0.41% and 0.66% for TAR. Adding TAR on top of RCA contributes 2.24% and 2.19%, respectively. Thus its standalone gain understates its conditional contribution in these configurations. These descriptive differences support joint use, but do not establish a statistically tested interaction or prove that TAR benefits only spiking

students. The additional ANN–ANN experiment tests RCA alone; a full-STARS ANN–ANN control is not available. Appendix Table 7 reports the cumulative stage-selection experiment, which does not test every subset or prove that any one layer is universally necessary.

Student dependence and practical scope. Synthesis uses the current student but retains teacher priors and teacher-derived alignment targets; the student is fixed within each synthesis inner loop (Appendix D). This does not promise portability of a frozen synthetic dataset to arbitrary students. DeepInversion and CMI have no student warm-up, whereas NAYER uses 20 epochs; positive gains across these schedules and paired seeds reduce concern about one favorable transient state but do not rule out student-specific effects elsewhere. RCA scales quadratically with batch size, and inference energy was not measured. The ANN–ANN RCA result in Appendix H tests an adjacent setting; the main claim remains BN-guided ANN–SNN image-classification DFKD.

J LIMITATIONS

Our analysis establishes a local pre-reset counterexample, not a closed-form result for reset-aware deep SNN firing rates. The method is evaluated mainly for BN-guided image-classification DFKD with convolutional ANN teachers; conventional LayerNorm-based Transformers need a compatible data-free synthesis prior, and structured-output tasks need task-specific objectives. Transfer to native event streams is not evaluated here. STARS also conditions synthesis on the current target student rather than producing a dataset guaranteed to transfer to arbitrary students. RCA's batch relation matrix has quadratic dependence on batch size. The multi-seed study covers three NAYER CIFAR-100 pairs, and the resource profile covers one architecture and hardware setting; inference energy was not measured.

K BROADER IMPACT

This work studies distillation to spiking students without using the original training set during transfer. That setting may be useful when data cannot be retained or transmitted, although access to a pretrained teacher can itself raise privacy or licensing questions. STARS does not change the deployed student's architecture or simulation steps; hardware energy efficiency was not evaluated.

The BN-moment counterexample highlights a limitation of relying on low-order teacher statistics alone when synthesizing data for a heterogeneous student.

The method does not require new human-participant data collection, but any downstream use should account for dataset bias, model errors, and the relevant data and teacher-model permissions.