StageCal: Shared Neuron Calibration for ANN-to-SNN Conversion with Few-Step Inference

Anonymous authors
Paper under double-blind review

Abstract

Converting pretrained artificial neural networks (ANNs) into spiking neural networks (SNNs) aims to reuse learned representations while addressing severe clipping, quantization, and temporal mismatch errors at short simulation steps. To mitigate these errors, existing calibration methods commonly reconstruct ANN activations through layer-wise parameter adjustment. Consequently, these methods fail to preserve network-wide feature alignment across heterogeneous stages and varying inference budgets. In this paper, we find that isolated parameter fitting inherently overlooks the complex interactions of temporal conversion errors. Effective calibration cannot treat neuron operating parameters as independent variables. Instead, *it must govern them as a globally coordinated system anchored by shared network statistics.* In light of this, we propose **STAGECAL**, a novel *shared neuron calibration* framework for plug-and-play ANN-to-SNN conversion that keeps pretrained ANN backbone weights strictly fixed. Our framework brings two key technical contributions. Firstly, we introduce a shared calibration network. It utilizes channel-wise activation statistics, normalized stage depth, and target spike budgets to jointly predict threshold adjustments and initial membrane potentials. On top, we then introduce a structured residual correction function. This function dynamically predicts mixing coefficients that combine three fixed residual bases from an initial pilot conversion into bounded stage-wise current corrections. These corrections are theoretically linked through an exact membrane-potential balance identity. Extensive experiments on classification tasks demonstrate that STAGECAL even **exceeds the original ANN by** 5.71% **with only** $0.17\times$ **energy and few simulation time steps** on ImageNet. It also reduces hidden-layer synaptic operations by **4.24%** relative to percentile-based conversion. This enables highly efficient SNN inference without running the source ANN and calibration network.

1 Introduction

Spiking neural networks (SNNs) encode information through discrete events and integrate inputs through membrane dynamics (Maass, 1997; Gerstner et al., 2014). Their sparse synaptic activity offers opportunities for efficient inference on hardware supporting event-driven computation (Merolla et al., 2014; Davies et al., 2018; Roy et al., 2019; Davies et al., 2021). However, directly training accurate SNNs remains challenging because non-differentiable spike generation and temporal membrane dynamics complicate gradient-based optimization. To address the challenges, ANN-to-SNN conversion provides a practical approach by reusing representations learned by a pretrained artificial neural network (ANN) (Rueckauer et al., 2017; Cao et al., 2015; Diehl et al., 2015). Continuous ANN activations are approximated by outputs decoded from spike counts accumulated over a finite simulation time steps (Deng & Gu, 2021; Bu et al., 2023). Shorter time steps reduce latency but limit the values that this representation can express (Pfeiffer & Pfeil, 2018; Hu et al., 2023). Preserving ANN predictions with only a few simulation steps therefore remains a central challenge.

Existing calibration methods commonly reduce conversion errors by reconstructing ANN activations through layer-wise parameter adjustment (Li et al., 2021; 2024). Related approaches modify ANN activation functions (Bu et al., 2023; Jiang et al., 2023), correct temporal bias (Wu et al., 2024), or adapt neuron dynamics (Wang et al., 2024; Liu et al., 2025). Within neuron calibration, firing thresholds, initial membrane potentials, and current corrections provide effective control over spiking responses (Hwang et al., 2021; Bu et al., 2022; Li et al., 2021). However, their effects are coupled: thresholds govern representable range and quantization resolution (Deng & Gu, 2021; Bu et al., 2023),

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.

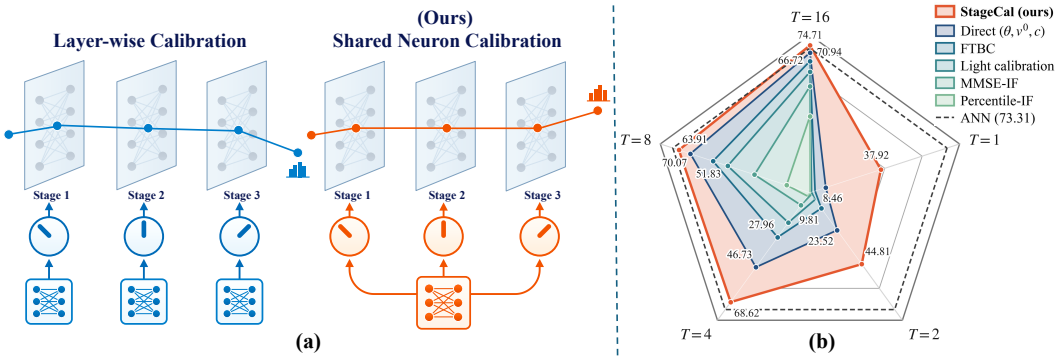

Figure 1: **(a)** StageCal replaces separate layer-wise calibrators with a shared network that generates channel-specific neuron parameters from activation statistics and inference conditions. **(b)** ImageNet-1K top-1 accuracy (%) on ResNet-34 at $b = 1$. StageCal consistently outperforms competing calibration methods across all simulation horizons, with substantial gains at short horizons and accuracy exceeding the frozen ANN reference at $T = 16$.

initialization shifts spike timing (Bu et al., 2022; Hao et al., 2023b), and current correction alters the input accumulated over time (Li et al., 2021; Wu et al., 2024). These dependencies motivate coordinated calibration across stages, conditioned on local activation statistics.

In this paper, we aim to address a novel **research question**: *Can a shared calibration network coordinate neuron parameters across network stages and inference budgets to preserve ANN predictions while keeping pretrained weights fixed?* Addressing this question is nontrivial due to two technical challenges. **First, conversion errors interact across mechanisms and stages.** Clipping and quantization errors are defined through a static approximation of ANN activations. Actual SNN responses also depend on time-varying inputs and residual membrane potentials as well as errors propagated from earlier stages. Thus, correcting these errors independently may overlook their interactions. Local reconstruction alone does not guarantee agreement with the ANN's final predictions. **Second, calibration must accommodate different inference budgets and channel statistics.** Channel-wise activation statistics require distinct parameters, while time steps and spike budgets shape the calibration objective. Therefore, independent fitting offers flexibility but lacks a shared calibration rule across the network and inference budgets.

To this end, we introduce **StageCal**, a novel shared neuron calibration framework for low-latency ANN-to-SNN conversion. As illustrated in Fig. 1, ***our key idea is to jointly calibrate neuron parameters through a shared mapping from activation statistics and inference budgets.*** The calibration network uses channel-wise statistics and normalized stage depth together with the number of simulation time steps and spike budget to predict threshold adjustments and initial membrane potentials alongside signed residual mixing coefficients. One network generates channel-specific parameters across stages and budgets for each source checkpoint. Its parameter count is independent of backbone depth and width. StageCal exactly decomposes pilot conversion error into clipping and static quantization terms plus a membrane and propagation residual. Their channel-wise means define three fixed residual bases whose learned bounded signed mixture provides current corrections before binary spike generation. Calibration uses unlabeled training images to align features and logits with the frozen ANN and encourage prediction agreement. Only the calibration network is updated under a spike-budget penalty and parameter regularization. Each simulation time step and budget yields a fixed SNN configuration with an additive correction to the signed linear readout. Inference requires neither the ANN nor the calibration network. Our analysis links current correction and initialization through an exact membrane-potential balance identity and characterizes scalar quantization error under constant-current assumptions. A conditional network-level analysis connects activation errors to ANN prediction agreement.

Our main contributions are three-fold: **1)** We introduce StageCal, a shared neuron calibration framework that jointly adjusts thresholds, membrane initialization, and residual mixing across stages and inference budgets while keeping the ANN backbone fixed. Three fixed residual bases yield bounded current corrections, whose effects are characterized by an exact membrane-potential balance identity and conditional error bounds. **2)** We develop an unlabeled calibration objective that aligns ANN and SNN representations and predictions while controlling spike activity. It updates only the

calibration network and produces fixed SNN configurations that require neither the source ANN nor the calibration network during inference. **3)** We demonstrate strong classification performance across CIFAR-10, CIFAR-100, and ImageNet. At $T = 1$, StageCal outperforms direct calibration by 9.26% and 9.41% percentage points on CIFAR-10 and CIFAR-100, respectively. STAGECAL even **exceeds the original ANN by** 5.71% **with only** $0.17\times$ **energy and few simulation time steps** on ImageNet. It also reduces hidden-layer synaptic operations by **4.24%** relative to percentile-based conversion. Selected configurations also surpass the source ANN.

2 Related Work

SNN conversion and low-latency approximation. Early ANN-to-SNN conversion mapped continuous activations to spike counts through weight and threshold balancing (Cao et al., 2015; Diehl et al., 2015). Subsequent work generalized supported operations and analyzed finite-window errors (Rueckauer et al., 2017). Conversion of VGG and residual networks highlighted activation-scale preservation (Sengupta et al., 2019). RMP-SNN retains residual membrane potential through soft reset (Han et al., 2020) while Spiking-YOLO combines channel-wise normalization with signed neurons for detection (Kim et al., 2020). For low-latency conversion, Deng & Gu (2021) analyze layer-wise error and combine threshold balancing with soft reset. Rate Norm Layer shapes source activations (Ding et al., 2021) while QCFS incorporates quantization-aware clip-floor-shift activations into ANN training (Bu et al., 2023). SlipReLU jointly addresses source performance and conversion error (Jiang et al., 2023). Fast-SNN links ANN quantization to temporal discretization and addresses sequential error through signed neurons and layer-wise fine-tuning (Hu et al., 2023). Temporal coding optimizes neuron dynamics to represent activations with few spikes (Stöckl & Maass, 2021). StageCal calibrates existing ReLU checkpoints with frozen synaptic parameters and binary hidden spikes to isolate converter-side gains from source-activation changes or backbone retraining.

Calibration and neuron operating parameters. Post-training calibration reduces ANN–SNN activation mismatch. SNN Calibration adjusts biases in its light pipeline and membrane potentials and weights in its advanced pipeline (Li et al., 2021). Error-Aware Conversion combines layer-wise error analysis with parameter calibration, supporting synthetic data (Li et al., 2024). These methods use activation reconstruction to address propagated errors, distinguishing operating-parameter correction from weight adaptation. Pre-charging reduces firing delay, and delayed evaluation suppresses transients (Hwang et al., 2021). Optimized initialization connects initial membrane states to conversion loss (Bu et al., 2022). Residual membrane analysis explains errors from uneven spike arrival (Hao et al., 2023a), while offset-spike calibration corrects spike counts through initialization (Hao et al., 2023b). FTBC calibrates time-dependent biases through forward passes (Wu et al., 2024). Adaptive Calibration combines adaptive firing, spike compression, and input-dependent horizons (Wang et al., 2024). Error Compensation Learning combines learnable clipping, dual-threshold neurons, and membrane initialization (Liu et al., 2025).

StageCal shares a statistical policy across channels and stages to jointly calibrate thresholds and initial potentials with static current corrections formed from signed mixtures of fixed pilot residual bases. Source kernels and biases remain frozen alongside batch-normalization statistics while a separate additive correction handles the signed linear readout. Independently optimized channel-wise controls provide a direct baseline for isolating the effect of sharing.

Shared parameter generation and matching losses. HyperNetworks generate weights through a shared auxiliary network (Ha et al., 2017). Input-conditioned filters (Jia et al., 2016) and affine feature modulation (Perez et al., 2018) provide related forms of conditional parameterization. StageCal calibrates one statistical policy per source checkpoint to generate converter controls across stages and operating points. Policy size is independent of channel count while exported controls remain channel specific. Offline export removes both the policy and source ANN from deployment.

Calibration draws on output distillation (Hinton et al., 2015) and intermediate supervision (Romero et al., 2015). Related approaches align attention and features (Zagoruyko & Komodakis, 2017; Heo et al., 2019) or relationships among examples (Park et al., 2019). StageCal matches features and logits while preserving ANN-predicted classes on unlabeled images. A fan-out-weighted spike penalty regulates an activity proxy under the target budget. Surrogate gradients enable optimization through discrete spike decisions (Neftci et al., 2019) as demonstrated by SuperSpike and spatio-temporal backpropagation (Zenke & Ganguli, 2018; Wu et al., 2018). Hybrid conversion initializes spike-based fine-tuning from converted networks (Rathi et al., 2020). StageCal backpropagates through a frozen SNN to update only the shared policy and exports bounded operating controls.

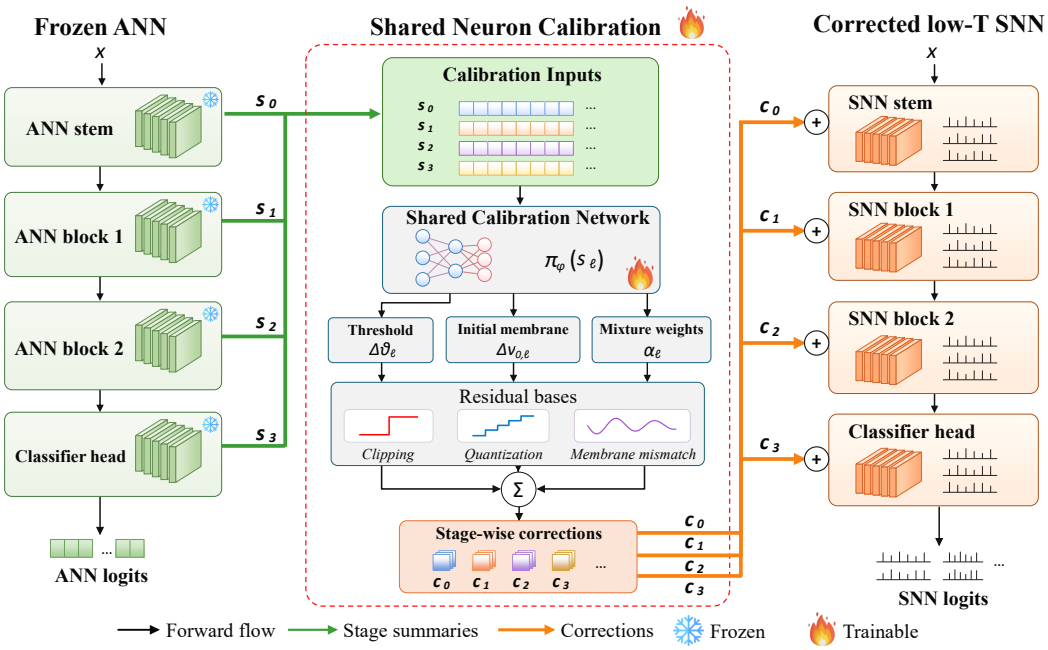

Figure 2: **StageCal framework.** A frozen ANN (left) supplies calibration statistics and targets. A shared network (center) predicts threshold adjustments, membrane initialization, and residual mixtures for stage-wise corrections. The SNN (right) uses exported parameters and a frozen linear readout with logit correction. Only the shared network is optimized, entirely offline.

3 Shared Neuron Calibration

3.1 Overview and Problem Setting

ANN-to-SNN conversion approximates continuous ANN activations with outputs decoded from accumulated spike counts (Rueckauer et al., 2017). At short simulation horizons, discretization and membrane dynamics introduce errors that can propagate across network stages. Calibration reduces these errors by adjusting neuron parameters to improve agreement with the source (Li et al., 2021).

We consider a pretrained ReLU network with L calibrated nonnegative activation stages and a signed linear classifier. Each stage corresponds to a conversion site at a nonnegative ANN activation. Batch normalization is folded into affine operations using frozen statistics, with kernels and biases fixed throughout calibration. An unlabeled training subset $\mathcal{D}_{\text{cal}} = \{x_i\}_{i=1}^{N}$ supplies activation statistics and ANN targets, while a disjoint training subset $\mathcal{D}_{\text{sel}}$ supports configuration selection. We calibrate neuron parameters for horizon T and target spike-density budget b to preserve agreement with ANN. The converted network uses non-leaky integrate-and-fire neurons with subtractive reset (Rueckauer et al., 2017; Han et al., 2020). For an ordinary feedforward connection with $l > 1$, the synaptic current and membrane update are

$$I_l^t = W_l\big(\theta_{l-1} \odot s_{l-1}^t\big) + b_l, \qquad u_l^t = v_l^{t-1} + I_l^t + c_l, \tag{1}$$

$$s_l^t = \mathbf{1}[u_l^t \geq \theta_l], \quad v_l^t = u_l^t - \theta_l \odot s_l^t, \quad \widehat{a}_l(T) = \frac{\theta_l}{T} \odot \sum_{t=1}^{T} s_l^t. \tag{2}$$

Here u_l^t and v_l^t are pre- and post-reset membrane potentials, and $\widehat{a}_l(T)$ approximates the ANN activation. Both $\mathbf{1}[\cdot]$ and $\odot$ act elementwise. Neurons emit at most one binary spike per step, retaining unclamped membrane potentials after subtractive reset. Channel-wise parameters (θ_l, v_l^0, c_l) are broadcast spatially. The first stage receives constant analog current $I_1^t = W_1 x + b_1$, while frozen residual edges use the same activation units. Stages execute in topological order at each of exactly T steps. A frozen linear readout produces signed logits:

$$z^{\text{S}}(x, T, b) = W_o\,\text{Pool}(\widehat{a}_L(T)) + b_o + c_o, \tag{3}$$

where Pool denotes feature pooling and c_o an additive output correction. Total inference cost includes the dense readout. For each exported (T, b), calibration parameters remain fixed across inputs and time, so inference uses only the converted SNN and its stored parameters.

Our idea. As illustrated in Fig. 2, StageCal learns a **shared calibration rule** that coordinates neuron parameters across channels and stages under different inference budgets. A shared calibration network uses **channel-wise activation statistics** and normalized stage depth together with (T, b) to predict threshold adjustments and initial membrane potentials. It also predicts signed coefficients that combine **three fixed residual bases** from an initial SNN conversion into **bounded current corrections**. The shared rule generates **distinct parameters for each channel** while coupling their optimization across the network. Feature and output matching through the full converted SNN trains **only the calibration network**. Statistics extraction and calibration are performed offline, and each (T, b) yields a **fixed SNN configuration**.

3.2 Activation Statistics and Shared Calibration Network

Insights. Calibration requirements vary across channels, network stages, and inference budgets. A shared calibration rule captures **common structure across these conditions while retaining channel-specific outputs.** Our network jointly predicts threshold adjustments, membrane initialization, and residual-mixture coefficients from activation statistics and inference conditions.

The shared calibration network generates neuron parameters from activation statistics and target inference budgets. Let $a_{lj}(x)$ denote channel j of the ANN activation at stage l. We compute its statistics over calibration examples and spatial positions. The reference threshold θ^0_{lj} is the 99.9th percentile of these activations, bounded below by ϵ. A pilot SNN uses these reference thresholds with half-threshold membrane initialization and zero current correction. For each channel, we combine normalized activation statistics with stage depth and the target inference budget to form the input

$$\mathbf{h}_{lj}(T, b) = \left[\frac{\mu_{lj}}{\theta^0_{lj}}, \frac{\sigma_{lj}}{\theta^0_{lj}}, \log\left(1 + \frac{m_{lj}}{\theta^0_{lj}}\right), \frac{p^{95}_{lj}}{\theta^0_{lj}}, \frac{l}{L}, \frac{1}{T}, b \right]^\top , \tag{4}$$

where μ_{lj} and σ_{lj} are the activation mean and standard deviation. The maximum and 95th percentile are denoted by m_{lj} and p^{95}_{lj}. The activation statistics are computed once and reused across inference budgets. A shared calibration network $\pi_\phi : \mathbb{R}^7 \to \mathbb{R}^5$ processes each input independently. It is implemented as an MLP with two ReLU hidden layers of width 64. Its outputs are $(d^\theta, d^v, q_1, q_2, q_3)_{lj}$. The first two outputs determine the firing threshold and initial membrane potential:

$$\theta_{lj} = \theta^0_{lj} \exp\big(\eta_\theta \tanh d^\theta_{lj}\big), \qquad \rho_{lj} = \tfrac{1}{2} + \eta_v \tanh d^v_{lj}, \qquad v^0_{lj} = \rho_{lj}\theta_{lj}. \tag{5}$$

Here ρ_{lj} is the initial membrane potential normalized by the threshold. We set $\eta_\theta = \log 2$ to keep the threshold within a factor of two of its reference value. Setting $\eta_v = 0.49$ ensures $0 < \rho_{lj} < 1$. The remaining outputs determine the signed mixing coefficients used for residual correction:

$$\alpha_{ljk} = \tfrac{1}{3} \tanh q_{ljk}, \qquad k \in \{1, 2, 3\}. \tag{6}$$

This ensures $\sum_k |\alpha_{ljk}| \le 1$, bounding residual contributions. Zero-initializing the final MLP layer recovers pilot settings without correction. Activation statistics and normalized depth encode stage and channel differences without learned embeddings, yielding channel-specific parameters with a trainable parameter count independent of backbone depth and width.

3.3 Residual-Based Current Correction

Insights. Finite-horizon conversion combines clipping, quantization, and membrane or propagation mismatch. **We construct fixed residual bases for these components and learn bounded mixtures through the shared calibration network.** The resulting channel-specific corrections act before spike generation, preserving binary hidden spikes.

We compare ANN activations a_l with pilot SNN outputs $\widehat{a}^0_l(T)$ at each calibration horizon. Define $C_\theta(a) = \mathrm{clip}(a, 0, \theta)$ and the static quantizer

$$Q_{\theta,\rho,T}(a) = \frac{\theta}{T} \,\mathrm{clip}\left(\left\lfloor \frac{Ta}{\theta} + \rho \right\rfloor , 0, T \right). \tag{7}$$

All operations are elementwise. The quantizer maps activations to finite output levels over T steps, with ρ shifting the quantization boundaries. Using pilot thresholds θ_l^0 and $\rho = 1/2$, we decompose the pilot conversion error as

$$R_l^{\text{clip}} = a_l - C_{\theta_l^0}(a_l), \qquad R_l^{\text{quant}} = C_{\theta_l^0}(a_l) - Q_{\theta_l^0,1/2,T}(a_l), \tag{8}$$

$$R_l^{\text{mem}} = Q_{\theta_l^0,1/2,T}(a_l) - \widehat{a}_l^0(T). \tag{9}$$

These components sum exactly to $a_l - \widehat{a}_l^0(T)$. The first two capture clipping and static quantization errors. The third captures the remaining discrepancy from membrane dynamics, time-varying inputs, and errors propagated from earlier stages.

Indexing the components by $k = 1, 2, 3$, their means over calibration examples and spatial positions define fixed channel-wise bases $g_{ljk}(T) = \mathbb{E}_{x,p}[R_{ljk}(x, p, T)]$. The coefficients from Eq. (6) combine them into a bounded current correction:

$$d_{lj} = \sum_{k=1}^{3} \alpha_{ljk} g_{ljk}(T), \qquad c_{lj} = \kappa\theta_{lj} \tanh\left(\frac{d_{lj}}{\kappa\theta_{lj} + \epsilon}\right). \tag{10}$$

The coefficient constraint ensures $|d_{lj}| \le \max_k |g_{ljk}(T)|$, while the final transformation bounds $|c_{lj}| \le \kappa\theta_{lj}$ with $\kappa = 0.1$. Thus, the pilot residuals constrain the mixture, and the threshold limits its magnitude. The bases remain fixed during calibration, while their mixing coefficients, thresholds, and initial membrane potentials are jointly optimized. Corrections enter Eq. (1) before spike generation.

For the signed linear readout, we use only the shared network's residual-mixing outputs. The fixed scale θ_{oj}^0 is the 99.9th percentile of $|z_j^{\text{A}}|$ on $\mathcal{D}_{\text{cal}}$, lower-bounded by ϵ. Descriptor means and standard deviations use signed logits, while maxima and percentiles use absolute logits. We set $g_{oj3} = \mathbb{E}_x[z_j^{\text{A}} - z_j^{\text{S},0}]$ and $g_{oj1} = g_{oj2} = 0$. Equation (10) uses θ_{oj}^0 to produce an additive correction c_{oj}, preserving the signed linear classifier. Only the shared network is trained on unlabeled data using feature and logit matching, teacher-decision preservation, a spike-budget penalty, and operating-parameter regularization. Each (T, b) yields a fixed SNN configuration requiring neither the ANN nor the calibration network at inference. Appendix A details the objective, optimization, and export procedure.

3.4 Conversion Error Analysis

We first relate current correction and membrane initialization through an exact identity for the simulated neuron dynamics. We then examine a constant-current case to clarify the static quantization model used to construct the residual bases.

Proposition 1 (Membrane-potential balance)**.** *For the dynamics in Eqs. (1) and (2) with fixed thresholds and current corrections and no membrane-potential clamping,*

$$\widehat{a}_l(T) = \frac{1}{T}\sum_{t=1}^{T} I_l^t + c_l + \frac{v_l^0 - v_l^T}{T}. \tag{11}$$

This identity holds for signed input currents and arbitrary terminal membrane potentials.

Summing the recurrence $v_l^t - v_l^{t-1} = I_l^t + c_l - \theta_l \odot s_l^t$ over time cancels the intermediate membrane states and yields Eq. (11). The decoded activation is therefore determined by the mean input current and the current correction together with the change in membrane potential. Since the terminal potential depends on both the correction and initialization, the identity motivates calibrating these parameters jointly. To obtain an explicit quantization bound, consider a scalar neuron receiving a constant current $a \in [0, \theta]$ with zero correction and $v^0 = \rho\theta$ for $0 \le \rho < 1$. Under these assumptions, its decoded output equals the static quantizer $Q_{\theta,\rho,T}(a)$ in Eq. (7). The quantization step is θ/T, and the initialization parameter ρ determines the balance between overestimation and underestimation. Consequently,

$$\left|a - Q_{\theta,\rho,T}(a)\right| \le \frac{\theta}{T}\max\{\rho, 1-\rho\}.$$

Half-threshold initialization minimizes the worst-case scalar bound to $\theta/(2T)$, motivating the pilot initialization. Time-varying inputs and upstream errors in deeper stages can violate the constant-current assumptions, so we allow calibrated initial membrane potentials to differ from pilot values. Appendix B provides proofs and a conditional error-propagation bound linking local activation errors to agreement with ANN predictions.

Table 1: **Comparison of ANN baselines and SNN methods on image classification benchmarks.** Results are grouped by baseline model. Accuracy is reported in percent.

Baseline model	Method	T	Accuracy (%) ↑				Relative energy
			CIFAR-10	CIFAR-100	Caltech-101	ImageNet-1K	
CLIP ViT-B/32	ANN Original [Radford et al. (2021)]	–	91.30	65.10	87.90	63.20	1.00×
	STA [Jiang et al. (2024)]	32	87.71	62.55	88.82	66.50	–
	SNNC-inspired Light (Hybrid) [Li et al. (2021)]	32	84.50	49.50	81.58	51.50	–
	SpikedAttention [Hwang et al. (2024)]	40	87.00	59.00	91.54	66.00	–
	Adaptive Calib. [Wang et al. (2025)]	32	84.50	60.00	88.31	64.00	–
	SpikeZIP-TF [You et al. (2024)]	256	**92.00**	56.00	43.98	16.50	–
	Ours	4	89.42	**63.84**	**92.61**	**67.37**	0.16×
CLIP ViT-B/16	ANN Original [Radford et al. (2021)]	–	91.60	68.70	89.30	68.60	1.00×
	STA [Jiang et al. (2024)]	32	78.50	45.50	80.00	52.00	–
	SNNC-inspired Light (Hybrid) [Li et al. (2021)]	32	81.00	39.00	66.81	48.50	–
	SpikedAttention [Hwang et al. (2024)]	40	88.00	67.00	91.57	73.50	–
	Adaptive Calib. [Wang et al. (2025)]	32	88.50	66.00	87.73	72.00	–
	Ours	4	**90.74**	**68.18**	**92.84**	**74.31**	0.17×
CLIP ResNet50	ANN Original [Radford et al. (2021)]	–	71.55	41.95	82.33	59.83	1.00×
	SNNC-Advanced‡ [Li et al. (2021)]	32	15.70	1.43	2.36	2.04	–
	QCFS(Distill)‡ [Bu et al. (2023)]	32	9.98	1.14	4.60	0.19	–
	Adaptive Calib‡ [Wang et al. (2025)]	32	35.72	14.68	67.86	39.40	–
	Calib(STA setting)‡ [Li et al. (2021); Jiang et al. (2024)]	32	66.69	32.46	80.10	54.74	–
	Ours	4	**69.23**	**39.84**	**81.67**	**56.18**	0.10×
ResNet34	ANN Original [He et al. (2016)]	–	96.82	84.19	89.59	73.31	1.00×
	SNNC-Advanced [Li et al. (2021)]	32	92.44	74.65	86.59	63.89	–
	QCFS [Bu et al. (2023)]	32	65.48	40.28	84.47	47.67	–
	Adaptive Calib. [Wang et al. (2025)]	32	96.28	83.03	89.75	**71.80**	–
	Ours	4	**96.56**	**83.75**	**90.24**	68.62	0.18×

4 EXPERIMENTS

We evaluate STAGECAL through controlled conversion experiments, component ablations, and deployment measurements. The evaluation addresses three questions:

Q-1: How well does STAGECAL preserve accuracy at short inference horizons?
Q-2: How does shared calibration compare with directly optimized operating parameters?
Q-3: What computational and deployment costs accompany the accuracy gains?

4.1 EXPERIMENTAL SETUP

The controlled evaluation uses CIFAR-10/100 (Krizhevsky, 2009), Caltech-101 (Fei-Fei et al., 2007), and ImageNet-1K (Russakovsky et al., 2015) with ResNet-18/34 (He et al., 2016), CLIP (Radford et al., 2021) ViT-B/32, ViT-B/16 (Dosovitskiy et al., 2021), and ResNet-50 backbones. Controlled comparisons share a fixed ANN checkpoint, preprocessing, and data splits. Calibration and configuration selection each use $1{,}024$ disjoint training images, without class-balanced sampling or ground-truth calibration losses. Baselines include percentile-threshold IF conversion, MMSE-IF, light calibration (Li et al., 2021), FTBC (Wu et al., 2024), and direct optimization of bounded channel-wise (θ, v^0, c) with readout-bias correction. The direct control shares our losses and parameter bounds but freely optimizes current corrections, testing the shared residual parameterization against a more flexible alternative. Tab. 1 includes additional SOTA conversion strategies and their corresponding references. We evaluate $T \in \{1, 2, 4, 8, 16\}$, training one shared network per checkpoint across (T, b) conditions and separate direct configurations under equal total unrolled sample-step budgets. Calibration draws are paired across methods using five seeds. Top-1 accuracy and, where shown, mean $\pm$ standard deviation measure calibration variability with fixed checkpoints. Main time-step comparisons use $b = 1$, with no active spike-density penalty. *Appendices A and C detail more on implementation specification and experimental protocol.*

4.2 CLASSIFICATION ACCURACY

Comparison across source models. Tab. 1 compares ANN baselines and SNN methods across image classification benchmarks. For CLIP ViT-B/16, Adp.C achieves 88.50% accuracy on C-10 and 72.00% on ImageNet at $T = 32$. The corresponding accuracies STAGECAL are **90.74%** and **74.31%** at $T = 4$, achieving higher accuracy (even exceeding ANN) on both with fewer inference steps.

Accuracy across inference horizons. Fig. 3 compares conversion accuracy on CIFAR-10 and CIFAR-100. At $T = 1$, STAGECAL exceeds direct optimization by 9.26 and 9.41 percentage points,

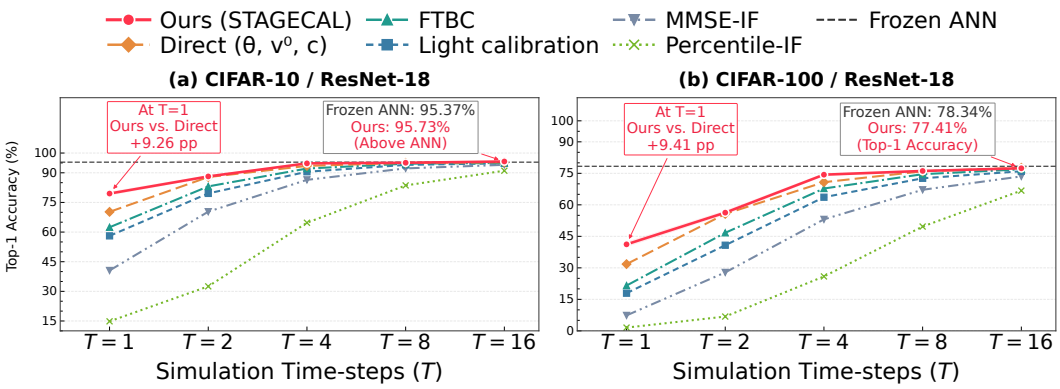

Figure 3: **Accuracy across inference horizons.** Top-1 accuracy at $b = 1$ on (a) CIFAR-10 and (b) CIFAR-100 with a fixed ResNet-18 checkpoint. *Zoom for a better view.*

Table 2: ImageNet-1K top-1 accuracy (%) at $b = 1$, over five calibration seeds.

Method	$T = 1$	$T = 2$	$T = 4$	$T = 8$	$T = 16$
ImageNet-1K / ResNet-18					
Frozen ANN reference			69.76		
Percentile-IF	0.16 ± 0.04	0.69 ± 0.12	4.82 ± 0.43	20.37 ± 0.91	47.62 ± 0.72
MMSE-IF	0.44 ± 0.09	2.91 ± 0.28	12.76 ± 0.67	36.84 ± 0.82	57.93 ± 0.49
Light calibration	1.83 ± 0.22	7.64 ± 0.53	24.59 ± 0.84	48.91 ± 0.63	63.44 ± 0.31
FTBC	5.72 ± 0.41	16.93 ± 0.69	36.81 ± 0.76	56.72 ± 0.46	66.14 ± 0.21
Direct (θ, v^0, c)	10.83 ± 0.62	27.96 ± 0.78	49.38 ± 0.61	63.27 ± 0.34	$\mathbf{68.49 \pm 0.13}$
StageCal (ours)	$\mathbf{20.26 \pm 0.65}$	$\mathbf{39.14 \pm 0.71}$	$\mathbf{61.06 \pm 0.54}$	$\mathbf{64.18 \pm 0.29}$	68.31 ± 0.12
ImageNet-1K / ResNet-34					
Frozen ANN reference			73.31		
Percentile-IF	0.11 ± 0.01	0.27 ± 0.06	2.13 ± 0.25	12.48 ± 0.74	38.75 ± 0.93
MMSE-IF	0.19 ± 0.04	1.17 ± 0.16	7.82 ± 0.51	29.64 ± 0.88	53.91 ± 0.62
Light calibration	0.86 ± 0.14	4.38 ± 0.39	18.73 ± 0.79	43.96 ± 0.77	61.27 ± 0.38
FTBC	2.74 ± 0.31	9.81 ± 0.57	27.96 ± 0.86	51.83 ± 0.58	66.72 ± 0.25
Direct (θ, v^0, c)	8.46 ± 0.55	23.52 ± 0.82	46.73 ± 0.73	63.91 ± 0.39	70.94 ± 0.16
StageCal (ours)	$\mathbf{37.92 \pm 0.59}$	$\mathbf{44.81 \pm 0.76}$	$\mathbf{68.62 \pm 0.66}$	$\mathbf{70.07 \pm 0.33}$	$\mathbf{74.71 \pm 0.14}$

respectively. The gap narrows as the horizon increases, consistent with the greater representational resolution available from longer spike sequences. At $T = 16$, StageCal reaches 95.73% on CIFAR-10 and 77.41% on CIFAR-100, compared with ANN references of 95.37% and 78.34%, respectively. These correspond to signed accuracy changes of $+0.36$ and -0.93 percentage points. Together, these results demonstrate the effectiveness of shared calibration at short inference horizons while retaining accuracy close to the ANN references at longer horizons. Tab. 2 extends the comparison to ImageNet-1K. At $T = 4$, StageCal reaches 61.06% on ResNet-18 and 68.62% on ResNet-34, exceeding direct parameter optimization by 11.68 and 21.89 percentage points. At $T = 8$, the corresponding margins are 0.91 and 6.16 points. The advantage is not uniform across operating conditions. At $T = 16$, direct optimization reaches 68.49% on ResNet-18, compared with 68.31% for StageCal. On ResNet-34, StageCal reaches **74.71%**, which is **3.77** points above the direct control and **1.40** points above its ANN reference.

Takeaway. Calibrated operating parameters reshape the network's input-output mapping, enabling accuracy gains with frozen synaptic weights. Across horizons, shared calibration delivers substantial improvements under tight simulation budgets while maintaining competitive performance at longer horizons, demonstrating its effectiveness for accurate few-step SNN inference.

4.3 Component Ablations

Table 3 examines the calibration heads and residual construction on CIFAR-100/ResNet-18. The full model reaches 56.28% at $T = 2$ and 71.36% at $T = 4$ in this ablation configuration. Removing threshold adjustment reduces accuracy by 4.23 and 2.31 points, while fixing membrane initialization reduces it by 2.86 and 1.44 points. Removing current correction produces larger reductions of 7.12 and 4.31 points. These comparisons support retaining all three controls within the joint calibration objective. Among the residual components, removing the membrane or propagation basis produces the largest accuracy reduction at both horizons, followed by the quantization and clipping bases. At

Table 3: Component ablations on CIFAR-100/ResNet-18. Accuracy (%) is measured at normalized feature MSE and hidden SynOps per image (M) use $T = 4$. Each removal retains the remaining.

Variant	Acc. $T = 2 \uparrow$	Acc. $T = 4 \uparrow$	NMSE ↓	SynOps (M) ↓
w/o threshold head ($\theta = \theta^0$)	52.05	69.05	0.07	214.67
w/o initialization head ($\rho = 1/2$)	53.42	69.92	0.06	220.91
w/o current correction ($c = 0$)	49.16	67.05	0.10	209.76
w/o clipping basis	54.53	70.42	0.05	228.53
w/o quantization basis	53.14	69.65	0.07	218.42
w/o membrane/propagation basis	51.92	68.84	0.08	215.36
Full StageCal	**56.28**	**71.36**	**0.04**	198.64

Table 4: Deployment comparison on ImageNet-1K/ResNet-18 with $T = 4$, $b = 1$, batch size one. Time is synchronized median/p95 milliseconds per image. Energy (mJ) is estimated from operation counts using $E_{\mathrm{MAC}} = 4.6$ pJ and $E_{\mathrm{AC}} = 0.9$ pJ under a 45 nm CMOS model.

Method	Top-1 Acc (%) ↑	Hidden SynOps (M) ↓	Dense MACs (M)↓	Energy mJ ↓	Median ms ↓	p95 ms ↓
Frozen ANN	69.76	–	1814.07	8.34	1.21	1.36
Percentile-IF	4.82	207.43	118.53	0.73	3.68	4.09
Light calibration	24.59	296.81	118.53	0.81	3.82	4.27
FTBC (matched)	36.81	348.26	118.53	0.86	3.94	4.41
Direct (θ, v^0, c)	49.38	437.92	118.53	0.94	4.15	4.64
StageCal	**61.06**	**198.64**	118.53	**0.72**	**2.06**	**2.51**

$T = 4$, the respective reductions are 2.52, 1.71, and 0.94 points. The full model also has an NMSE of 0.04, compared with 0.10 without current correction. These are controlled removal effects, rather than an additive decomposition of total MSE, because the residual bases can be correlated.

Takeaway. These ablations demonstrate the effectiveness of jointly calibrating thresholds, membrane initialization, and residual currents for improving short-horizon accuracy and feature fidelity. Each residual basis contributes to accuracy recovery, with the membrane/propagation basis providing the largest benefit, supporting the proposed structured correction design. Additional ablations are provided in Appendix E.

4.4 Deployment and Calibration Costs

Table 4 evaluates ImageNet-1K/ResNet-18 at $T = 4$, $b = 1$, and batch size one. StageCal achieves 61.06% accuracy with 198.64 million hidden SynOps and 118.53 million dense MACs per image. Synchronized median/p95 latencies are $2.06/2.51$ ms, versus $4.15/4.64$ ms for direct optimization under identical device, backend, and precision settings. Hidden SynOps count spike-triggered operations between spiking stages, while dense MACs cover analog input processing and the linear readout, or all dense layers for the ANN reference. Membrane updates, threshold comparisons, resets, and memory traffic are accounted for separately. Since $b = 1$ imposes no density restriction, these results measure the calibrated operating cost. Operation counts and latency are reported separately.

Takeaway. These results demonstrate that StageCal delivers higher accuracy with fewer hidden SynOps, lower estimated energy, and shorter inference latency than the evaluated SNN baselines. Offline export preserves these deployment benefits without requiring calibration-network evaluation at inference, establishing an effective accuracy-cost advantage through shared calibration. Appendix D details more information on efficiency evaluation protocol.

5 Conclusion

We presented StageCal, a shared calibration network that maps activation summaries and inference conditions to bounded thresholds, initial membrane states, and residual-based current corrections while keeping the source backbone frozen. Offline parameter export eliminates calibration-network evaluation during inference. Experiments demonstrate substantial accuracy gains over matched calibration baselines at short simulation horizons. These results establish shared, condition-dependent calibration as an effective approach to preserving classification performance under limited simulation horizons, advancing practical ANN-to-SNN conversion without backbone retraining.

AI USE STATEMENT

We used generative AI tools to assist with refining methodological descriptions and the interpretation and presentation of results. Additional uses included reorganizing and polishing manuscript text, troubleshooting LaTeX formatting, and developing the StageCal logo. Generative AI tools are not components of the shared calibration network or the deployed SNN. The authors take responsibility for the final content of this work, including methodological choices, mathematical claims and proofs, experimental results, and all text and artifacts produced with AI assistance.

ETHICS STATEMENT

This work studies ANN-to-SNN conversion using existing image-classification benchmarks and pretrained models. It involves no new participant recruitment or collection of human-subject data. Dataset, model, and software use and redistribution are subject to their respective licenses and access conditions.

REPRODUCIBILITY STATEMENT

Sections 3.2 and 3.3 specify the shared calibration network, operating-parameter constraints, and residual correction construction. Appendix B provides proofs and assumptions supporting the theoretical analysis. Appendix A details the calibration objective, optimization, and export procedure, while Appendix C describes the experimental protocol, including calibration and selection splits, baseline comparisons, and inference settings. Section 4.4 and Appendix D describe operation accounting, synchronized latency measurement, calibration-cost accounting, and energy estimation. We will publicly release the implementation, configuration files, and evaluation scripts upon acceptance, together with instructions for obtaining the source datasets and checkpoints under their applicable terms, to support reproduction and consistent comparison.

REFERENCES

Tong Bu, Jianhao Ding, Zhaofei Yu, and Tiejun Huang. Optimized potential initialization for low-latency spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 11–20, 2022. doi: 10.1609/aaai.v36i1.19874. URL https://ojs.aaai.org/index.php/AAAI/article/view/19874.

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.

Yongqiang Cao, Yang Chen, and Deepak Khosla. Spiking deep convolutional neural networks for energy-efficient object recognition. *International Journal of Computer Vision*, 113:54–66, 2015. doi: 10.1007/s11263-014-0788-3. URL https://link.springer.com/article/10.1007/s11263-014-0788-3.

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, Steve 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. URL https://doi.org/10.1109/MM.2018.112130359.

Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A. Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R. Risbud. Advancing neuromorphic computing with Loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5):911–934, 2021. doi: 10.1109/JPROC.2021.3067593. URL https://doi.org/10.1109/JPROC.2021.3067593.

Shikuang Deng and Shi Gu. Optimal conversion of conventional artificial neural networks to spiking neural networks. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2103.00476.

Peter U. Diehl, Daniel Neil, Jonathan Binas, Matthew Cook, Shih-Chii Liu, and Michael Pfeiffer. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In *International Joint Conference on Neural Networks*, pp. 1–8, 2015. doi: 10.1109/IJCNN.2015.7280696. URL https://doi.org/10.1109/IJCNN.2015.7280696.

Jianhao Ding, Zhaofei Yu, Yonghong Tian, and Tiejun Huang. Optimal ANN-SNN conversion for fast and accurate inference in deep spiking neural networks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 2328–2336, 2021. doi: 10.24963/ijcai.2021/321. URL https://www.ijcai.org/proceedings/2021/321.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007. doi: 10.1016/j.cviu.2005.09.012. URL https://authors.library.caltech.edu/records/whcsw-v0f31.

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. URL https://neuronaldynamics.epfl.ch/.

David Ha, Andrew Dai, and Quoc V. Le. HyperNetworks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkpACe1lx.

Bing Han, Gopalakrishnan Srinivasan, and Kaushik Roy. RMP-SNN: Residual membrane potential neuron for enabling deeper high-accuracy and low-latency spiking neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13558–13567, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Han_RMP-SNN_Residual_Membrane_Potential_Neuron_for_Enabling_Deeper_High-Accuracy_and_CVPR_2020_paper.html.

Zecheng Hao, Tong Bu, Jianhao Ding, Tiejun Huang, and Zhaofei Yu. Reducing ANN-SNN conversion error through residual membrane potential. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11–21, 2023a. doi: 10.1609/aaai.v37i1.25071. URL https://ojs.aaai.org/index.php/AAAI/article/view/25071.

Zecheng Hao, Jianhao Ding, Tong Bu, Tiejun Huang, and Zhaofei Yu. Bridging the gap between ANNs and SNNs by calibrating offset spikes. In *International Conference on Learning Representations*, 2023b. URL https://arxiv.org/abs/2302.10685.

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.

Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1921–1930, 2019. URL https://arxiv.org/abs/1904.01866.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

Yangfan Hu, Qian Zheng, Xudong Jiang, and Gang Pan. Fast-SNN: Fast spiking neural network by converting quantized ANN. *arXiv preprint arXiv:2305.19868*, 2023. URL https://arxiv.org/abs/2305.19868.

Sangwoo Hwang, Seunghyun Lee, Dahoon Park, Donghun Lee, and Jaeha Kung. SpikedAttention: Training-free and fully spike-driven transformer-to-SNN conversion with winner-oriented spike shift for softmax operation. In *Advances in Neural Information Processing Systems*, volume 37,

2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/7c9341ad0263428b5057d92f4d88dfa0-Abstract-Conference.html.

Sungmin Hwang, Jeesoo Chang, Min-Hye Oh, Kyung Kyu Min, Taejin Jang, Kyungchul Park, Junsu Yu, Jong-Ho Lee, and Byung-Gook Park. Low-latency spiking neural networks using pre-charged membrane potential and delayed evaluation. *Frontiers in Neuroscience*, 15:629000, 2021. doi: 10.3389/fnins.2021.629000. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2021.629000/full.

Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/8bf1211fd4b7b94528899de0a43b9fb3-Abstract.html.

Haiyan Jiang, Srinivas Anumasa, Giulia De Masi, Huan Xiong, and Bin Gu. A unified optimization framework of ANN-SNN conversion: Towards optimal mapping from activation values to firing rates. In *Proceedings of the International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14945–14974, 2023. URL https://proceedings.mlr.press/v202/jiang23a.html.

Yizhou Jiang, Kunlin Hu, Tianren Zhang, Haichuan Gao, Yuqian Liu, Ying Fang, and Feng Chen. Spatio-temporal approximation: A training-free snn conversion for transformers. In *International Conference on Learning Representations*, volume 2024, pp. 38769–38790, 2024.

Seijoon Kim, Seongsik Park, Byunggook Na, and Sungroh Yoon. Spiking-YOLO: Spiking neural network for energy-efficient object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11270–11277, 2020. doi: 10.1609/aaai.v34i07.6787. URL https://ojs.aaai.org/index.php/AAAI/article/view/6787.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yuhang Li, Shikuang Deng, Xin Dong, Ruihao Gong, and Shi Gu. A free lunch from ann: Towards efficient, accurate spiking neural networks calibration. In *International conference on machine learning*, pp. 6316–6325. PMLR, 2021.

Yuhang Li, Shikuang Deng, Xin Dong, and Shi Gu. Error-aware conversion from ANN to SNN via post-training parameter calibration. *International Journal of Computer Vision*, 132:3586–3609, 2024. doi: 10.1007/s11263-024-02046-2. URL https://link.springer.com/article/10.1007/s11263-024-02046-2.

Chang Liu, Jiangrong Shen, Xuming Ran, Mingkun Xu, Qi Xu, Yi Xu, and Gang Pan. Efficient ANN-SNN conversion with error compensation learning. *arXiv preprint arXiv:2506.01968*, 2025. URL https://arxiv.org/abs/2506.01968.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Wolfgang Maass. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10(9):1659–1671, 1997. doi: 10.1016/S0893-6080(97)00011-7. URL https://igi-web.tugraz.at/PDF/85a.pdf.

Paul A. Merolla, John V. Arthur, Rodrigo Alvarez-Icaza, Andrew S. Cassidy, Jun Sawada, Filipp Akopyan, Bryan L. Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, Bernard Brezzo, Ivan Vo, Steven K. Esser, Rathinakumar Appuswamy, Brian Taba, Arnon Amir, Myron D. Flickner, William P. Risk, Rajit Manohar, and Dharmendra S. Modha. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673, 2014. doi: 10.1126/science.1254642. URL https://doi.org/10.1126/science.1254642.

Emre O. Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks. *arXiv preprint arXiv:1901.09948*, 2019. URL https://arxiv.org/abs/1901.09948.

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. URL https://arxiv.org/abs/1904.05068.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. URL https://arxiv.org/abs/1709.07871.

Michael Pfeiffer and Thomas Pfeil. Deep learning with spiking neurons: Opportunities and challenges. *Frontiers in Neuroscience*, 12:774, 2018. doi: 10.3389/fnins.2018.00774. URL https://doi.org/10.3389/fnins.2018.00774.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Nitin Rathi, Gopalakrishnan Srinivasan, Priyadarshini Panda, and Kaushik Roy. Enabling deep spiking neural networks with hybrid conversion and spike timing dependent backpropagation. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/2005.01807.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6550.

Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575:607–617, 2019. doi: 10.1038/s41586-019-1677-2. URL https://www.nature.com/articles/s41586-019-1677-2.

Bodo Rueckauer, Iulia-Alexandra Lungu, Yuhuang Hu, Michael Pfeiffer, and Shih-Chii Liu. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in Neuroscience*, 11:682, 2017. doi: 10.3389/fnins.2017.00682. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2017.00682/full.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015. doi: 10.1007/s11263-015-0816-y. URL https://arxiv.org/abs/1409.0575.

Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy. Going deeper in spiking neural networks: VGG and residual architectures. *Frontiers in Neuroscience*, 13:95, 2019. doi: 10.3389/fnins.2019.00095. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2019.00095/full.

Christoph Stöckl and Wolfgang Maass. Optimized spiking neurons can classify images with high accuracy through temporal coding with two spikes. *Nature Machine Intelligence*, 3:230–238, 2021. doi: 10.1038/s42256-021-00311-4. URL https://www.nature.com/articles/s42256-021-00311-4.

Ziqing Wang, Yuetong Fang, Jiahang Cao, Hongwei Ren, and Renjing Xu. Adaptive calibration: A unified conversion framework of spiking neural network. *arXiv preprint arXiv:2412.16219*, 2024. URL https://arxiv.org/abs/2412.16219.

Ziqing Wang, Yuetong Fang, Jiahang Cao, Hongwei Ren, and Renjing Xu. Adaptive calibration: A unified conversion framework of spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 1583–1591, 2025.

Xiaofeng Wu, Velibor Bojkovic, Bin Gu, Kun Suo, and Kai Zou. FTBC: Forward temporal bias correction for optimizing ANN-SNN conversion. *arXiv preprint arXiv:2403.18388*, 2024. URL https://arxiv.org/abs/2403.18388.

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. doi: 10.3389/fnins.2018.00331. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2018.00331/full.

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 the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57367–57383. PMLR, 2024. URL https://proceedings.mlr.press/v235/you24b.html.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1612.03928.

Friedemann Zenke and Surya Ganguli. SuperSpike: Supervised learning in multilayer spiking neural networks. *Neural Computation*, 30(6):1514–1541, 2018. doi: 10.1162/neco_a_01086. URL https://arxiv.org/abs/1705.11146.

A Implementation Details

This appendix specifies the calibration procedure, spiking dynamics, export representation, and experimental protocol used by StageCal.

A.1 Calibration and export procedure

Calibration operates on a frozen source ANN and produces fixed SNN operating parameters for each deployment condition.

1. **Source preparation.** The source checkpoint remains frozen throughout calibration. Batch normalization operates in evaluation mode and is folded into the corresponding affine transformations. Converted stages correspond to nonnegative activation sites, with channel dimensions and residual connections inherited from the source architecture.
2. **Activation statistics.** Channel moments and quantiles are computed on $\mathcal{D}_{\text{cal}}$ in the original activation units. Initial thresholds θ^0 use the 99.9th-percentile activation with a numerical floor of $\epsilon = 10^{-6}$. Teacher features and logits are detached from the calibration graph.
3. **Pilot simulation.** Each training horizon uses an uncorrected pilot SNN with $v^0 = \theta^0/2$. The pilot computes the three channel residual means in Eqs. (8)–(9) on the same calibration samples. These channel means remain fixed throughout policy optimization.
4. **Policy optimization.** The two hidden MLP layers use variance-preserving initialization, and the output layer is initialized to zero. Each optimization step samples a calibration minibatch and a uniformly drawn (T, b) pair. The shared policy generates the operating parameters, the SNN executes hard-spike forward simulation, and the surrogate gradient propagates Eq. (16) through the temporal graph. Only ϕ receives optimizer updates.
5. **Checkpoint selection.** The policy is evaluated on $\mathcal{D}_{\text{sel}}$ every 100 optimization steps. Selection minimizes the mean unregularized calibration objective across the training conditions using fixed loss weights and teacher predictions. Ground-truth labels are excluded from calibration and selection.
6. **Offline export.** A single policy evaluation produces θ, v^0, c, and the signed readout correction for each requested profile. Every new image starts from the exported membrane initialization. The deployed computation contains the converted SNN and its exported parameters; the teacher, policy, summaries, and pilot tensors are absent from the inference graph.

Surrogate and regularization. For $q = u/\theta - 1$, the backward derivative of $\mathbf{1}[q \geq 0]$ is approximated by $\gamma \max(0, 1 - |q|)$ with $\gamma = 1$. Gradients propagate through the normalized argument, threshold-dependent reset, and trainable initial state. Forward simulation retains the hard binary spike function. Let $\langle \cdot \rangle_{l,j}$ denote an equal average over hidden stage–channel pairs. The operating regularizer is

$$\mathcal{R}_{\text{op}} = \left\langle \left(\frac{\log(\theta_{lj}/\theta^0_{lj})}{\eta_\theta} \right)^2 + \left(\frac{\rho_{lj} - 1/2}{\eta_v} \right)^2 + \left(\frac{c_{lj}}{\kappa \theta_{lj} + \epsilon} \right)^2 \right\rangle_{l,j} + \frac{1}{C_o} \sum_j \left(\frac{c_{oj}}{\kappa q^{99.9}_{oj} + \epsilon} \right)^2, \tag{12}$$

where $q^{99.9}_{oj}$ is the floored 99.9th percentile of the absolute teacher logits. The same numerical floor stabilizes divisions for inactive channels. In the signed output summary, μ and σ are the mean and standard deviation of signed logits, whereas m, p^{95}, and $q^{99.9}$ are computed from absolute logits. The output depth coordinate equals one.

Policy size and storage. The shared 7–64–64–5 MLP contains $(7+1)64+(64+1)64+(64+1)5 = 4{,}997$ trainable scalars, including biases. Each exported profile stores $3\sum_l C_l + C_o$ operating scalars before bias folding. Storage for P profiles is therefore $P(3\sum_l C_l + C_o)$ scalars. The shared policy is used only during calibration and export. Frozen backbone weights, optimizer state, temporal

activations, teacher tensors, and exported profiles occupy distinct memory components; trainable parameter count describes only the optimized component.

Residual and pooling conventions. At each converted post-addition ReLU, the main-branch and identity or projection currents are summed before the spike decision. Intermediate ReLU sites in the main branch are also converted. Projection weights and skip connections remain frozen, and signed affine transformations without a following nonlinearity remain current transforms. Average pooling acts linearly on decoded spikes.

The ImageNet ResNet stem retains its source max-pooling operation through cumulative-count routing. For each pooling window, the routing state tracks the maximum cumulative presynaptic spike count and forwards its increment at each time step. With a common threshold within each pooled channel, the time-averaged decoded output equals the maximum cumulative decoded rate. Binary presynaptic spikes increase each cumulative count by at most one per step, so the routed count increment is also binary. This stateful operator uses cumulative-count buffers and comparisons. At the corresponding edge, the routed increment replaces the presynaptic event in Eq. (1). The budget in Eq. (15) covers ordinary binary hidden edges; routed pooling events and routing overhead have separate accounting boundaries. The same pooling convention applies to all matched methods. The CIFAR VGG average-pooling variant is defined at source training and retains that architecture during conversion.

A.2 Calibration Objective and Offline Export

Calibration aligns intermediate representations and final predictions while controlling spike activity. For input x and condition (T, b), define $p_\tau^{\mathrm{A}} = \mathrm{softmax}(z^{\mathrm{A}}/\tau)$ and $p_\tau^{\mathrm{S}} = \mathrm{softmax}(z^{\mathrm{S}}/\tau)$ as the ANN and SNN class distributions at temperature τ. The teacher-predicted class is $y^{\mathrm{A}} = \arg\max_j z_j^{\mathrm{A}}$.

Intermediate activations are aligned through normalized squared error:

$$\mathcal{L}_{\mathrm{feat}} = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{C_l}\sum_{j}\frac{\mathbb{E}_p[(\widehat{a}_{lj} - a_{lj})^2]}{\sigma_{lj}^2 + \epsilon}, \tag{13}$$

where C_l is the number of channels at stage l, $\mathbb{E}_p$ averages over spatial positions, and σ_{lj}^2 is the calibration variance. Variance normalization reduces sensitivity to channel scale.

Output matching aligns class distributions and preserves the teacher's predicted class:

$$\mathcal{L}_{\mathrm{logit}} = \tau^2\,\mathrm{KL}\big(\mathrm{sg}(p_\tau^{\mathrm{A}}) \,\|\, p_\tau^{\mathrm{S}}\big), \qquad \mathcal{L}_{\mathrm{pres}} = -\log p_1^{\mathrm{S}}(y^{\mathrm{A}}). \tag{14}$$

The KL term transfers the teacher's soft class distribution, and the preservation term treats its predicted class as a pseudo-label. Both losses use teacher outputs without ground-truth annotations and encourage agreement with the source ANN.

Let f_{li} denote the number of outgoing synaptic connections from hidden neuron i to other spiking stages under the ordinary-edge counting convention. The fan-out-weighted spike density and budget penalty are

$$R(x, T, b) = \frac{\sum_{l,i,t} f_{li} s_{li}^t}{T\sum_{l,i} f_{li}}, \qquad \mathcal{L}_{\mathrm{budget}} = [R(x, T, b) - b]_+^2, \tag{15}$$

where $[u]_+ = \max(u, 0)$. Hidden synaptic operations under this convention equal $TR(x, T, b)\sum_{l,i} f_{li}$. Thus, R determines the operation count at a fixed horizon, and comparisons across horizons include the factor T. The penalty imposes a soft activity target. Budget evaluation reports both achieved density and the fraction of images with $R > b$.

The complete calibration objective is

$$\min_{\phi}\ \mathbb{E}_{x\sim\mathcal{D}_{\mathrm{cal}},\,(T,b)\sim\mathcal{P}}\left[\mathcal{L}_{\mathrm{feat}} + \lambda_z\mathcal{L}_{\mathrm{logit}} + \lambda_p\mathcal{L}_{\mathrm{pres}} + \lambda_b\mathcal{L}_{\mathrm{budget}} + \lambda_r\mathcal{R}_{\mathrm{op}}\right], \tag{16}$$

where $\mathcal{P}$ is uniform over the selected horizon–budget pairs. The operating regularizer penalizes normalized changes in thresholds, membrane initialization, and current corrections. Optimization updates only ϕ, with triangular surrogate gradients propagated through the temporally unrolled

SNN (Neftci et al., 2019). Calibration therefore includes temporal simulation and activation-storage costs.

One policy is fitted per source checkpoint across the selected (T, b) conditions. For each deployment condition, π_ϕ maps cached summaries and pilot residuals to fixed parameters (θ_l, v_l^0, c_l) and c_o. Bias folding replaces b_l by $b_l + c_l$, and an equivalent unit-amplitude spike representation absorbs presynaptic threshold scales into outgoing weights. These deterministic transformations preserve the calibrated computation and introduce no additional backbone optimization.

B Proofs and Conditional Error Propagation

B.1 Membrane accounting

For a single neuron, Eq. (2) gives $v^t - v^{t-1} = I^t + c - \theta s^t$. Summing over $t = 1, \dots, T$ yields

$$\theta \sum_{t=1}^{T} s^t = \sum_{t=1}^{T} I^t + Tc + v^0 - v^T. \tag{17}$$

Division by T proves Eq. (11), and elementwise application establishes the vector identity. The derivation holds for arbitrary input currents under the stated fixed-threshold, non-leaky dynamics. Additional voltage clamps, leakage, or refractory updates introduce their corresponding terms into the membrane balance.

B.2 Scalar quantization bound

Assume constant input $I^t = a \in [0, \theta]$, zero correction $c = 0$, and initialization $v^0 = \rho\theta$ with $0 \le \rho < 1$. For $v^{t-1} \in [0, \theta)$, the pre-reset voltage lies in $[0, 2\theta)$. Subtracting θ whenever the pre-reset voltage reaches or exceeds θ preserves $v^t \in [0, \theta)$. Consequently,

$$\sum_{t=1}^{T} s^t = \left\lfloor \frac{Ta}{\theta} + \rho \right\rfloor, \qquad a - \widehat{a}(T) = \frac{\theta}{T}\left(\left\{ \frac{Ta}{\theta} + \rho \right\} - \rho \right), \tag{18}$$

where $\{\cdot\}$ denotes the fractional part. The count lies in $[0, T]$ and agrees with Eq. (7). Since the fractional part belongs to $[0, 1)$,

$$|a - \widehat{a}(T)| \le \frac{\theta}{T} \max\{\rho, 1 - \rho\}. \tag{19}$$

The worst-case scalar error is minimized at $\rho = 1/2$. This result establishes the half-threshold reference for a constant-current quantizer. Deep-network calibration additionally accounts for time-varying inputs and inter-stage propagation through the simulated pilot residuals.

B.3 A conditional network bound

Represent the sequential or residual network by mappings F_l with Lipschitz constants K_l in a chosen norm, where $a_l = F_l(a_{l-1})$. Define $e_l = \|\widehat{a}_l - a_l\|$ and $\varepsilon_l = \|\widehat{a}_l - F_l(\widehat{a}_{l-1})\|$. The local discrepancy ε_l includes temporal coding and current corrections. The triangle inequality gives

$$e_l \le \varepsilon_l + K_l e_{l-1}, \qquad e_L \le \sum_{l=1}^{L} \left(\prod_{k=l+1}^{L} K_k \right) \varepsilon_l \quad \text{for } e_0 = 0. \tag{20}$$

The second inequality follows by induction, with the empty product equal to one. A residual block $F(x) = x + G(x)$ satisfies $K_F \le 1 + K_G$; for a projection skip P, the corresponding bound is $K_F \le \|P\| + K_G$.

Under the infinity norm and a linear final pooling operator, the signed readout satisfies

$$\|z^{\mathrm{S}} - z^{\mathrm{A}}\|_\infty \le \|W_o \,\mathrm{Pool}\,\|_{\infty\to\infty} e_L + \|c_o\|_\infty. \tag{21}$$

This norm bound isolates the magnitudes of the propagated feature discrepancy and readout correction. Their signed interaction determines the realized logit error. Let $\delta = \|z^{\mathrm{S}} - z^{\mathrm{A}}\|_\infty$. An ANN top-two

Table 5: Calibration settings. Optimization steps are totals across sampled operating conditions. The shared policy and gradient-based matched controls use the same optimization settings.

Setting	CIFAR	ImageNet
Calibration / selection images	1,024 / 1,024	1,024 / 1,024
Hidden widths / outputs	$64, 64$ / 5	$64, 64$ / 5
Optimizer / learning rate	AdamW / 10^{-3}	AdamW / 10^{-3}
Weight decay / batch size	10^{-4} / 32	10^{-4} / 32
Total optimization steps	2,000	1,000
Schedule / warm-up	Cosine / 5%	Cosine / 5%
Gradient-norm cap / precision	1.0 / float32	1.0 / float32
Training horizons	$\{1, 2, 4, 8, 16\}$	$\{1, 2, 4, 8, 16\}$
Training budgets	$\{0.1, 0.25, 0.5, 1\}$	$\{0.1, 0.25, 0.5, 1\}$
τ	2	2
$\lambda_z, \lambda_p, \lambda_b, \lambda_r$	$1, 0.1, 10, 10^{-4}$	$1, 0.1, 10, 10^{-4}$
$\eta_\theta, \eta_v, \kappa$	$\log 2, 0.49, 0.1$	$\log 2, 0.49, 0.1$
Calibration seeds	5	5

logit margin greater than 2δ guarantees identical ANN and SNN predictions, because the leading logit decreases by at most δ and every competitor increases by at most δ. The guarantee concerns prediction agreement with the teacher. Ground-truth accuracy and the behavior of the calibration optimizer are evaluated empirically.

C Experimental Protocol

C.1 Data, checkpoints, and model selection

Each CIFAR dataset contains 50,000 training images and 10,000 official test images. Preprocessing uses training-set channel means and standard deviations. The CIFAR ResNet-18 uses a 3×3, stride-one stem without an initial max-pooling layer. The sequential control is a VGG-16-BN variant with average pooling, global average pooling, and a linear classifier. These architectural choices are fixed before source training and retained during conversion.

The CIFAR source-training protocol uses SGD with momentum 0.9, weight decay 5×10^{-4}, batch size 128, and initial learning rate 0.1, followed by cosine decay over 200 epochs. Training augmentation consists of random cropping and horizontal flipping. A disjoint set of 1,024 training images is reserved for converter selection before custom source training.

ImageNet-1K evaluation uses all 50,000 official validation images. Each ResNet architecture has one fixed public supervised ReLU checkpoint across matched methods. Preprocessing follows that checkpoint's resize, center-crop, and normalization convention. Calibration and selection use disjoint subsets of the training split with deterministic inference transforms. Selection is held out from converter optimization; the public source checkpoint retains its original training history. Within each seed, all matched methods share calibration indices, selection indices, and the ordering of nested calibration subsets.

Calibration uses AdamW (Loshchilov & Hutter, 2019) with the settings in Table 5 and five seeds (0–4) on both CIFAR and ImageNet. Paired comparisons share minibatch and condition schedules. Checkpoint selection minimizes the unregularized objective averaged over all training conditions on $\mathcal{D}_{\text{sel}}$, with fixed loss weights and teacher-generated targets. The selection rule uses no evaluation labels. Achieved density and per-image violation rate characterize the realized activity of budget-conditioned profiles.

C.2 Baseline implementation and fairness

Matched comparisons use a common source checkpoint, encoding scheme, stage mapping, pooling convention, and signed readout.

- **Percentile-IF.** Channel thresholds use the 99.9th percentile, initial voltages equal half the threshold, and current corrections are zero. This configuration is also the pilot for StageCal.
- **MMSE-IF.** For each channel and horizon, the threshold minimizes $\mathbb{E}[(a - Q_{\theta,1/2,T}(a))^2]$ on $\mathcal{D}_{\text{cal}}$ over 64 log-spaced candidates in $[0.1\theta^0, 2\theta^0]$. Initialization remains $v^0 = \theta/2$, with $c = 0$. This baseline isolates scalar quantizer search.
- **Light calibration (matched).** The baseline applies the layer-wise analytic bias correction of Li et al. (2021) after the same MMSE initialization. The backbone remains frozen throughout calibration.
- **FTBC (matched).** Forward temporal bias correction (Wu et al., 2024) operates with the shared encoding, stage mapping, and signed readout. Source kernels remain frozen, and the correction representation retains its temporal indexing.
- **Direct controls.** Each (T, b) profile has independently trainable channel parameters. Thresholds and initial states use the bounds in Eq. (5); current corrections use $c = \kappa\theta \tanh \beta$, with the fixed percentile scale at the readout. Unconstrained parameters start at zero. Direct controls optimize the same objective with the same condition sequence, minibatches, total iterations, and temporal sample exposure as the shared policy. The bias-only variant fixes thresholds and initialization. Independent bounded biases provide a broader correction family than the residual-mixture parameterization.
- **Independent-stage policies.** Each stage receives its own MLP with the same summary features and residual bases. The comparison includes width-64 networks and networks with widths selected to approximate the shared policy's total parameter count. Calibration steps follow the same schedule across variants.

The shared iteration and horizon schedules align temporal sample exposure. Calibration time and peak memory provide separate measures of computational cost. The single-profile direct control allocates the full calibration budget to the target condition, providing a specialized reference for comparison with the policy fitted across multiple conditions. Standard comparisons fit a new policy for each source checkpoint; the transfer experiments explicitly freeze the source policy.

The native-protocol comparison in Table 13 includes QCFS (Bu et al., 2023), SlipReLU (Jiang et al., 2023), Fast-SNN (Hu et al., 2023), and error-compensation learning (Liu et al., 2025). Each row retains the method-specific source ANN, activation and neuron formulation, and fitting recipe. Accuracy differences are computed against the corresponding source ANN. This comparison characterizes complete native pipelines, while the matched comparison isolates converter behavior under a common frozen source.

C.3 Metrics and Variability

Evaluation reports top-1 accuracy, ImageNet top-5 accuracy, signed $\Delta\text{Acc} = \text{Acc}_{\text{S}} - \text{Acc}_{\text{A}}$ in percentage points, and teacher–student prediction agreement. Feature alignment is measured by the normalized MSE in Eq. (13). Hidden SynOps, achieved spike density, and the fraction of images with $R > b$ characterize activity and budget compliance. Mean density and per-image violation rate describe different aspects of the activity distribution.

Residual diagnostics use the evaluated network's thresholds and initialization, whereas optimization uses the frozen pilot bases. Signed residual means quantify systematic bias, and mean absolute residuals quantify magnitude. The residual components contain cross terms in squared error, so component MSEs do not form an additive decomposition of total MSE.

Across-seed summaries use the calibration seed as the unit of replication. A paired comparison forms one accuracy difference for each shared seed, preserving the common data subsets and optimization schedules. Reported seed variability describes calibration randomness on the fixed evaluation set. Individual evaluation images contribute to accuracy estimates rather than the number of independent calibration runs.

C.4 Ablations, sensitivity, and transfer

Default ablations use CIFAR-100/ResNet-18 with $N = 1{,}024$ calibration images and the common checkpoint-selection rule. Removing a head fixes its output at the pilot value. Removing a residual basis sets that hidden-stage mixture contribution to zero without renormalizing the remaining α coefficients, while preserving the readout correction. Removing readout correction sets $c_o = 0$, and the readout-only control fixes all hidden operating parameters at their pilot values. Loss ablations retain the original weights of the remaining terms. Sharing controls include a summary-free lookup with five unconstrained outputs per stage and operating condition, shared across the stage's channels, and a shared policy without the depth feature.

Data efficiency is evaluated at six calibration-set sizes:

$$N \in \{64, 128, 256, 512, 1024, 2048\}.$$

The subsets are nested, with a fixed $1{,}024$-image selection set and a fixed number of optimization steps. This protocol varies calibration-data volume while preserving the optimization-step budget and selection data.

Horizon transfer fits a separate policy on $T \in \{2, 4, 8\}$ at $b = 1$ and evaluates $T \in \{1, 3, 6, 16\}$. At each target horizon, the policy weights remain frozen and a new pilot pass forms the target-dependent residual input $g(T)$ from the calibration data. Re-export time includes this pilot pass and policy evaluation. The nearest trained profile identifies the neighboring deployment condition, and target-specific refitting provides the adapted reference. Budget evaluation measures accuracy and activity across the reported (T, b) profiles.

Cross-architecture transfer freezes a policy fitted on CIFAR-100/ResNet-18 and applies it to CIFAR-100/ResNet-34 or the specified VGG variant. Target-network summaries and pilot residuals supply the new policy inputs, with zero target-gradient updates in the frozen-transfer condition. Target-specific refitting provides the reference under the same dataset and label space.

Robustness evaluation uses clean CIFAR-100 for both calibration and selection. CIFAR-100-C accuracy and teacher agreement are averaged over 15 corruption types and five severity levels. Clean accuracy and worst-type accuracy accompany the aggregate corrupted accuracy. The exported profiles remain fixed across corrupted evaluation images.

D Resource Accounting and Export Validation

Operations. Hidden SynOps count emitted spike events multiplied by their realized outgoing fan-out, including border effects and residual projection edges. Ordinary binary hidden edges define the count associated with Eq. (15); cumulative max-pooling routes have a distinct event representation. Dense MACs cover analog input processing and the signed linear readout for the converted SNN, and dense arithmetic throughout the ANN reference. Cached and repeated operations contribute according to their execution multiplicity. The exported representation consists of fixed thresholds, membrane initializations, and corrections, with the equivalent bias-folded and threshold-rescaled forms described above. Operation counts quantify algorithmic arithmetic, while measured latency characterizes execution under the selected backend.

Timing and memory. The timing protocol uses at least 50 warm-up batches followed by 200 timed batches, with device synchronization at measurement boundaries. Comparisons use common device, backend, precision, and timing boundaries. The per-image inference interval starts with the input already on the device and includes membrane reset, analog input processing, all T simulation steps, pooling and routing, and the signed readout. Reported latency excludes host preprocessing and host-to-device transfer. The deployment comparison reports median and 95th-percentile latency at batch size one. Peak allocated device memory characterizes calibration memory, and calibration time covers summary extraction, pilot simulation, method-specific search, optimization, checkpoint selection, and export.

Energy. We report analytical estimates of arithmetic energy per image using

$$E_{\text{est}} = N_{\text{MAC}} E_{\text{MAC}} + N_{\text{AC}} E_{\text{AC}}, \tag{22}$$

with $E_{\mathrm{MAC}} = 4.6\,\mathrm{pJ}$ and $E_{\mathrm{AC}} = 0.9\,\mathrm{pJ}$ under the 45 nm CMOS arithmetic model. For converted SNNs, N_{MAC} includes analog input processing and the linear readout, while N_{AC} counts hidden-layer spike-triggered synaptic accumulations across all T simulation steps. The ANN reference uses dense MAC counts throughout. All methods use the same coefficients and accounting convention, and results are reported in millijoules per image. These estimates cover the dense and synaptic arithmetic represented by the reported counts. Membrane-state updates, threshold comparisons, resets, pooling and routing overhead, and memory traffic lie outside this accounting boundary. The reported values characterize model-based arithmetic energy rather than measured device or board energy.

Export validation. The export contract preserves source affine parameters up to deterministic batch-normalization folding and the stated scale and bias transformations. Optimizer updates are restricted to the calibration policy. Each inference resets membrane states to their exported initial values, executes exactly T steps, and uses hard binary IF outputs within the parameter bounds. Numerical export equivalence is defined by a maximum absolute difference of 10^{-5} between exported and unexported float32 logits for the same inputs and operating profile. The deployed graph contains zero teacher or calibration-policy calls. Calibration and selection indices are disjoint, and evaluation labels are used only for final accuracy metrics.

E SUPPLEMENTARY RESULTS

The following tables report additional classification results and loss ablations, sensitivity to calibration-set size, accuracy–activity trade-offs, horizon and architecture transfer, corruption robustness, and native-protocol comparisons. Each caption specifies the dataset, architecture, and operating conditions for its comparison. Accuracy differences are expressed in percentage points relative to the corresponding source ANN.

Table 6: Additional accuracy results at $b = 1$. VGG denotes the declared CIFAR average-pooling variant. All converted-SNN entries use $T = 4$. ΔAcc. denotes the signed Top-1 accuracy difference from the corresponding frozen ANN, while Agree. denotes Top-1 prediction agreement with the frozen ANN.

Dataset / model / method	Top-1 (%) ↑	Top-5 (%) ↑	ΔAcc. (pp) ↑	Agree. (%) ↑
C100 / VGG / frozen ANN	74.62	92.38	0.00	100.00
C100 / VGG / Percentile-IF	52.74	77.86	-21.88	60.41
C100 / VGG / Direct	66.83	87.21	-7.79	82.56
C100 / VGG / StageCal	**72.94**	**91.52**	**-1.68**	**93.27**
ImageNet / R18 / Direct	49.38	72.86	-20.38	60.74
ImageNet / R18 / StageCal	**61.06**	**83.42**	**-8.70**	**82.31**
ImageNet / R34 / Direct	46.73	70.38	-26.58	57.62
ImageNet / R34 / StageCal	**68.62**	**88.37**	**-4.69**	**88.46**

Table 7: Loss ablations on CIFAR-100/ResNet-18 at $T = 4$ and $b = 0.25$. Agreement is measured against the source ANN, and R denotes the achieved weighted spike density, for which the target is 0.25. Bold values indicate the strongest budget-compliant configuration.

Loss configuration	Top-1 (%) ↑	Agree. (%) ↑	NMSE ↓	R ($\rightarrow 0.25$)
Full objective	**71.36**	**91.84**	**0.04**	**0.25**
Without feature matching	68.47	86.37	0.10	0.25
Without logit matching	69.62	87.91	0.06	0.25
Without decision preservation	70.18	84.76	0.05	0.25
Without budget penalty	71.84	92.16	0.03	0.39
Without operating regularizer	69.93	88.63	0.08	0.28
Feature matching only	66.79	80.52	0.06	0.34

Table 8: Calibration-data efficiency on CIFAR-100/ResNet-18 at $T = 4$ and $b = 1$. Nested calibration subsets, a fixed number of optimization steps, and an unchanged model-selection set are used for all configurations. Agreement is measured between StageCal and the source ANN.

Calibration N	Direct top-1 (%) ↑	StageCal top-1 (%) ↑	Agreement (%) ↑	Cal. min ↓
64	59.84	**66.92**	85.74	27.92
128	62.17	**68.15**	87.21	28.01
256	64.08	**69.24**	88.74	28.12
512	65.72	**70.13**	89.96	28.28
1024	66.91	**70.89**	91.08	28.49
2048	67.77	**71.36**	91.84	28.74

Table 9: Accuracy–cost trade-off on CIFAR-100/ResNet-18 across simulation horizons and spike-density budgets. Violation denotes the fraction of evaluated images for which $R > b$. Hidden SynOps exclude routed dense operations and neuron-update costs and should therefore be interpreted together with the complete operation breakdown.

T	b	Top-1 (%) ↑	Achieved R	Violation (%) ↓	SynOps (M) ↓
2	0.10	50.84	0.11	22.66	31.29
2	0.25	53.67	0.26	13.67	73.95
2	0.50	55.92	0.47	2.34	133.68
2	1.00	56.28	0.48	0.00	136.52
4	0.10	67.42	0.11	18.75	62.57
4	0.25	71.36	0.25	7.81	142.21
4	0.50	71.42	0.38	0.78	176.15
4	1.00	71.36	0.39	0.00	198.64
8	0.10	70.48	0.10	12.50	113.77
8	0.25	73.62	0.24	4.30	273.04
8	0.50	74.18	0.33	0.39	375.43
8	1.00	74.31	0.34	0.00	386.80

Table 10: Held-out horizon transfer on CIFAR-100/ResNet-18. The shared policy is trained only at $T \in \{2, 4, 8\}$ with $b = 1$. Re-export freezes the learned policy but uses a pilot subset from the new target horizon.

Target T	Nearest profile	Re-export top-1 (%) ↑	Refit top-1 (%) ↑	Re-export min ↓
1	$T = 2$	44.31	45.62	1.08
3	$T = 4$	66.37	66.84	1.36
6	$T = 8$	73.18	73.62	1.94
16	$T = 8$	74.52	75.18	4.72

Table 11: Cross-architecture transfer on CIFAR-100 at $T = 4$ and $b = 1$. The source policy is trained on ResNet-18. Target activation summaries and a pilot subset are used, while the transferred policy weights remain frozen. Calibration costs include target-statistics collection and pilot evaluation.

Target / fitting mode	ANN top-1 (%)	SNN top-1 (%) ↑	Agree. (%) ↑	Cal. min ↓
ResNet-34 / frozen source policy	84.19	80.63	90.84	**2.96**
ResNet-34 / target-specific refit	84.19	**83.75**	**93.58**	38.46
VGG variant / frozen source policy	74.62	71.48	89.72	**2.14**
VGG variant / target-specific refit	74.62	**72.94**	**93.27**	31.62

Table 12: Robustness under clean-only calibration and model selection on CIFAR-100/ResNet-18 at $T = 4$ and $b = 1$. Corrupted accuracy and agreement are averaged over the 15 CIFAR-100-C corruption types and five severity levels. No calibration, selection, or adaptation uses corrupted images. Bold indicates the best converted-SNN result.

Method	Clean top-1 (%) ↑	Corrupted top-1 (%) ↑	Corrupted agreement (%) ↑	Worst-type top-1 (%) ↑
Frozen ANN	77.42	46.18	100.00	24.87
Percentile-IF	38.42	20.84	31.74	6.83
Light calibration	58.73	32.75	53.82	13.74
FTBC (matched)	64.91	37.86	62.45	17.92
Direct operating parameters	67.77	40.92	68.37	20.64
StageCal	**71.36**	**45.37**	**81.76**	**26.31**

Table 13: Native-protocol comparison on CIFAR-100 using ResNet-18 family backbones. Each method retains its native source-ANN, activation, neuron, and fitting recipe. ΔAcc. is the signed difference between the SNN and its method-specific ANN. GPU-hours include method-specific ANN/QAT training, fine-tuning, and calibration, but exclude final evaluation.

Native method	ANN top-1 (%)	SNN top-1 (%) ↑	ΔAcc. (pp) ↑	Total fitting GPU-hours ↓
QCFS ($T = 4$)	78.80	75.67	-3.13	11.84
SlipReLU ($N = 1, c = 0.3, T = 4$)	74.01	74.89	+0.88	13.62
Fast-SNN (2-bit, $T = 3$)	77.86	77.16	-0.70	16.47
Error Compensation Learning ($T = 4$)	79.09	78.81	-0.28	14.28
StageCal ($T = 4, b = 1$)	77.42	79.36	+1.94	0.48