Adaptive Calibration: A Unified Conversion Framework of Spiking Neural Networks

Ziqing Wang[1, 2*], **Yuetong Fang**[1*], **Jiahang Cao**[1], **Hongwei Ren**[1], **Renjing Xu**[1†]
[1]The Hong Kong University of Science and Technology (Guangzhou), China
[2]Northwestern University, USA
ziqingwang2029@u.northwestern.edu,
{yfang870, jcao248, hren066}@connect.hkust-gz.edu.cn, renjingxu@hkust-gz.edu.cn

Abstract

Spiking Neural Networks (SNNs) are seen as an energy-efficient alternative to traditional Artificial Neural Networks (ANNs), but the performance gap remains a challenge. While this gap is narrowing through ANN-to-SNN conversion, substantial computational resources are still needed, and the energy efficiency of converted SNNs cannot be ensured. To address this, we present a unified training-free conversion framework that significantly enhances both the performance and efficiency of converted SNNs. Inspired by the biological nervous system, we propose a novel Adaptive-Firing (AdaFire) Neuron Model, which dynamically adjusts firing patterns across different layers to substantially reduce the *Unevenness Error* - the primary source of error of converted SNNs within limited inference timesteps. We further introduce two efficiency-enhancing techniques: the Sensitivity Spike Compression (SSC) technique for reducing spike operations, and the Input-aware Adaptive Timesteps (IAT) technique for decreasing latency. These methods collectively enable our approach to achieve state-of-the-art performance with significant energy savings of up to **70.1%**, **60.3%**, and **43.1%** on CIFAR-10, CIFAR-100, and ImageNet datasets, respectively. Extensive experiments across 2D, 3D, event-driven classification tasks, object detection, and segmentation tasks, demonstrate the effectiveness of our method in various domains.

Code — https://github.com/bic-L/burst-ann2snn

Introduction

Spiking Neural Networks (SNNs) have gained great attention for their potential to revolutionize the computational efficiency of artificial intelligence systems. Unlike traditional Artificial Neural Networks (ANNs), which rely on the intensity of neuron activations, SNNs utilize the timing of sparse and discrete spikes to encode and process information (Maass 1997). This spike-based computing paradigm is particularly suited to neuromorphic hardware, which utilizes spiking neurons and synapses as fundamental components (Davies et al. 2018, 2021; Akopyan et al. 2015). In SNNs, incoming spikes trigger the retrieval of synaptic weights from memory and generate subsequent spike messages routed to other cores, promoting energy-efficient operations over traditional energy-intensive matrix multiplications. The inherent properties of SNNs, coupled with recent advances in neuromorphic hardware, position them as a promising solution for developing energy-efficient and high-performance artificial intelligence systems. Consequently, recent works spanning classification (Wang et al. 2023; Zhou et al. 2022; Deng et al. 2022; Fang et al. 2025), tracking (Zhang et al. 2022), and image generation (Cao et al. 2024) are striving to combine advanced network architectures (Vaswani et al. 2017; He et al. 2023; Ren et al. 2024) with this spike-driven computing paradigm.

Method	T	Acc. (%)	Energy (mJ)	Training	Time Cost (h)
QCFS [ICLR]	32	68.47	77.41	✓	742.51
FastSNN [TPAMI]	7	72.95	16.93	✓	484.45
Calibration [ICML]	32	62.14	57.13	✗	0.06
Ours	5.72	73.46	22.47	✗	0.09

Table 1: **Comparison of the proposed method versus existing methods on ImageNet (VGG-16).** "T' refers to averaged inference time steps, and "Time Cost' represents the total GPU hours before final inference. Our Adaptive Calibration method achieves competitive accuracy with fewer timesteps and lower energy consumption. Notably, it only requires a short setup time and eliminates the need for re-training.

Despite the significant energy efficiency benefits offered by SNNs, achieving performance on par with ANNs remains a challenge. The main difficulty stems from the non-differentiability of discrete spikes and the complex computational graph due to multi-timestep operations. These factors make training SNNs from scratch both intricate and computationally demanding (Neftci, Mostafa, and Zenke 2019). To address this challenge, ANN-to-SNN conversion techniques have emerged as a promising approach, enabling the direct conversion of pre-trained ANNs into high-performance SNNs. Recent advancements aim to reduce conversion errors by replacing the ReLU activation in ANNs with specially designed quantized functions, following extra training. Typically, the quantized ReLU activation function is employed, as it better mimics spiking neuron dynamics (Stöckl and Maass 2021; Bu et al. 2021a; Ding et al. 2021). However, these re-training-based methods come with certain disadvantages. Firstly, as shown in Tab. 1, these studies require training an intermediate surrogate ANN on top of the original ReLU-

*These authors contributed equally.
†Corresponding Author

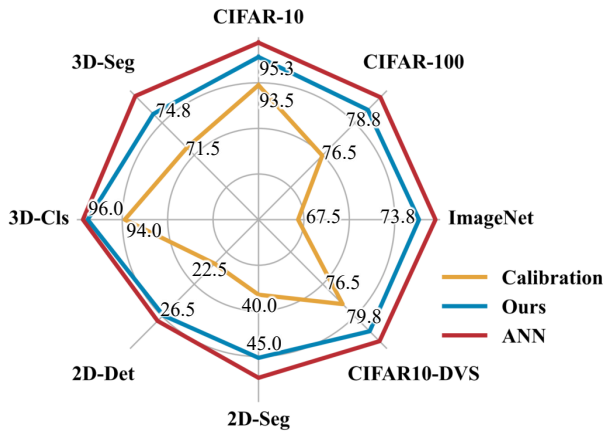


Figure 1: Performance comparison on different tasks.

based ANN, extending the overall training period. Secondly, they compromise the inherent energy efficiency of SNNs by necessitating longer simulation timesteps to minimize conversion errors between quantized ReLU and spiking neurons during inference, thus increasing synaptic operations.

SNN Calibration (Li et al. 2021a) offers a fast, training-free alternative to re-training-based methods for converting advanced ANN architectures into spike-driven models, completing in minutes without extra training, as shown in Table 1. However, the previous study focuses solely on minimizing errors in the network output space, neglecting the differences between ANN and SNN neurons, such as unevenness errors. This results in a larger performance gap, especially under the same inference timestep, limiting practical use in real-time, energy-constrained applications. To address this issue, we draw inspiration from the burst-firing mechanism, widely observed in the human brain. This mechanism features rapid sequences of action potentials that have proven to enhance the reliability of information transmission, in contrast to regular-spiking cells that fire at consistent rates (Connors and Gutnick 1990; Izhikevich et al. 2003; Lisman 1997). Remarkably, this mechanism is well-supported by neuromorphic hardware, such as Intel's Loihi 2 and Synsense's Speck (Orchard et al. 2021; Davies et al. 2018, 2021; Akopyan et al. 2015).

This motivates us to investigate an Adaptive Calibration framework that utilizes adaptable firing patterns of spiking neurons across layers, aiming to simultaneously reduce ANN-to-SNN conversion error, energy, and latency, as shown in Fig. 2. The contributions of our paper are as follows:

- An Adaptive-Firing Neuron (AdaFire) Model integrated into the SNN Calibration process to automatically search for optimum firing patterns, significantly reducing Unevenness Error—the primary error source during conversion within limited timesteps.
- A Sensitivity Spike Compression (SSC) technique that dynamically adjusts thresholds based on layer sensitivity, ensuring energy efficiency of the converted SNNs.
- An Input-aware Adaptive Timesteps (IAT) technique that adjusts timesteps based on input complexity, further decreasing energy consumption and latency.
- Extensive experiments across multiple domains, demonstrating state-of-the-art performance and remarkable energy savings up to **70.1%**, **60.3%**, and **43.1%** for CIFAR-10, CIFAR-100, and ImageNet datasets, respectively.

Related Work

Spiking Neuron Model. In SNNs, inputs are transmitted through the neuronal units, typically the Integrate-and-Fire (IF) spiking neuron in ANN-to-SNN conversions (Ding et al. 2021; Li et al. 2021b; Bu et al. 2021b):

$$u^{(\ell)}(t+1) = v^{(\ell)}(t) + W^{(\ell)} s^{(\ell)}(t) \tag{1}$$

$$v^{(\ell)}(t+1) = u^{(\ell)}(t+1) - s^{(\ell+1)}(t) \tag{2}$$

$$s^{(\ell+1)}(t) = \begin{cases} V_{th}^{(\ell)} & \text{if } u^{(\ell)}(t+1) \geq V_{th}^{(\ell)} \\ 0 & otherwise \end{cases} \tag{3}$$

where $u^{(\ell)}(t+1)$ denotes the membrane potential of neurons before spike generation, $v^{(\ell)}(t+1)$ denotes the membrane potential of neurons in layer ℓ at time step $t+1$, corresponding to the linear transformation matrix W^{ℓ}, the threshold $V_{th}^{(\ell)}$, and binary input $s^{\ell}(t)$ of layer ℓ.

Burst-firing Neurons. Recent studies have integrated burst-firing neurons into SNNs to more accurately mimic the intricate dynamics of biological neural systems (Park et al. 2019; Lan et al. 2023; Li and Zeng 2022). These neurons are supported by neuromorphic hardware, such as Intel's Loihi 2 and Synsense's Speck (Orchard et al. 2021; Davies et al. 2018), enabling better performance. However, existing methods typically employ uniform burst-firing patterns across neural layers, overlooking layer-specific temporal sensitivities and the important balance between performance and energy consumption. To address these challenges, our research introduces the Adaptive Calibration framework that automatically optimizes firing patterns, enhancing performance and energy efficiency within low latency, i.e., simulation timestep, through a training-free methodology.

ANN-to-SNN conversion and SNN Calibration The fundamental principle of ANN-to-SNN conversion is to ensure that the converted SNN closely approximates the input-output function mapping of the original ANN:

$$x^{(\ell)} \approx \overline{s}^{(\ell)} = \frac{1}{T} \sum_{t=0}^{T} s^{(\ell)}(t) \tag{4}$$

where $x^{(\ell)}$ represents the activation input of the ANN model, and $\overline{s}^{(\ell)}$ denotes the averaged binary input over T timesteps in the converted SNN. It is important to note that this approximation becomes valid only as T approaches infinity.

To address this limitation, prior works (Ho and Chang 2021; Ding et al. 2021; Bu et al. 2021a) proposed to replace the *ReLU* activation function in the original ANNs with a trainable $Clip$ function, then find the optimal data-normalization

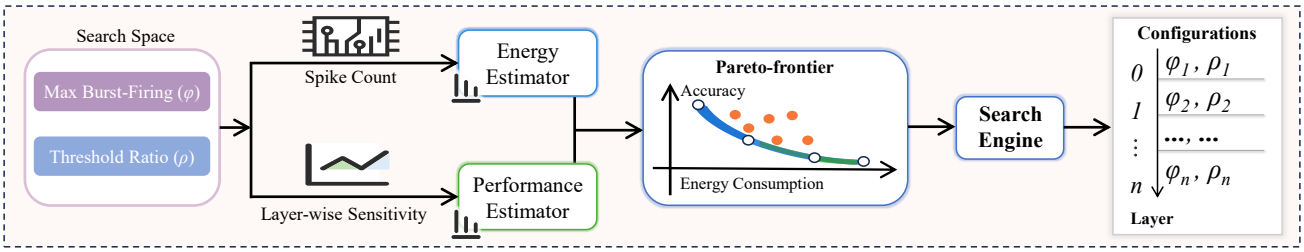


Figure 2: **Optimization Process for Adaptive Calibration.** The process begins within a search space containing candidates for Max Burst-firing Patterns (φ) and Threshold Ratio (ρ). The estimators assess these candidates to evaluate their performance and energy efficiency. Optimum configurations of each layer are then selected using the Pareto-frontier method.

factor through an additional training process to consider both accuracy and latency in the converted SNNs:

$$\begin{aligned} \bar{s}^{(\ell+1)} &= ClipFloor\left(W^{(\ell)}\bar{s}^{(\ell)}, T, V_{th}^{(\ell)}\right) \\ &= \frac{V_{th}^{(\ell)}}{T} Clip\left(\left\lfloor \frac{T}{V_{th}^{(\ell)}} W^{(\ell)}\bar{s}^{(\ell)} \right\rfloor, 0, T\right) \end{aligned} \tag{5}$$

where $\lfloor x \rfloor$ refers to the round down operator. The $Clip$ function limits above but allows below. Although these methods are promising, they often require extensive re-training epochs (hundreds of hours, as shown in Tab 1) to achieve optimal weights and thresholds, as well as prolonged timesteps during inference, making them computationally intensive.

To address the retraining burden, Li et al. (Li et al. 2021a) proposed a layer-wise Calibration algorithm designed to minimize the discrepancy between the output of original ANNs and the converted SNNs. This Spike Calibration method determines the optimal threshold by leveraging Eq. 5:

$$\min_{V_{th}^{(\ell)}} \left(ClipFloor\left(\bar{s}^{(\ell+1)}, T, V_{th}^{(\ell)}\right) - ReLU\left(\bar{s}^{(\ell+1)}\right)\right)^2 \tag{6}$$

Notably, the energy efficiency of converted SNNs and their inference latency compared to directly trained SNNs represent important yet unexplored research opportunities.

Adaptive Calibration

Motivation

The ANN-to-SNN conversion process introduces three main types of errors: clipping error, quantization error, and unevenness error (Bu et al. 2021a; Hao et al. 2023a,b). As illustrated in Fig. 4(a), the unevenness error is the dominant factor during conversion. Therefore, the key challenge lies in reducing this error within limited timesteps.

Unevenness Error. The unevenness error is defined as the difference between the average output of the converted SNN and the output of the source ANN (Hao et al. 2023a):

$$E^{(\ell)} = \bar{s}^{(\ell)} - x^{(\ell)} \tag{7}$$

The conversion error represents the deviation between the expected and actual output patterns in the converted SNN.

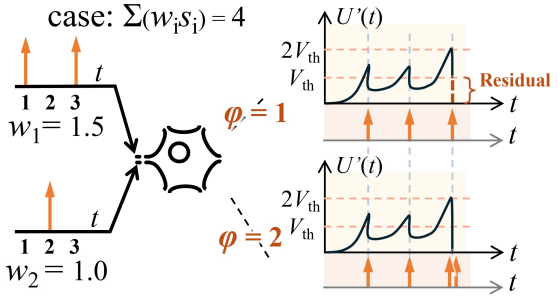


Figure 3: **Burst-firing mechanism in the AdaFire model.** The AdaFire Neuron minimizes this loss by allowing multiple spikes to be generated in rapid succession when the membrane potential exceeds the threshold.

It particularly occurs when the desired firing frequency surpasses the maximum achievable firing rate of neurons within the constrained temporal resolution. Fig. 3 illustrates this concept: consider a neuron receiving three spikes (two weighing 1.5 each and one weighing 1). In an ANN, the neuron's expected output would be 4. However, in an SNN constrained to fire only once per timestep over three timesteps, the actual activation achieves only 3, resulting in a 1-unit unevenness error. Our Adaptive Calibration framework helps to mitigate the residual membrane potential error.

Burst-firing for Error Mitigation. To mitigate this error and enhance information processing efficiency, we can turn to biological systems for inspiration. In the brain, brief bursts of high-frequency firing play a crucial role in enhancing neural communication reliability. This diversity in neuronal responses allows the neocortex to dynamically adjust its information processing based on input characteristics and network demands, optimizing both performance and efficiency (Connors and Gutnick 1990; Izhikevich et al. 2003; Lisman 1997). In Fig.4(b), we provide empirical results to confirm that burst-firing neurons can significantly reduce non-uniform errors. This reduction occurs because burst-firing neurons have more diverse firing patterns and an increased capacity to handle uneven input spikes. Fig.3 exemplifies how allowing a maximum firing time φ of 2 can reduce unevenness and mitigate conversion loss. The burst-firing neuron model can expand the potential range of neuronal activation output

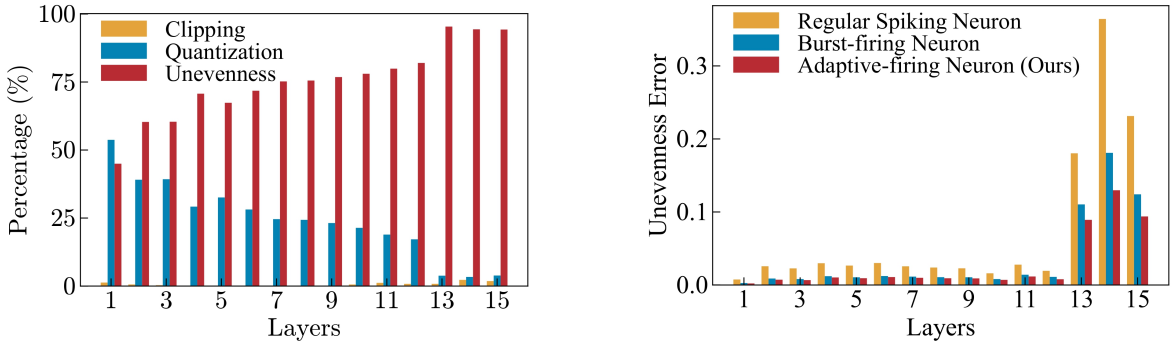


Figure 4: **The unevenness error dominates ANN-to-SNN conversion loss.** (Left) Percentage of three main conversion errors, with the unevenness error dominating. (Right) The adoption of AdaFire Neurons greatly reduces the unevenness error.

$\overline{s}^{(\ell)}$ to $[0, V_{th}^{(\ell-1)} \times \varphi]$. Thus, the relationship between the activation output of ANNs and converted SNNs (Eq. 6) is:

$$\begin{aligned}\overline{s}^{(\ell+1)} &= ClipFloor\left(W^{(\ell)}\overline{s}^{(\ell)}, T, V_{th}^{(\ell)}, \varphi^{(\ell)}\right) \\ &= \frac{V_{th}^{(\ell)}}{T} Clip\left(\left\lfloor \frac{T}{V_{th}^{(\ell)}} W^{(\ell)}\overline{s}^{(\ell)} \right\rfloor, 0, T \times \varphi\right) \end{aligned} \quad (8)$$

While burst-firing neurons can reduce the unevenness error by increasing the neuron's capacity for rapid firing, this approach also allows for the generation of a large number of spikes, potentially leading to significant energy consumption. To address this challenge, we propose an adaptive calibration framework that provides a unified solution to significantly reduce the unevenness error while simultaneously decreasing energy consumption and latency.

Metric Design

Metrics for Performance and Efficiency The success of our Adaptive Calibration depends on precise metrics reflecting the performance and efficiency of the converted SNNs.

(1) Performance Metric: We estimate the performance of SNNs using sensitivity, which is demonstrated inversely related to SNN performance (shown in the Appendix). To quantify layer sensitivity to a parameter k, we employ Kullback-Leibler (KL) divergence (Cai et al. 2020):

$$S_i(k) = \frac{1}{N}\sum_{j=1}^{N} \mathrm{KL}\left(\mathcal{M}\left(\mathrm{ANN_i}; x_j\right), \mathcal{M}\left(\mathrm{SNN_i(k)}; x_j\right)\right) \quad (9)$$

This layer-specific sensitivity metric, $S_i(k)$, quantifies the distributional divergence between ANN and SNN outputs for layer i with respect to parameter k. A reduced $S_i(k)$ value indicates enhanced alignment between SNN and ANN output distributions, suggesting improved conversion fidelity.

(2) Efficiency Metric: The energy efficiency assessment of SNNs follows standardized methodologies (Wang et al. 2023; Ding et al. 2021; Cao, Chen, and Khosla 2015). The total energy consumption is computed as the total spike count during inference and the energy cost per spike event (μ Joules):

$$E = \frac{\text{total spikes}}{1 \times 10^{-3}} \times \mu \quad \text{(in Watts)} \quad (10)$$

Adaptive-Firing Neuron

Observation 1: *The sensitivity to variations in max burst-firing pattern φ differs significantly across network layers.*

Empirical analysis presented in the Appendix demonstrates this statistical correlation. This quantitative evidence suggests that neural layers that exhibit high sensitivity to φ perturbations are associated with an expanded distribution of firing patterns, whereas less sensitive neural layers allow for a more restricted range of firing. As shown in Fig.4(b), the proposed **AdaFire Neuron** further alleviates the unevenness error while optimizing the trade-off between layer-specific firing sensitivity and computational efficiency. This observation, shown by our experimental results in the Appendix, serves as a cornerstone for our novel approach. Building on this observation, we propose that layers with higher sensitivity to φ variations should be given a broader range of firing patterns, and vice versa. Our design of **AdaFire Neuron** aims to balance layer-wise firing sensitivity with energy efficiency.

Layer-Specific Firing Patterns Adaptation. The implementation of uniform burst-firing capabilities, max burst-firing pattern φ across all SNN layers, may not lead to optimal network performance. This perspective is grounded in neurobiological evidence, where neurons demonstrate layer-specific firing patterns tailored to their functional roles within neural circuits (Connors and Gutnick 1990; Izhikevich et al. 2003; Lisman 1997). Our empirical analyses (detailed in the Appendix) suggest the necessity for layer-specific burst-firing patterns φ to further improve conversion efficacy.

Pareto Frontier Driven Search Algorithm. The optimization of layer-specific burst-firing parameters φ presents significant computational challenges. For an SNN model with L layers and n configurations per layer, the solution space is n^L, exhibiting exponential growth with network depth. To tackle this complexity, inspired by (Cai et al. 2020), we adopt the layer-wise independence assumption proposed in (Cai et al. 2020), which posits that each layer's sensitivity to its configuration remains independent of other layers' parameters. This assumption enables the decomposition of the global optimization problem into L independent local optimizations, reducing the computational complexity from $O(n^L)$ to $O(nL)$ and rendering the problem tractable for practical architectures.

Within this framework, we formulate the optimization objective as minimizing the cumulative sensitivity metric S_{sum} subject to an energy constraint E_{target}. The problem is approached through Pareto optimization to establish an optimal trade-off between sensitivity reduction and energy efficiency. The formal optimization formulation is as follows:

$$\min_{\{\varphi_i\}_{i=1}^{L}} S_{\text{sum}} = \sum_{i=1}^{L} S_i(\varphi_i), \quad \sum_{i=1}^{L} E_i \leq E_{\text{target}} \tag{11}$$

where φ_i represents the chosen configuration for the i^{th} layer, and E_i denotes the estimated energy consumption for that layer. This formulation allows us to optimize performance as a sum of individual layer sensitivities, significantly simplifying the search process. Our method demonstrates a promising balance between energy efficiency and model performance with the bridge of layer-wise sensitivity.

Sensitivity Spike Compression

Although AdaFire automatically optimizes layer-wise burst-firing patterns within the desired energy constraints, its flexible spike generation mechanism may incur additional energy overhead. This motivates our investigation into energy-efficient conversion strategies, addressing a critical yet under-explored aspect of ANN-to-SNN conversion methodology.

Observation 2: *The sensitivity to variations in threshold ratio ρ varies distinctly across network layers.*

Leveraging this insight, we optimize network efficiency through strategic spike inhibition. By assigning higher ρ values to less sensitive layers, we achieve a substantial reduction in the overall spiking activity of the network.

Adaptive Threshold. As depicted in Fig. 5, the temporal regularity of neuronal spiking patterns enables effective spike sequence compression through equivalent representation: consecutive spikes with fixed inter-spike intervals can be mathematically reconstructed from a single spike carrying doubled amplitude while preserving complete temporal information. This can be formulated as:

$$V_{th}^{(\ell)} = \rho^{(\ell)} \cdot v_{th}^{(\ell)} \tag{12}$$

where $\rho^{(\ell)}$ refers to the threshold amplification ratio and $v_{th}^{(\ell)}$ signifies the initial threshold of layer ℓ. The subsequent spike output of an IF neuron can be described by:

$$s^{(\ell+1)}(t) = \begin{cases} \rho^{(\ell)} \cdot V_{th}^{(\ell)} & \text{if } u^{(\ell)}(t+1) \geq \rho^{(\ell)} \cdot V_{th}^{(\ell)} \\ 0 & otherwise \end{cases} \tag{13}$$

Subsequently, the updated firing rate for SNN output is:

$$r^{(\ell+1)} = \sum_{i=1}^{n} W_i^{(\ell)} \frac{\sum_{t=1}^{T} s_i^{(\ell)}(t) \cdot \rho^{(\ell)}}{T} \tag{14}$$

This methodology achieves a significant reduction in spike activity while guaranteeing an information throughput amplification factor of $\rho^{(\ell)}$, per neural unit, thereby maintaining consistent inter-layer information transmission.

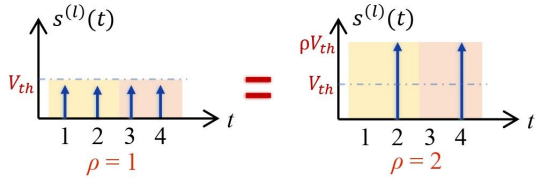


Figure 5: **Spike Compression Mechanism.** Our approach enables the compression of regular spikes.

Adaptive Threshold Search Algorithm. Conventional threshold-based compression strategies often incur significant performance degradation due to their inability to handle non-stationary spatiotemporal patterns in spike trains. To mitigate this, we propose the SSC method that quantifies the sensitivity of output distributions to the threshold ratio (ρ) variations. With insight from Observation 2, we establish layer-specific optimization objectives to determine the optimal threshold ratio. The goal is formulated as follows:

$$\min_{\{\rho_i\}_{i=1}^{L}} E_{\text{sum}} = \sum_{i=1}^{L} E_i(\rho_i), \quad \sum_{i=1}^{L} S_i \leq S_{\text{target}}. \tag{15}$$

Input-aware Adaptive Timesteps

To further enhance the efficiency of converted SNNs, we focus on adapting the number of timesteps T, which conventionally remains fixed across all inputs. However, this static approach fails to capitalize on the potential benefits of dynamically adjusting timesteps to accommodate the unique characteristics of each input image. Recent studies have highlighted the capacity of SNNs to adapt timesteps dynamically based on individual input features (Li et al. 2024).

Entropy as a Confidence Measure. Inspired by (Teerapittayanon, McDanel, and Kung 2016; Guo et al. 2017), we employ entropy as a confidence measure for predictions at each timestep. Formally, the entropy $H(p)$ is defined as:

$$H(p) = \sum_{y \in \mathcal{Y}} p_y \log p_y, \tag{16}$$

where p_y represents the probability of label y.

Dynamic Timestep Adjustment Mechanism. We adopt a confidence level-based mechanism, with a predefined boundary α, to dynamically determine the required inference timestep. During the inference process, our SNN exits when the confidence score exceeds the boundary α, thus optimizing the balance between accuracy and latency. Our analysis shown in the Appendix indicates non-uniform accuracy contributions across timesteps, in contrast to conventional approaches that adopt a single fixed boundary α (Li et al. 2024). This observation motivates the formulation of timestep-dependent confidence thresholds:

$$\alpha_t = \alpha_{base} + \beta e^{-\frac{\bar{E}_t - \bar{E}_{min}}{\delta}} \tag{17}$$

where α_{base} is the base boundary, β is the scaling factor, δ represents the decay constant, $\bar{E}_t$ denotes the average entropy of the network's output distribution associated with each timestep t, and $\bar{E}_{min}$ is the minimum value within $\bar{E}_t$. This

Arch.	Method	ANN	T=8	T=16	T=32	T=64	Rt.
VGG-16	OPT (Deng and Gu 2021)[ICLR]	75.36	-	-	0.11	0.12	✓
	SNM (Wang et al. 2022)[IJCAI]	73.18	-	-	64.78	71.50	✓
	QCFS (Bu et al. 2021a)[ICLR]	74.39	-	50.97	68.47	72.85	✓
	SRP (Hao et al. 2023a)[AAAI]	74.29	68.37	69.13	69.35	69.43	✓
	Calibration (Li et al. 2021a)[ICML]	75.36	25.33	43.99	62.14	65.56	✗
	AdaFire (Ours)	75.36	73.53	74.25	74.98	75.22	✗
ResNet-34	OPT (Deng and Gu 2021)[ICLR]	75.66	-	-	0.11	0.12	✓
	QCFS (Bu et al. 2021a)[ICLR]	74.32	-	-	69.37	72.35	✓
	SRP (Hao et al. 2023a)[AAAI]	74.23	67.62	68.02	68.40	68.61	✓
	Calibration (Li et al. 2021a)[ICML]	75.66	0.25	34.91	61.43	69.53	✗
	AdaFire (Ours)	75.66	72.96	73.85	75.04	75.38	✗
ViT	Calibration (Li et al. 2021a)[ICML]	79.36	0.34	3.58	38.36	60.45	✗
	AdaFire (Ours)	79.36	**68.08**	**74.22**	**76.36**	**77.09**	✗

Table 2: Performance comparison between the proposed model and the state-of-the-art models on the ImageNet dataset. Rt. represents the need for re-training.

formulation allows for a dynamic boundary α adjustment: higher average entropy at a given timestep, indicating lower confidence in the output, warrants a higher simulation time to ensure accurate inference. The detailed methodology and pseudo-code are provided in the Appendix.

Experiment

We thoroughly evaluate our adaptive calibration framework on multiple benchmarks, including tasks in 2D and 3D classification, event-driven classification, object detection, and segmentation. Detailed results and additional experiments (3D tasks and segmentation) are provided in the Appendix.

Effectiveness of Adaptive-Firing Neuron Model

Performance on Static Classification. We evaluated the benefits of Adafire Neuron using the ImageNet dataset (Deng et al. 2009). Tab. 2 shows that AdaFire maintains high accuracy with fewer timesteps compared to leading conversion methods. Notably, our method achieves these results **without requiring additional training**. As shown in Tab. 1, our method requires only 0.09 hours of setup time, in stark contrast to methods like QCFS (Bu et al. 2021a), which demands 742 hours for additional training. This significant reduction in setup time translates to improved practicality and faster deployment. For a fair comparison, we evaluated our model at $T = 8$ against competitors at $T = 32$, equalizing energy consumption by setting φ to 4. Results indicate that AdaFire exceeds the Calibration base framework by **11.39%**, and outperforms QCFS (Bu et al. 2021a) and SNM (Wang et al. 2022) by **5.06%** and **8.75%**, respectively, on the VGG-16 architecture. In addition, we further evaluate the versatility of our method on other advanced models, like Vision Transformer (Dosovitskiy et al. 2020). Our model achieves better performance with limited time steps and supports a wide range of network architectures.

Performance on Event-driven Classification. Tab. 3 presents our evaluation across various neuromorphic datasets, such as CIFAR10-DVS and N-Caltech101, which were derived from static datasets using event-based cameras. AdaFire consistently outperformed leading SNN models, including PLIF (Fang et al. 2021), by **6.45%** using only 8 timesteps. More results are shown in the Appendix.

Dataset	Model	T	Acc. (%)
CIFAR10-DVS	PLIF (Fang et al. 2021)[ICCV]	20	74.80
	Dspkie (Li et al. 2021c)[NeurIPS]	10	75.40
	DSR (Meng et al. 2022)[CVPR]	10	77.30
	AdaFire (Ours)	**8**	**81.25**
N-Caltech101	SALT (Kim and Panda 2021)[NN]	20	55.00
	NDA (Li et al. 2022)[ECCV]	10	83.70
	AdaFire (Ours)	**8**	**85.21**

Table 3: Performance comparison between the proposed model and the state-of-the-art models on different neuromorphic datasets.

Dataset	Method	ANN	T	mAP
VOC	Spiking-YOLO (Kim et al. 2020b)[AAAI]	53.01	8000	51.83
	B-Spiking-YOLO (Kim et al. 2020a)[Access]	53.01	5000	51.44
	Calibration (Li et al. 2021a)[ICML]	54.34	128	47.15
	AdaFire (Ours)	54.34	16	**51.91**
COCO	Spiking-YOLO (Kim et al. 2020b)[AAAI]	26.24	8000	25.66
	B-Spiking-YOLO (Kim et al. 2020a)[Access]	26.24	5000	25.78
	Calibration (Li et al. 2021a)[ICML]	26.78	128	20.12
	AdaFire (Ours)	26.78	16	**26.13**

Table 4: Performance comparison for object detection on PASCAL VOC 2012 and MS COCO 2017 datasets. mAP represents the mean Average Precision.

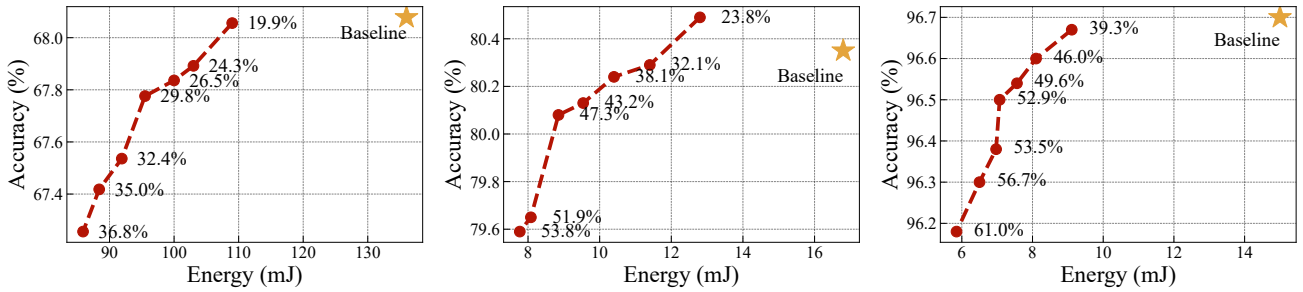
Performance on Object Detection. We evaluated our method on the PASCAL VOC 2012 and MS COCO 2017 datasets, benchmarked against established models on the Tiny-YOLO model. Tab. 4 shows our method's substantial efficiency improvement on COCO, where it achieved a mAP of **26.13%** with only 16 timesteps, compared to Spiking-YOLO's (Kim et al. 2020b) 25.66% mAP at 8000 timesteps. This represents a **500**$\times$ speed-up, highlighting our method's potential for real-time applications.

Effectiveness of Sensitivity Spike Compression

We evaluate the SSC technique on CIFAR-10, CIFAR-100, and ImageNet datasets, tuning S_{target} as per Eq 15 to balance energy consumption and performance. The method of calculating the theoretical energy is shown in the Appendix. Results in Fig. 6 show a **61.0%** energy reduction on CIFAR-10 with minimal accuracy loss (0.5%). On ImageNet, SSC achieved a **32.4%** energy saving with a comparable accuracy decrease. These results validate SSC's efficiency in saving energy without performance degradation.

Effectiveness of Input-aware Adaptive Timesteps

As illustrated in Fig. 7(a), our IAT method dynamically adjusts the confidence threshold α to optimize processing time. This approach reduces latency and increases energy efficiency by allowing early exits for simpler images and extended processing for complex ones. Fig. 7(b) shows a **2.4-fold** increase in speed and a **2.7-fold** reduction in energy consumption, with a performance improvement of **1.1%** over the baseline.



(a) ImageNet (b) CIFAR-100 (c) CIFAR-10

Figure 6: **Effectiveness of Sensitivity Spike Compression (SSC)**. The baseline is the results without using the SSC.

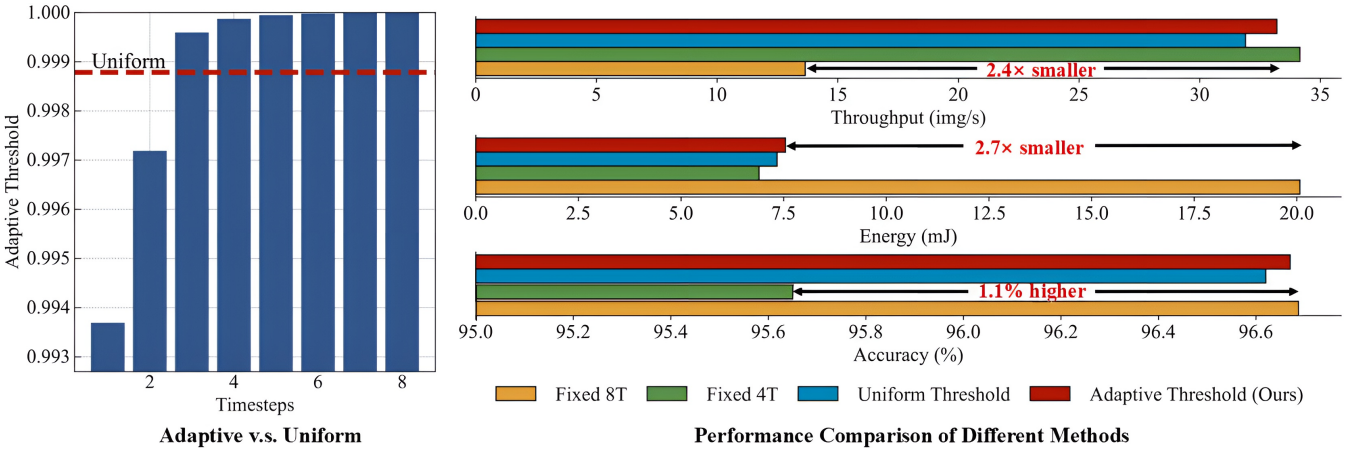


Figure 7: Effectiveness of Input-aware Adaptive Timesteps Technique.

AdaFire	SSC	IAT	CIFAR-10 (ResNet-20)		CIFAR-100 (ResNet-20)		ImageNet (ResNet-18)	
			Accuracy (%)	Energy (mJ)	Accuracy (%)	Energy (mJ)	Accuracy (%)	Energy (mJ)
			96.34	14.86 (0)	79.90	16.83 (0)	56.74	162.56 (0)
✓			**96.69**	20.49 (+37.88%)	**80.64**	21.44 (+27.37%)	**68.45**	169.52 (+0.04%)
✓	✓		96.5	7.71 (-48.12%)	80.37	10.00 (-40.58%)	68.32	120.32 (-25.98%)
✓		✓	96.67	7.06 (-52.48%)	80.55	12.16 (-27.75%))	68.39	105.26 (-35.25%)
✓	✓	✓	95.47	**4.44 (-70.12%)**	80.00	**6.69 (-60.25%))**	68.27	**92.50 (-43.10%)**

Table 5: Ablation Study of Different Techniques of our Adaptive Calibration. T is set to 8 by default.

These outcomes highlight the effectiveness of IAT in lowering latency and energy costs without compromising accuracy.

Ablation Study

We evaluate three proposed techniques—AdaFire, SSC, and IAT on the CIFAR-10, CIFAR-100, and ImageNet datasets. Our results reveal that the AdaFire significantly boosts the accuracy of SNNs. Concurrently, the SSC and IAT techniques contribute to a substantial reduction in energy consumption. Remarkably, the synergistic application of three techniques leads to a groundbreaking **70.12%** energy reduction and a **0.13%** accuracy enhancement for the CIFAR-10 dataset. For the more challenging ImageNet dataset, the combined implementation achieves a **43.10%** decrease in energy consumption while simultaneously enhancing accuracy by **11.53%**.

These results underscore the efficacy of our proposed conversion framework as a unified solution capable of both improving performance and efficiency.

Conclusion

In our paper, we propose a unified training-free ANN-to-SNN conversion framework optimized for both performance and efficiency. We introduce the Adaptive-firing (AdaFire) Neuron Model , which automatically searches for optimum burst-firing patterns of each layer, significantly improving the SNN performance at low timesteps. Moreover, to improve efficiency, we propose a Sensitivity Spike Compression (SSC) technique and an Input-aware Adaptive Timesteps (IAT) technique, reducing both the energy consumption and latency during the conversion process.

Acknowledgments

This work is supported by the Guangzhou-HKUST(GZ) Joint Funding Program (Grant No. 2023A03J0682) and partially supported by the collaborative project with Brain Mind Innovation, inc.

References

Akopyan, F.; Sawada, J.; Cassidy, A.; Alvarez-Icaza, R.; Arthur, J.; Merolla, P.; Imam, N.; Nakamura, Y.; Datta, P.; and Nam, G.-J. 2015. Truenorth: Design and Tool Flow of a 65 Mw 1 Million Neuron Programmable Neurosynaptic Chip. *IEEE transactions on computer-aided design of integrated circuits and systems*, 34(10): 1537–1557.

Bu, T.; Fang, W.; Ding, J.; Dai, P.; Yu, Z.; and Huang, T. 2021a. Optimal ANN-SNN Conversion for High-accuracy and Ultra-low-latency Spiking Neural Networks. In *International Conference on Learning Representations*.

Bu, T.; Fang, W.; Ding, J.; Dai, P.; Yu, Z.; and Huang, T. 2021b. Optimal ANN-SNN Conversion for High-accuracy and Ultra-low-latency Spiking Neural Networks. In *International Conference on Learning Representations*.

Cai, Y.; Yao, Z.; Dong, Z.; Gholami, A.; Mahoney, M. W.; and Keutzer, K. 2020. Zeroq: A Novel Zero Shot Quantization Framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13169–13178.

Cao, J.; Wang, Z.; Guo, H.; Cheng, H.; Zhang, Q.; and Xu, R. 2024. Spiking denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 4912–4921.

Cao, Y.; Chen, Y.; and Khosla, D. 2015. Spiking deep convolutional neural networks for energy-efficient object recognition. *International Journal of Computer Vision*, 113: 54–66.

Connors, B. W.; and Gutnick, M. J. 1990. Intrinsic Firing Patterns of Diverse Neocortical Neurons. *Trends in neurosciences*, 13(3): 99–104.

Davies, M.; Srinivasa, N.; Lin, T.-H.; Chinya, G.; Cao, Y.; Choday, S. H.; Dimou, G.; Joshi, P.; Imam, N.; and Jain, S. 2018. Loihi: A Neuromorphic Manycore Processor with on-Chip Learning. *Ieee Micro*, 38(1): 82–99.

Davies, M.; Wild, A.; Orchard, G.; Sandamirskaya, Y.; Guerra, G. A. F.; Joshi, P.; Plank, P.; and Risbud, S. R. 2021. Advancing neuromorphic computing with loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5): 911–934.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A Large-Scale Hierarchical Image Database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. Ieee.

Deng, S.; and Gu, S. 2021. Optimal Conversion of Conventional Artificial Neural Networks to Spiking Neural Networks. *arXiv preprint arXiv:2103.00476*.

Deng, S.; Li, Y.; Zhang, S.; and Gu, S. 2022. Temporal Efficient Training of Spiking Neural Network via Gradient Re-weighting. *arXiv preprint arXiv:2202.11946*.

Ding, J.; Yu, Z.; Tian, Y.; and Huang, T. 2021. Optimal Ann-Snn Conversion for Fast and Accurate Inference in Deep Spiking Neural Networks. *arXiv preprint arXiv:2105.11654*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fang, W.; Yu, Z.; Chen, Y.; Masquelier, T.; Huang, T.; and Tian, Y. 2021. Incorporating Learnable Membrane Time Constant to Enhance Learning of Spiking Neural Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2661–2671.

Fang, Y.; Wang, Z.; Zhang, L.; Cao, J.; Chen, H.; and Xu, R. 2025. Spiking wavelet transformer. In *European Conference on Computer Vision*, 19–37. Springer.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On calibration of modern neural networks. In *International conference on machine learning*, 1321–1330. PMLR.

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

Hao, Z.; Ding, J.; Bu, T.; Huang, T.; and Yu, Z. 2023b. Bridging the Gap between ANNs and SNNs by Calibrating Offset Spikes. In *The Eleventh International Conference on Learning Representations*.

He, C.; Fang, C.; Zhang, Y.; Ye, T.; Li, K.; Tang, L.; Guo, Z.; Li, X.; and Farsiu, S. 2023. Reti-diff: Illumination degradation image restoration with retinex-based latent diffusion model. *arXiv preprint arXiv:2311.11638*.

Ho, N.-D.; and Chang, I.-J. 2021. TCL: An ANN-to-SNN Conversion with Trainable Clipping Layers. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*, 793–798. IEEE.

Izhikevich, E. M.; Desai, N. S.; Walcott, E. C.; and Hoppensteadt, F. C. 2003. Bursts as a Unit of Neural Information: Selective Communication via Resonance. *Trends in neurosciences*, 26(3): 161–167.

Kim, S.; Park, S.; Na, B.; Kim, J.; and Yoon, S. 2020a. Towards Fast and Accurate Object Detection in Bio-Inspired Spiking Neural Networks through Bayesian Optimization. *IEEE Access*, 9: 2633–2643.

Kim, S.; Park, S.; Na, B.; and Yoon, S. 2020b. Spiking-Yolo: Spiking Neural Network for Energy-Efficient Object Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 11270–11277.

Kim, Y.; and Panda, P. 2021. Optimizing Deeper Spiking Neural Networks for Dynamic Vision Sensing. *Neural Networks*, 144: 686–698.

Lan, Y.; Zhang, Y.; Ma, X.; Qu, Y.; and Fu, Y. 2023. Efficient converted spiking neural network for 3d and 2d classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9211–9220.

Li, Y.; Deng, S.; Dong, X.; Gong, R.; and Gu, S. 2021a. A Free Lunch from ANN: Towards Efficient, Accurate Spiking

Neural Networks Calibration. In *International Conference on Machine Learning*, 6316–6325. PMLR.

Li, Y.; Deng, S.; Dong, X.; Gong, R.; and Gu, S. 2021b. A Free Lunch from ANN: Towards Efficient, Accurate Spiking Neural Networks Calibration. In *International Conference on Machine Learning*, 6316–6325. PMLR.

Li, Y.; Geller, T.; Kim, Y.; and Panda, P. 2024. SEENN: Towards Temporal Spiking Early Exit Neural Networks. *Advances in Neural Information Processing Systems*, 36.

Li, Y.; Guo, Y.; Zhang, S.; Deng, S.; Hai, Y.; and Gu, S. 2021c. Differentiable Spike: Rethinking Gradient-Descent for Training Spiking Neural Networks. *Advances in Neural Information Processing Systems*, 34: 23426–23439.

Li, Y.; Kim, Y.; Park, H.; Geller, T.; and Panda, P. 2022. Neuromorphic Data Augmentation for Training Spiking Neural Networks. *arXiv preprint arXiv:2203.06145*.

Li, Y.; and Zeng, Y. 2022. Efficient and Accurate Conversion of Spiking Neural Network with Burst Spikes. *arXiv preprint arXiv:2204.13271*.

Lisman, J. E. 1997. Bursts as a Unit of Neural Information: Making Unreliable Synapses Reliable. *Trends in neurosciences*, 20(1): 38–43.

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

Meng, Q.; Xiao, M.; Yan, S.; Wang, Y.; Lin, Z.; and Luo, Z.-Q. 2022. Training High-Performance Low-Latency Spiking Neural Networks by Differentiation on Spike Representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12444–12453.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-Based Optimization to Spiking Neural Networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Orchard, G.; Frady, E. P.; Rubin, D. B. D.; Sanborn, S.; Shrestha, S. B.; Sommer, F. T.; and Davies, M. 2021. Efficient neuromorphic signal processing with loihi 2. In *2021 IEEE Workshop on Signal Processing Systems (SiPS)*, 254–259. IEEE.

Park, S.; Kim, S.; Choe, H.; and Yoon, S. 2019. Fast and efficient information transmission with burst spikes in deep spiking neural networks. In *Proceedings of the 56th Annual Design Automation Conference 2019*, 1–6.

Ren, H.; Zhou, Y.; Zhu, J.; Fu, H.; Huang, Y.; Lin, X.; Fang, Y.; Ma, F.; Yu, H.; and Cheng, B. 2024. Rethinking Efficient and Effective Point-based Networks for Event Camera Classification and Regression: EventMamba. *arXiv preprint arXiv:2405.06116*.

Stöckl, C.; and Maass, W. 2021. Optimized Spiking Neurons Can Classify Images with High Accuracy through Temporal Coding with Two Spikes. *Nature Machine Intelligence*, 3(3): 230–238.

Teerapittayanon, S.; McDanel, B.; and Kung, H.-T. 2016. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd international conference on pattern recognition (ICPR)*, 2464–2469. IEEE.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, \.; and Polosukhin, I. 2017. Attention Is All You Need. *Advances in neural information processing systems*, 30.

Wang, Y.; Zhang, M.; Chen, Y.; and Qu, H. 2022. Signed Neuron with Memory: Towards Simple, Accurate and High-Efficient Ann-Snn Conversion. In *International Joint Conference on Artificial Intelligence*.

Wang, Z.; Fang, Y.; Cao, J.; Zhang, Q.; Wang, Z.; and Xu, R. 2023. Masked spiking transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1761–1771.

Zhang, J.; Dong, B.; Zhang, H.; Ding, J.; Heide, F.; Yin, B.; and Yang, X. 2022. Spiking Transformers for Event-Based Single Object Tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8801–8810.

Zhou, Z.; Zhu, Y.; He, C.; Wang, Y.; Yan, S.; Tian, Y.; and Yuan, L. 2022. Spikformer: When Spiking Neural Network Meets Transformer. *arXiv preprint arXiv:2209.15425*.