ReverB-SNN: Reversing Bit of the Weight and Activation for Spiking Neural Networks

Yufei Guo Yuhan Zhang Zhou Jie Xiaode Liu Xin Tong Yuanpei Chen Weihang Peng Zhe Ma
英寸

Abstract

脉冲神经网络（SNN）是一种受生物启发的神经网络结构，最近受到了广泛关注。SNNs 利用二进制脉冲激活实现高效的信息传递，用加法代替乘法，从而提高了能源效率。然而，二进制脉冲激活图通常无法捕捉足够的数据信息，导致准确性降低。为了解决这一挑战，我们提出了一种逆转 SNNs 的权重和激活比特的方法，称为 ReverB-SNN ，该方法受到最近研究结果的启发，强调对激活进行量化比对权重进行量化导致更大的准确性下降。具体来说，我们的方法在 SNNs 中使用实数值的脉冲激活和二进制权重。这保留了标准 SNNs 的事件驱动和无乘法的优点，同时提高了激活的信息容量。此外，我们在二进制权重中引入了一个可训练因子，以在训练过程中自适应地学习合适的权重幅度，从而增加网络容量。为了保持与标准 ReverB-SNN 相似的效率，我们在推理过程中使用重新参数化技术将可训练的二进制权重 SNNs 转回标准形式。大量网络架构和数据集（静态和动态）的实验显示，我们的方法始终优于当前最先进的方法。

1. 介绍

人工神经网络（ANNs）目前广泛应用于各种领域，如目标识别 (He et al., 2016; Ming et al., 2023) 、目标分割 (Ronneberger et al., 2015) 和目标跟踪 (Bewley et al., 2016) 。然而，为了提升性能，网络架构变得越来越复杂 (Huang et al., 2017; Devlin et al., 2018) 。为了解决这种复杂性，提出了几种技术，包括量化 (Gong et al., 2019) 、剪枝 (He et al., 2017) 、知识蒸馏 (Hinton et al., 2015; Polino et al., 2018; Zhang et al., 2022) ，以及尖峰神经网络（SNNs）的出现 (Maass, 1997; Li et al., 2021a; Xiao et al., 2021; Wang et al., 2022; Bohnstingl et al., 2022; Yu et al., 2025; Guo et al., 2025; Yao et al., 2023; Guo et al., 2023a) 。作为下一代神经网络的代表，SNNs 通过尖峰通信模拟类脑的信息处理，减少能源消耗，将权重和激活的乘法转换为加法，从而实现无乘法的推理。此外，它们的事件驱动计算模型在神经形态硬件平台上展示了卓越的能量效率 (Ma et al., 2017; Akopyan et al., 2015; Davies et al., 2018; Pei et al., 2019) 。

然而，已观察到 SNN 的二值化尖峰激活图受限于信息容量的限制，未能在量化过程中充分捕获膜电位信息，从而降低了准确性 (Guo et al., 2022d;a; Wang et al., 2023; Sun et al., 2022) 。为了应对这一问题，一些研究探索了诸如三值尖峰 (Sun et al., 2022) 、整数尖峰 (Wang et al., 2023; Fang et al., 2021b; Feng et al., 2022) ，甚至实值尖峰 (Guo et al., 2024c;d; 2025) 等替代方法，然而这些方法往往因无法将权重和激活的乘法转换为加法而导致增加的能量消耗。

最近的研究 (Gong et al., 2019; Qin et al., 2024) 表明，在人工神经网络中使用低位权重相比低位激活可以实现更高的准确性。受这些发现的启发，本文提出了一种新颖的方法来增强脉冲激活的信息容量，同时保留无乘法和事件驱动的脉冲神经网络（SNN）的优势。具体来说，与传统的二进制脉冲激活方法不同，我们倡导类似于 EGRU (Subramoney et al., 2023) 的实数值脉冲激活以增加信息容量。相应地，我们将实数值权重调整为二进制权重 $\{-1, 1\}$ ，以确保保留无乘法和事件驱动的优势。鉴于二进制权重可能限制网络容量，我们将其扩展为可学习形态 $\{-\alpha, \alpha\}$ ，其中 α 是一个可学习的参数。在推理过程中，我们引入了一种重新参数化技术，将 α 因子整合到激活过程中，从而仍然保留无乘法的推理能力。

我们 SNN 和传统 SNN 之间的区别在图 1 中进行了说明。总而言之，我们的贡献可以总结如下：

- 我们主张通过在脉冲神经网络中使用实值脉冲结合二值权重来增强脉冲激活的信息容量。该方法在引入实值脉冲神经元和二值权重的新范式的同时，保留了标准脉冲神经网络中无需乘法的计算模式和事件驱动的优势。
- 此外，我们提出了一种具有可学习二进制权重和重新参数化技术的变体。在训练期间，权重幅度 α 被学习，而在推断期间，通过重新参数化将该

Intelligent Science & Technology Academy of CASIC, China. Yufei Guo and Yuhan Zhang contributed equally to this study. Correspondence to: Yufei Guo < yfguo@pku.edu.cn >, Zhe Ma < mazhe_thu@163.com >.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

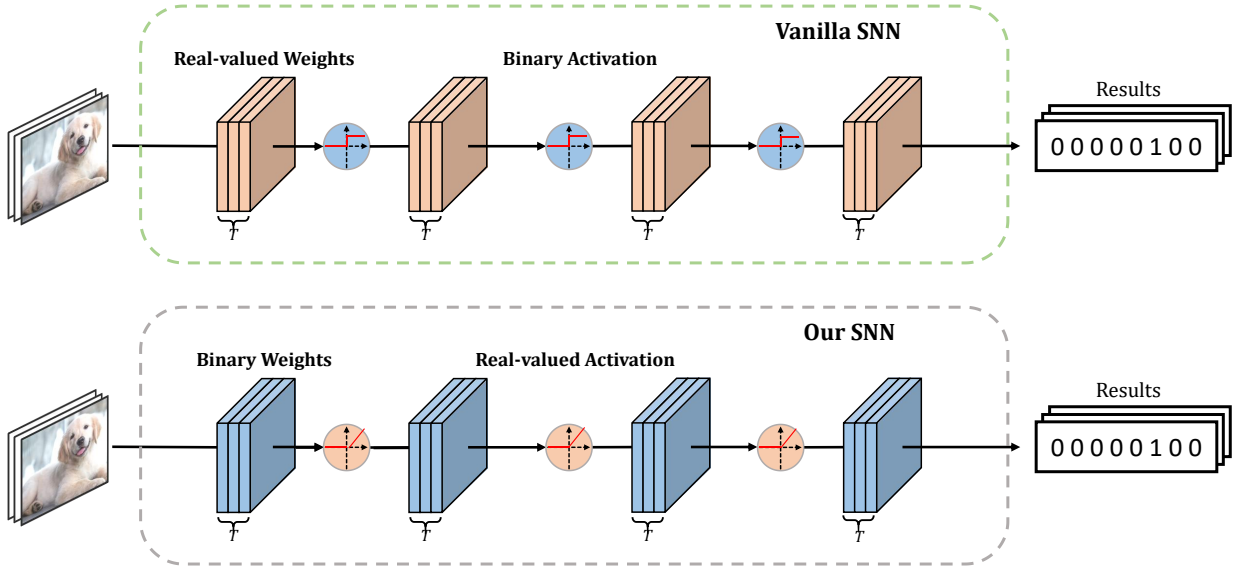

Figure 1. 我们的 SNN 与传统 SNN 的区别。我们的 SNN 与传统 SNN 有显著差异。传统 SNN 使用二进制脉冲，这导致激活信息的显著丢失。相比之下，我们的 SNN 利用实数脉冲和二进制权重，从而增强了神经元的信息容量。此方法保留了事件驱动处理和乘法-加法转换的优点。

幅度折叠到激活中。这确保二进制权重 $\{-\alpha, \alpha\}$ 恢复为标准二进制权重 $\{-1, 1\}$，保持仅加法的优势。

- 我们使用广泛接受的骨干网络对静态数据集（CIFAR-10（Krizhevsky et al., 2010）、CIFAR-100（Krizhevsky et al., 2010）、ImageNet（Deng et al., 2009））和脉冲数据集（CIFAR10-DVS（Li et al., 2017））进行方法评估。结果证明了我们方法的有效性和效率。例如，使用 ResNet34，仅用 4 个时间步长，我们的方法在 ImageNet 上实现了 70.91 % 的 top-1 准确率，比其他最先进的 SNN 模型提高了 3.22 % 。

2. 相关工作

在本节中，我们简要概述了最近在脉冲神经网络（SNN）方面的进展，重点介绍两个关键方面：学习方法和信息损失减少技术。

2.1. 脉冲神经网络的学习方法

主要有两种方法可以实现高性能的深层 SNN。第一种方法是将一个训练良好的人工神经网络（ANN）转换成 SNN，称为 ANN-SNN 转换（Han & Roy, 2020; Kim et al., 2020; Han et al., 2020; Liu et al., 2022; Yu et al., 2021）。此方法通过将预训练的 ANN 的参数映射到其 SNN 对应物上，以对齐 ANN 的激活值与 SNN 的平均发放率。尽管由于与从头训练 SNN 相比资源效率较高而被广泛使用，但 ANN-SNN 转换存在固有的局限性。它受到率编码方案的限制，忽视了 SNN 独有的时间动态特性，限制了其在神经形态数据集上的有效性。此外，要达到与 ANN 相当的准确性通常需要大量时间步，这与 SNN 的低功耗设计意图相悖地增加了能耗。此外，SNN 的准确性通常不及 ANN 的准确性，从而限制了 SNN 的潜力和研究价值。

直接从头开始训练 SNNs，尤其适用于神经形态数据集，因其在减少时间步长方面的效率而受到关注，有时时间步长减少到不到 5。另一种新兴的方法是混合学习，它结合了 ANN-SNN 转换和直接训练方法的优点。这种方法也引起了极大的兴趣。在本文中，我们专注于通过解决信息丢失问题来提升直接训练的 SNNs 的性能，这是现有文献中探索不足的领域。

2.2. 脉冲神经网络信息损失减少方法

一些方法旨在通过改变脉冲激活精度来减轻 SNNs 中激活的信息损失（Guo et al., 2022c;a;b; Wang et al., 2023）。例如，一种通过 $\{0, 1, 2\}$ 脉冲传递信息的三值脉冲神经元在（Sun et al., 2022）中被提出，这提高了信息容量但缺少乘-加变换的优势。然后，一种新方法在此基础上改进，通过使用 $\{-1, 0, 1\}$ 值的三值脉冲，同时保持改进的激活信息容量和乘-加优势（Guo et al., 2024b）。在 MT-SNN（Wang et al., 2023）和 DS-ResNet（Feng et al., 2022）中，为泄漏积分发放（LIF）神经元引入了一种多重门限（MT）算法，允许省略整数脉冲以增强信息传输。SEWNet（Fang et al., 2021b）通过修改快捷模块的位置提出了一种整数脉冲格式。一些方法直接采用实值脉冲以显著提高信息容量（Guo et al., 2024c; 2025）。然而，上述这些工作都是以增加能量消耗为代价的。

本文探索了在使用二值化权重的情况下，采用实值脉

冲激活的同时保持无乘法运算的优势。

3. 方法论

在本节中，我们首先介绍 SNN 的基本原理，以说明其信息处理方法及其在信息损失方面的固有局限性。随后，我们介绍我们的 ReverB-SNN 方法，作为克服这一挑战的解决方案。最后，我们提出一个可学习的二进制权重变体，旨在进一步增强网络容量。

3.1. 初步

受大脑功能启发的尖峰神经元，是 SNNs 中基本且独特的计算单位。它近似模拟生物神经元的行为，其特征在于膜电位与输入电流动态之间的相互作用。在本文中，我们专注于广泛使用的泄漏积分-发火（LIF）神经元模型，其由迭代方程 (Wu et al., 2019) 控制:

$$U_l^t = \tau U_l^{t-1} + \boldsymbol{W}_l O_{l-1}^t, \qquad U_l^t < V_{\text{th}}. \tag{1}$$

其中，U_l^t 代表第 l 层在时间步 t 的膜电位，O_{l-1}^t 表示前一层的尖峰输出，$\boldsymbol{W}_l$ 表示第 l 层的权重矩阵，V_{th} 是发火阈值，τ 是控制膜电位泄漏的时间常数。当膜电位超过发火阈值时，神经元发出尖峰并重置到其静息状态，其特征为:

$$O_l^t = \begin{cases} 1, & \text{if } U_l^t \geq V_{\text{th}} \\ 0, & \text{otherwise} \end{cases}. \tag{2}$$

虽然基于二值尖峰的处理方式能效高，但因信息表示上的局限性，任务性能不尽如人意。这促使我们探索替代方法以增强 SNNs 的信息容量。

脉冲神经网络（SNNs）中的分类器。在一个分类模型中，最终输出通常使用 Softmax 函数进行处理，以预测所需的类别对象。在 SNN 模型的背景下，最近的研究（如 (Guo et al., 2022c;d; Fang et al., 2021c) 中所示）中一种常见的方法是聚合所有时间步长的输出以获得最终输出:

$$O_{\text{out}} = \frac{1}{T}\sum_{t=1}^{T} O_{\text{out}}^t. \tag{3}$$

。随后，使用真实标签和 Softmax(O_{out}) 计算交叉熵损失。

3.2. ReverB：反转权重和激活的位

为了应对激活中的信息损失问题，我们引入了 ReverB-SNN 方法，这一方法的灵感来源于最近的研究，该研究强调激活量化相比权重量化更容易导致准确度下降 (Gong et al., 2019; Qin et al., 2024) 。具体而言，我们采用实值尖峰激活，其中在时刻 t ，l 层的输出尖峰定义如下:

$$O_l^t = \begin{cases} U_l^t, & \text{if } U_l^t \geq V_{\text{th}} \\ 0, & \text{otherwise} \end{cases}. \tag{4}$$

与此同时，为了保持标准 SNN 的无乘法和事件驱动的优势，实值权重被转换为二值权重。因此，膜电位动态更新为:

$$U_l^t = \tau U_l^{t-1} + \boldsymbol{W}_l^b O_{l-1}^t, \qquad U_l^t < V_{\text{th}}, \tag{5}$$

其中 $\boldsymbol{W}^b = \text{sign}(\boldsymbol{W}) = \begin{cases} +1, & \text{if } \boldsymbol{W} \geq 0 \\ -1, & \text{otherwise} \end{cases}$. 这种方法确保激活保持为实值，同时利用二值权重提高计算效率，从而减轻 SNN 中的准确度损失。

事件驱动的优势保持。SNNs 的事件驱动信号处理特性显著增强了能源效率。具体来说，只有当其膜电位超过发放阈值 V_{th} 时，脉冲神经元才会发出信号并启动后续计算；否则，它保持不活动。类似地，我们的实值脉冲神经元也受益于这一事件驱动的特性。只有当其膜电位超过 V_{th} 时，它才会激活并发出实值脉冲以启动计算。

“仅使用加法的优势保留”。脉冲神经网络使用加法而不是乘法的能力显著增强了其能量效率。在一个标准的二进制脉冲神经元中，当一个脉冲被发射时，它传统上会将一个权重 w 与另一个神经元相乘以传递信息，表示为:

$$x = 1 \times w, \tag{6}$$

，其中 $w \in \mathbb{R}$ 。鉴于脉冲振幅为 1，这个乘法就简化为加法:

$$x = 0 + w. \tag{7}$$

。在我们的实值脉冲神经元中，虽然脉冲 o 是实值的，但权重 $w^b \in \mathbb{B}$ 是二进制的，且 o 与 w^b 的乘积可以表示为:

$$x = o \times 1, \text{or}, \text{o} \times -1. \tag{8}$$

。这也可以简化为加法操作:

$$x = 0 + o, \text{or}, 0 - \text{o}. \tag{9}$$

。

总之，我们提出的方法在保持传统 SNNs 的事件驱动和只加法优势的同时，增强了 SNNs 的激活表达能力。

权重二值化的代理梯度。在基础的脉冲神经网络中，脉冲神经元的发射行为是不可微分的，因此许多研究中需要使用代理梯度（SG）方法来解决这个问题 (Rathi & Roy, 2020; Guo et al., 2022c) 。在我们的脉冲神经网络框架中，尽管脉冲神经元的发射活动变得可微分，但权重二值化的过程仍然存在不可微分的挑战。因此，与用于管理发射活动的其他 SG 方法类似，我们采用直接传播估计器（STE）(Rathi & Roy, 2020; Guo et al., 2022c) 梯度来解决这个问题。在数学上，STE 代理梯度定义为:

$$\frac{d\boldsymbol{W}^b}{d\boldsymbol{W}} = \begin{cases} 1, & \text{if } -1 \leq \boldsymbol{W} \leq 1 \\ 0, & \text{otherwise} \end{cases}. \tag{10}$$

这种方法使我们能够在 SNN 框架内有效地处理权重二值化固有的不可微性。

我们方法的训练。在我们的研究中，我们采用空间-时间反向传播（STBP）算法 (Wu et al., 2019) 来有效地训练我们的 SNN 模型。STBP 将 SNN 视为自递归神经网络，便于像卷积神经网络（CNN）中使用的原理一样进行误差反向传播。通过链式法则导出的第 l 层的梯度表示为：

$$\frac{\partial L}{\partial \boldsymbol{W}_l}=\sum_t(\frac{\partial L}{\partial O_l^t}\frac{\partial O_l^t}{\partial U_l^t}+\frac{\partial L}{\partial U_l^{t+1}}\frac{\partial U_l^{t+1}}{\partial U_l^t})\frac{\partial U_l^t}{\partial \boldsymbol{W}_l^b}\frac{\partial \boldsymbol{W}_l^b}{\partial \boldsymbol{W}_l}, \tag{11}$$

，其中 $\frac{\partial \boldsymbol{W}_l^b}{\partial \boldsymbol{W}_l}$ 是第 l 层中权重二值化的替代梯度。该方法使我们能够通过在时间和网络层之间传播误差，利用神经处理中的时间信息和空间信息的优势，有效地训练 SNN。

3.3. 可学习的二值权重变体

正如前面提到的，虽然实值激活函数可以增加信息容量，但二值化权重可能会降低网络容量。为了解决这个问题，我们将二值化权重扩展为可学习的形式，不再局限于 $\{-1,1\}$，而是 $\{-\alpha,\alpha\}$，其中 α 是一个可学习的参数，定义如下：

$$\boldsymbol{W}_{\text{trainable}}^b=\alpha\cdot\text{sign}(\boldsymbol{W})=\begin{cases}+1\cdot\alpha, & \text{if } \boldsymbol{W}\geq 0\\ -1\cdot\alpha, & \text{otherwise}\end{cases}. \tag{12}$$

引入 α 可以让权重适应它们的幅度。这个参数 α 在我们的 SNN 模型中以通道的方式应用。因此，膜电位动态被调整为：

$$U_l^t=\tau U_l^{t-1}+\boldsymbol{W}_{l,\text{trainable}}^b O_{l-1}^t,\qquad U_l^t<V_{\text{th}}. \tag{13}$$

关于梯度，$\boldsymbol{W}_l$ 在层 l 的梯度为：

$$\frac{\partial L}{\partial \boldsymbol{W}_l}=\sum_t(\frac{\partial L}{\partial O_l^t}\frac{\partial O_l^t}{\partial U_l^t}+\frac{\partial L}{\partial U_l^{t+1}}\frac{\partial U_l^{t+1}}{\partial U_l^t})\frac{\partial U_l^t}{\partial \boldsymbol{W}_l^b}\frac{\partial \boldsymbol{W}_l^b}{\partial \boldsymbol{W}_l}. \tag{14}$$

而 α_l 在层 l 的梯度为：

$$\frac{\partial L}{\partial \alpha_l}=\sum_t(\frac{\partial L}{\partial O_l^t}\frac{\partial O_l^t}{\partial U_l^t}+\frac{\partial L}{\partial U_l^{t+1}}\frac{\partial U_l^{t+1}}{\partial U_l^t})\frac{\partial U_l^t}{\partial \alpha_l}. \tag{15}$$

由于在我们的 SNN 中 $\boldsymbol{W}_{\text{trainble}}^b$ 和 O 都是实值，使用可训练的权重引入了挑战，即权重和激活的乘法不能转化为加法，可能会失去 SNN 的计算效率优势。为了解决这个问题，我们提出了一种训练-推理解耦技术。该方法通过重新参数化，在推理阶段将不同幅度的权重转换为标准化的二进制形式，以确保保留无乘法的效率优势。

重参数化技术。为了在推理过程中保持 SNNs 的计算效率，我们提出了一种重参数化技术。显然，方程 16 可以进一步写为

$$U_l^t=\tau U_l^{t-1}+\alpha_l\boldsymbol{W}_l^b O_{l-1}^t,\qquad U_l^t<V_{\text{th}}. \tag{16}$$

Algorithm 1 我们的 SNN 的训练和推理。

训练

Input : An SNN to be trained where the precision of weights and activations was reversed; training dataset; total training iteration: I_{train} .

Output : The trained SNN.

1: for all $i=1,2,\ldots,I_{\text{train}}$ iteration do
2: Get mini-batch training data, $\boldsymbol{x}_{\text{in}}(i)$ and class label, $\boldsymbol{y}(i)$;
3: Feed the $\boldsymbol{x}_{\text{in}}(i)$ into the SNN and calculate the SNN output, $O_{\text{out}}(i)$ by Eq. 3 ;
4: Compute classification loss $L_{\text{CE}}=\mathcal{L}_{\text{CE}}(O_{\text{out}}(i),\boldsymbol{y}(i))$;
5: Calculate the gradient w.r.t. $\boldsymbol{W}$ by Eq. 14 and the gradient w.r.t. α by Eq. 15;
6: Update $\boldsymbol{W}$: $(\boldsymbol{W}\leftarrow\boldsymbol{W}-\eta\frac{\partial L}{\partial \boldsymbol{W}})$ and α : $(\alpha\leftarrow\alpha-\eta\frac{\partial L}{\partial \alpha})$ where η is learning rate.
7: end for

重新参数化

Input : The trained SNN with trainable weights and real-valued spikes ; total layer of SNN: l .

Output : The re-parameterized trained SNN without normalized binary weight and real-valued spikes.

1: for all $i=1,2,\ldots,l$ number do
2: Fold the parameters of α_i into $i-1$ firing function by Eq. 18;
3: end for

推理

Input : The re-parameterized trained SNN; test dataset; total test iteration: I_{test} .

Output : The output.

1: for all $i=1,2,\ldots,I_{\text{test}}$ iteration do
2: Get mini-batch test data, $\boldsymbol{x}_{\text{in}}(i)$ and class label, $\boldsymbol{y}(i)$ in test dataset;
3: Feed the $\boldsymbol{x}_{\text{in}}(i)$ into the reparameterized SNN and calculate the SNN output, $O_{\text{out}}(i)$ by Eq. 3 ;
4: Compare the classification factor $O_{\text{out}}(i)$ and $\boldsymbol{y}(i)$ for classification.
5: end for

。为了在推理过程中有效地将实值权重 W_l^b 转换回二进制，我们将 α 折叠进上一层的输出 O_{l-1}^t 中作为新的输出，定义为 $O_{\text{new},l-1}^t=\alpha_l O_{l-1}^t$ 。这个调整使得方程 16 简化为：

$$U_l^t=\tau U_l^{t-1}+\boldsymbol{W}_l^b O_{\text{new},l-1}^t,\qquad U_l^t<V_{\text{th}}. \tag{17}$$

。因此，实值权重将再次转换为二进制权重。通过这

Dataset	Method	Time-step	Accuracy
CIFAR-10	Vanilla SNN	2	92.80 %
	ReverB	2	94.14 %
	Learnable variant	2	94.45 %
	Vanilla SNN	4	93.85 %
	ReverB	4	94.55 %
	Learnable variant	4	94.96 %
CIFAR-100	Vanilla SNN	2	70.18 %
	ReverB	2	72.54 %
	Learnable variant	2	72.95 %
	Vanilla SNN	4	71.77 %
	ReverB	4	72.93 %
	Learnable variant	4	73.28 %

Table 1. 对 CIFAR 进行三值尖峰消融研究。

种方式，在时间 t 时刻，$l-1$ 层的输出脉冲更新如下：

$$O^t_{\text{new},l-1} = \begin{cases} \alpha U^t_{l-1}, & \text{if } U^t_{l-1} \geq V_{\text{th}} \\ 0, & \text{otherwise} \end{cases}. \tag{18}$$

。因此，权重和激活的乘法可以再次在推理中转换为加法。

总之，通过在训练过程中将一个可学习因子 α 嵌入到权重中，我们增强了网络的能力。在推理过程中，我们从权重中提取该因子并将其折叠到上一层的输出脉冲中。这种方法使我们能够在训练的 SNN 中保持归一化二值权重和实值脉冲的优势，而不改变神经元的更新过程。

有关我们 SNN 的训练和推理过程的详细说明，请参阅算法 1 。

4. 实验

我们进行了全面的实验来评估所提出的 ReverB-SNN 方法及其可学习二值权重变体的有效性。我们的评估包括与多种公认架构中的几种最先进方法进行比较。具体来说，我们在 CIFAR-10(100) (Krizhevsky et al., 2010) 上测试了尖峰 ResNet20 (Rathi & Roy, 2020; Sengupta et al., 2019) 和 ResNet19 (Zheng et al., 2021)，在 ImageNet 上测试了 ResNet18 (Fang et al., 2021a) 和 ResNet34 (Fang et al., 2021a)，以及在 CIFAR10-DVS (Li et al., 2017) 上测试了 ResNet20 和 ResNet19。

在我们的工作中，我们使用 SGD 优化器以动量 0.9 和学习率 0.1 训练模型，学习率根据余弦计划衰减至 0。对于 CIFAR10(100) 和 CIFAR-DVS 数据集，我们使用 128 的批量大小训练了 400 个 epoch。在 ImageNet 上，我们用相同的批量大小训练了 300 个 epoch。数据增强仅使用翻转操作。训练和测试数据集的划分遵循官方数据集提供的设置。膜电位衰减常数 τ 设置为 0.25。在这些静态数据集中，V_{th} 始终为 0，因为静态数据集不能提供时间信息。而对于神经形态数据集，我们将其设置为 0.25。

我们进行了一系列消融实验，以评估所提出的 ReverB-SNN 方法及其可学习二进制权重变体在 CIFAR-10 和 CIFAR-100 数据集上的有效性，采用 ResNet20 作为骨干模型，并用不同的时间步长进行实验。结果汇总在表 1 中。

在 CIFAR-10 和 CIFAR-100 上，基础 ResNet20 在 2 个时间步的基线准确率分别达到 92.80 % 和 70.18 %，这与之前的研究一致。实施 ReverB-SNN 方法显著提高了性能，分别提升至 94.14 % 和 72.54 %，大约相当于 1.30 % 和 2.50 % 的显著增强。此外，结合可学习的二进制权重变体带来了额外的性能增益，使最终的准确率达到了 94.45 %（CIFAR-10）和 72.95 %（CIFAR-100）。这些发现突显了我们方法在提升模型性能方面的有效性。当模型在 4 个时间步下进行评估时，我们的方法继续展示其有效性。通过这种配置观察到的性能提升进一步验证了 ReverB-SNN 技术的稳健性和有效性，强调了其在提高模型准确率方面的潜力，适用于各种设置。

4.1. 与最新技术方法的比较

在本节中，我们对比分析了我们的方法与当前最先进的方法。我们展示了 top-1 准确率结果及从 3 次试验中得到的平均准确率和标准差。我们的评估最初集中在 CIFAR-10 和 CIFAR-100 数据集上。总结后的结果如表 2 所示。对于 CIFAR-10 数据集，以前的方法使用 ResNet19 达到了 95.53 % 的峰值准确率，使用 ResNet20 达到了 93.66 %。而我们的 ReverB-SNN 方法分别实现了 96.39 % 和 94.55 %，同时使用较少的时间步。此外，利用可学习的二进制权重使我们的 SNN 模型能够获得更高的准确率。移至 CIFAR-100 数据集，我们可学习二进制权重的变体应用于 ResNet19 和 ResNet20 仅需 2 个时间步就能实现出色的性能。我们的方法超过了像 TET 和 RecDis-SNN 等领先的方法，并在使用 ResNet19 时大约提高了 4.0 %，尽管这些方法使用了 4 个时间步。这些实验结果强调了我们提出的方法的效率和效能。

在我们后续的实验中，我们在以复杂性著称的 ImageNet 数据集上评估了我们的方法，该数据集相比 CIFAR 更具挑战性。表格 3 展示了比较结果。该数据集最近的 SoTA 基准包括 RecDis-SNN (Guo et al., 2022c)、GLIF (Yao et al., 2022)、DSR (Meng et al., 2022)、Real Spike (Guo et al., 2022d) 和 SEW ResNet (Fang et al., 2021a)，其准确率分别为 67.33 %、67.52 %、67.74 %、67.69 % 和 67.04 %。我们的方法显著提高了准确率，达到了 70.91 %，比其他 SoTA SNN 模型提高了 3.22 %。这一显著的提升突显了我们的方法在大规模数据集上的有效性。

在我们的最终评估中，我们将 SNN 模型应用于 CIFAR10-DVS 神经形态数据集。利用 ResNet19 和

Dataset	Method	Type	Architecture	Timestep	Accuracy
CIFAR-10	SpikeNorm (Sengupta et al., 2019)	ANN2SNN	VGG16	2500	91.55 %
	Hybrid-Train (Rathi et al., 2020)	Hybrid training	VGG16	200	92.02 %
	TSSL-BP (Zhang & Li, 2020)	SNN training	CIFARNet	5	91.41 %
	TL (Wu et al., 2021a)	Tandem Learning	CIFARNet	8	89.04 %
	PTL (Wu et al., 2021b)	Tandem Learning	VGG11	16	91.24 %
	PLIF (Fang et al., 2021c)	SNN training	PLIFNet	8	93.50 %
	DSR (Meng et al., 2022)	SNN training	ResNet18	20	95.40 %
	KDSNN (Xu et al., 2023)	SNN training	ResNet18	4	93.41 %
	Diet-SNN (Rathi & Roy, 2020)	SNN training	ResNet20	5	91.78 %
				10	92.54 %
	Dspike (Li et al., 2021b)	SNN training	ResNet20	2	93.13 %
				4	93.66 %
	STBP-tdBN (Zheng et al., 2021)	SNN training	ResNet19	2	92.34 %
				4	92.92 %
	TET (Deng et al., 2022)	SNN training	ResNet19	2	94.16 %
				4	94.44 %
	RecDis-SNN (Guo et al., 2022c)	SNN training	ResNet19	2	93.64 %
				4	95.53 %
	Real Spike (Guo et al., 2022d)	SNN training	ResNet19	2	95.31 %
				4	95.51 %
			ResNet20	4	91.89 %
	ReverB	SNN training	ResNet19	1	95.97 % ±0.08
				2	96.39 % ±0.11
			ResNet20	2	94.14 % ±0.08
				4	94.55 % ±0.08
	Learnable variant	SNN training	ResNet19	1	96.22 % ±0.12
				2	96.62 % ±0.11
			ResNet20	2	94.45 % ±0.07
				4	94.96 % ±0.10
CIFAR-100	RMP (Han et al., 2020)	ANN2SNN	ResNet20	2048	67.82 %
	Hybrid-Train (Rathi et al., 2020)	Hybrid training	VGG11	125	67.90 %
	T2FSNN (Park et al., 2020)	ANN2SNN	VGG16	680	68.80 %
	Real Spike (Guo et al., 2022d)	SNN training	ResNet20	5	66.60 %
	LTL (Yang et al., 2022)	Tandem Learning	ResNet20	31	76.08 %
	Diet-SNN (Rathi & Roy, 2020)	SNN training	ResNet20	5	64.07 %
	RecDis-SNN (Guo et al., 2022c)	SNN training	ResNet19	4	74.10 %
	Dspike (Li et al., 2021b)	SNN training	ResNet20	2	71.68 %
				4	73.35 %
	TET (Deng et al., 2022)	SNN training	ResNet19	2	72.87 %
				4	74.47 %
	ReverB	SNN training	ResNet19	1	77.62 % ±0.10
				2	78.13 % ±0.13
			ResNet20	2	72.54 % ±0.09
				4	72.93 % ±0.12
	Learnable variant	SNN training	ResNet19	1	78.06 % ±0.08
				2	78.46 % ±0.12
			ResNet20	2	72.95 % ±0.11
				4	73.28 % ±0.08

Table 2. 与 CIFAR-10(100) 上的 SoTA 方法的比较。

ResNet20 作为我们的骨干架构，我们的方法分别取得了 80.50 % 和 78.10 % 的准确率，甚至超越了 ResNet19 的 80 % 里程碑。这标志着在这个广泛使用的神经形态数据集上的性能有了显著提升。

Method	Type	Architecture	Timestep	Accuracy
STBP-tdBN (Zheng et al., 2021)	SNN training	ResNet34	6	63.72 %
TET (Deng et al., 2022)	SNN training	ResNet34	6	64.79 %
RecDis-SNN (Guo et al., 2022c)	SNN training	ResNet34	6	67.33 %
OTTT (Xiao et al., 2022)	SNN training	ResNet34	6	65.15 %
GLIF (Yao et al., 2022)	SNN training	ResNet34	4	67.52 %
DSR (Meng et al., 2022)	SNN training	ResNet18	50	67.74 %
Ternary spike (Guo et al., 2024b)	SNN training	ResNet34	4	70.12 %
SSCL (Zhang et al., 2024)	SNN training	ResNet34	4	66.78 %
TAB (Jiang et al.)	SNN training	ResNet34	4	67.78 %
MPBN (Guo et al., 2023b)	SNN training	ResNet34	4	64.71 %
Shortcut back (Guo et al., 2024a)	SNN training	ResNet34	4	67.90 %
Multi-hierarchical model (Hao et al., 2023)	SNN training	ResNet34	4	69.73 %
SML (Deng et al., 2023)	SNN training	ResNet34	4	68.25 %
Real Spike (Guo et al., 2022d)	SNN training	ResNet18	4	63.68 %
		ResNet34	4	67.69 %
SEW ResNet (Fang et al., 2021a)	SNN training	ResNet18	4	63.18 %
		ResNet34	4	67.04 %
ReverB	SNN training	ResNet18	4	66.22 % ±0.16
		ResNet34	4	70.74 % ±0.13
Learnable variant	SNN training	ResNet18	4	66.58 % ±0.14
		ResNet34	4	70.91 % ±0.13

Table 3. 与当前最高水平方法在 ImageNet 上的比较。

在本节中，我们评估了使用 ResNet20 在 CIFAR10 上对单个图像进行 2 个时间步推理的传统 SNN 模型和 ReverB-SNN 模型相关的硬件能量成本。由于第一层的速率编码不具有无乘法特性，因此会产生 FLOPs（浮点运算）。而其他层是通过 SOPs（突触运算）计算的。SOPs 由 $s \times T \times A$ 计算，其中 s 是平均稀疏性，T 是时间步长，A 表示等效人工神经网络（ANN）模型中的加法次数。对于传统模型，SNN 的稀疏性为 16.42 %，而对于 ReverB-SNN 模型，其稀疏性为 17.18 %。我们基于 (Hu et al., 2021) 中概述的方法计算能耗，其中一个 FLOP 消耗 12.5 pJ，一个 SOP 消耗 77 fJ。能量成本的总结见表 5。与基准的传统模型相比，ReverB-SNN 方法仅导致能量成本小幅增加 0.52 %。这种微小的增加突出了 ReverB-SNN 方法的效率，表明它能够以相对较小的额外能量消耗实现性能的提升。

本研究引入了 ReverB-SNN，一种通过结合实值尖峰激活和二值权重来增强 SNN 的新方法。我们的方法解决了由于二值尖峰激活映射的信息捕获有限而导致 SNN 准确性降低的挑战。通过逆转权重和激活的位，我们保留了传统 SNN 的节能和无乘法特性，同时显著增强了激活的信息容量。此外，在二值权重中引入的可训练因子能够在训练期间自适应学习权重幅度，从而提升了整体网络容量。重要的是，为了确保与标准 SNNs 相当的操作效率，我们提出了一种重新参数化技术，在推理过程中将可训练的二值权重 SNNs 转换回标准形式。通过在多种网络架构和数据集上的广泛实验验证，涵盖静态和动态场景，我们的方法一直显示出相对现有最先进方法的优越性。

本研究得到了中国国家重点研发计划（编号 2024YDLN0013）和国家自然科学基金（编号 12202412）的支持。

5. 影响声明

本文提出的工作旨在推动机器学习领域的发展。我们的工作可能会对社会产生许多潜在影响，但我们认为这里不需要特别强调这些影响。

References

Akopyan, F., Sawada, J., Cassidy, A., Alvarez-Icaza, R., Arthur, J., Merolla, P., Imam, N., Nakamura, Y., Datta, P., Nam, G.-J., et al. 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, 2015.

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B. Simple online and realtime tracking. In 2016 IEEE international conference on image processing (ICIP), pp. 3464–3468. IEEE, 2016.

Bohnstingl, T., Woźniak, S., Pantazi, A., and Eleft-

Method	Type	Architecture	Timestep	Accuracy
DSR (Meng et al., 2022)	SNN training	VGG11	20	77.27 %
GLIF (Yao et al., 2022)	SNN training	7B-wideNet	16	78.10 %
STBP-tdBN (Zheng et al., 2021)	SNN training	ResNet19	10	67.80 %
RecDis-SNN (Guo et al., 2022c)	SNN training	ResNet19	10	72.42 %
Real Spike (Guo et al., 2022d)	SNN training	ResNet19	10	72.85 %
Dspike (Li et al., 2021b)	SNN training	ResNet20	10	75.40 %
Spikformer (Zhou et al., 2023)	SNN training	Spikformer	10	78.90 %
SSCL (Zhang et al., 2024)	SNN training	ResNet19	10	80.00 %
ReverB	SNN training	ResNet19	10	80.30 % ±0.20
		ResNet20	10	77.80 % ±0.10
Learnable variant	SNN training	ResNet19	10	80.50 % ±0.10
		ResNet20	10	78.10 % ±0.10

Table 4. 与 CIFAR10-DVS 上的 SoTA 方法比较。

Method	Accuracy	# Flops	# Sops	Energy
Vanilla SNN	92.80 %	3.54M	71.20M	49.73uJ
ReverB	94.14 %	3.54M	74.50M	49.99uJ

Table 5. 能量估计。

heriou, E. Online spatio-temporal learning in deep neural networks. IEEE Transactions on Neural Networks and Learning Systems, pp. 1–15, 2022. doi: 10.1109/TNNLS.2022.3153985.

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

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

Deng, S., Li, Y., Zhang, S., and Gu, S. Temporal efficient training of spiking neural network via gradient re-weighting. arXiv preprint arXiv:2202.11946, 2022.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

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

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

Fang, W., Yu, Z., Chen, Y., Masquelier, T., Huang, T., and Tian, Y. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2661–2671, 2021c.

Feng, L., Liu, Q., Tang, H., Ma, D., and Pan, G. Multilevel firing with spiking ds-resnet: Enabling better and deeper directly-trained spiking neural networks. In Raedt, L. D. (ed.), Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022, pp. 2471–2477. ijcai.org, 2022. doi: 10.24963/ijcai.2022/343. URL https://doi.org/10.24963/ijcai.2022/343.

Gong, R., Liu, X., Jiang, S., Li, T., Hu, P., Lin, J., Yu, F., and Yan, J. Differentiable soft quantization: Bridging full-precision and low-bit neural networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4852–4861, 2019.

Guo, Y., Chen, Y., Zhang, L., Liu, X., Wang, Y., Huang, X., and Ma, Z. IM-loss: Information maximization loss for spiking neural networks. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), Advances in Neural Information Process-

ing Systems, 2022a. URL https://openreview.net/forum?id=Jw34v_84m2b.

Guo, Y., Chen, Y., Zhang, L., Wang, Y., Liu, X., Tong, X., Ou, Y., Huang, X., and Ma, Z. Reducing information loss for spiking neural networks. In Avidan, S., Brostow, G., Cissé, M., Farinella, G. M., and Hassner, T. (eds.), Computer Vision – ECCV 2022, pp. 36–52, Cham, 2022b. Springer Nature Switzerland. ISBN 978-3-031-20083-0.

Guo, Y., Tong, X., Chen, Y., Zhang, L., Liu, X., Ma, Z., and Huang, X. Recdis-snn: Rectifying membrane potential distribution for directly training spiking neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 326–335, June 2022c.

Guo, Y., Zhang, L., Chen, Y., Tong, X., Liu, X., Wang, Y., Huang, X., and Ma, Z. Real spike: Learning real-valued spikes for spiking neural networks. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XII, pp. 52–68. Springer, 2022d.

Guo, Y., Huang, X., and Ma, Z. Direct learning-based deep spiking neural networks: a review. Frontiers in Neuroscience, 17:1209795, 2023a.

Guo, Y., Zhang, Y., Chen, Y., Peng, W., Liu, X., Zhang, L., Huang, X., and Ma, Z. Membrane potential batch normalization for spiking neural networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 19420–19430, 2023b.

Guo, Y., Chen, Y., Hao, Z., Peng, W., Jie, Z., Zhang, Y., Liu, X., and Ma, Z. Take a shortcut back: Mitigating the gradient vanishing for training spiking neural networks. arXiv preprint arXiv:2401.04486, 2024a.

Guo, Y., Chen, Y., Liu, X., Peng, W., Zhang, Y., Huang, X., and Ma, Z. Ternary spike: Learning ternary spikes for spiking neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pp. 12244–12252, 2024b.

Guo, Y., Peng, W., Chen, Y., Zhou, J., and Ma, Z. Improved event-based image de-occlusion. IEEE Signal Processing Letters, 2024c.

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

Guo, Y., Liu, X., Chen, Y., Peng, W., Zhang, Y., and Ma, Z. Spiking transformer: Introducing accurate addition-only spiking self-attention for transformer. arXiv preprint arXiv:2503.00226, 2025.

Han, B. and Roy, K. Deep spiking neural network: Energy efficiency through time based coding. In European Conference on Computer Vision, pp. 388–404. Springer, 2020.

Han, B., Srinivasan, G., and Roy, K. 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.

Hao, Z., Shi, X., Huang, Z., Bu, T., Yu, Z., and Huang, T. A progressive training framework for spiking neural networks with learnable multi-hierarchical model. In The Twelfth International Conference on Learning Representations, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016.

He, Y., Zhang, X., and Sun, J. Channel pruning for accelerating very deep neural networks. In Proceedings of the IEEE international conference on computer vision, pp. 1389–1397, 2017.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Hu, Y., Tang, H., and Pan, G. Spiking deep residual networks. IEEE Transactions on Neural Networks and Learning Systems, 2021.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

Jiang, H., Zoonekynd, V., De Masi, G., Gu, B., and Xiong, H. Tab: Temporal accumulated batch normalization in spiking neural networks. In The Twelfth International Conference on Learning Representations.

Kim, S., Park, S., Na, B., and Yoon, S. 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.

Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 (canadian institute for advanced research). URL http://www. cs. toronto. edu/kriz/cifar. html, 5(4): 1, 2010.

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

Li, Y., Deng, S., Dong, X., Gong, R., and Gu, S. A free lunch from ann: Towards efficient, accurate spiking neural networks calibration. In International Conference on Machine Learning, pp. 6316–6325. PMLR, 2021a.

Li, Y., Guo, Y., Zhang, S., Deng, S., Hai, Y., and Gu, S. Differentiable spike: Rethinking gradient-descent for training spiking neural networks. Advances in Neural Information Processing Systems, 34:23426–23439, 2021b.

Liu, F., Zhao, W., Chen, Y., Wang, Z., and Jiang, L. Spikeconverter: An efficient conversion framework zipping the gap between artificial neural networks and spiking neural networks. In Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022, pp. 1692–1701. AAAI Press, 2022. URL https://ojs.aaai.org/index.php/AAAI/article/view/20061.

Ma, D., Shen, J., Gu, Z., Zhang, M., Zhu, X., Xu, X., Xu, Q., Shen, Y., and Pan, G. Darwin: A neuromorphic hardware co-processor based on spiking neural networks. Journal of Systems Architecture, 77:43–51, 2017.

Maass, W. Networks of spiking neurons: The third generation of neural network models. Neural Networks, 10(9):1659–1671, 1997. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(97)00011-7. URL https://www.sciencedirect.com/science/article/pii/S0893608097000117.

Meng, Q., Xiao, M., Yan, S., Wang, Y., Lin, Z., and Luo, Z.-Q. 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, pp. 12444–12453, 2022.

Ming, Q., Miao, L., Ma, Z., Zhao, L., Zhou, Z., Huang, X., Chen, Y., and Guo, Y. Deep dive into gradients: Better optimization for 3d object detection with gradient-corrected iou supervision. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5136–5145, 2023. doi: 10.1109/CVPR52729.2023.00497.

Park, S., Kim, S., Na, B., and Yoon, S. T2fsnn: Deep spiking neural networks with time-to-first-spike coding. In 2020 57th ACM/IEEE Design Automation Conference (DAC), pp. 1–6. IEEE, 2020.

Pei, J., Deng, L., Song, S., Zhao, M., Zhang, Y., Wu, S., Wang, G., Zou, Z., Wu, Z., He, W., et al. Towards artificial general intelligence with hybrid tianjic chip architecture. Nature, 572(7767):106–111, 2019.

Polino, A., Pascanu, R., and Alistarh, D. Model compression via distillation and quantization. arXiv preprint arXiv:1802.05668, 2018.

Qin, H., Zhang, Y., Ding, Y., Liu, X., Danelljan, M., Yu, F., et al. Quantsr: accurate low-bit quantization for efficient image super-resolution. Advances in Neural Information Processing Systems, 36, 2024.

Rathi, N. and Roy, K. Diet-snn: Direct input encoding with leakage and threshold optimization in deep spiking neural networks. arXiv preprint arXiv:2008.03658, 2020.

Rathi, N., Srinivasan, G., Panda, P., and Roy, K. Enabling deep spiking neural networks with hybrid conversion and spike timing dependent backpropagation. arXiv preprint arXiv:2005.01807, 2020.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pp. 234–241. Springer, 2015.

Sengupta, A., Ye, Y., Wang, R., Liu, C., and Roy, K. Going deeper in spiking neural networks: Vgg and residual architectures. Frontiers in neuroscience, 13: 95, 2019.

Subramoney, A., Nazeer, K. K., Schöne, M., Mayr, C., and Kappel, D. Efficient recurrent architectures through activity sparsity and sparse backpropagation through time. In The Eleventh International Conference on Learning Representations, 2023. URL https://openreview.net/forum?id=lJdOlWg8td.

Sun, C., Chen, Q., Fu, Y., and Li, L. Deep spiking neural network with ternary spikes. In 2022 IEEE Biomedical Circuits and Systems Conference (BioCAS), pp. 251–254. IEEE, 2022.

Wang, S., Schmutz, V., Bellec, G., and Gerstner, W. Mesoscopic modeling of hidden spiking neurons. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), Advances in Neural Information Processing Systems, 2022. URL https://openreview.net/forum?id=cYPja_wj9d.

Wang, X., Zhang, Y., and Zhang, Y. Mt-snn: Enhance spiking neural network with multiple thresholds, 2023.

Wu, J., Chua, Y., Zhang, M., Li, G., Li, H., and Tan, K. C. A tandem learning rule for effective training and rapid inference of deep spiking neural networks. IEEE Transactions on Neural Networks and Learning Systems, 2021a.

Wu, J., Xu, C., Han, X., Zhou, D., Zhang, M., Li, H., and Tan, K. C. Progressive tandem learning for pattern recognition with deep spiking neural networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(11):7824–7840, 2021b.

Wu, Y., Deng, L., Li, G., Zhu, J., Xie, Y., and Shi, L. Direct training for spiking neural networks: Faster, larger, better. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pp. 1311–1318, 2019.

Xiao, M., Meng, Q., Zhang, Z., Wang, Y., and Lin, Z. Training feedback spiking neural networks by implicit differentiation on the equilibrium state. Advances in Neural Information Processing Systems, 34:14516–14528, 2021.

Xiao, M., Meng, Q., Zhang, Z., He, D., and Lin, Z. Online training through time for spiking neural networks. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), Advances in Neural Information Processing Systems, 2022. URL https://openreview.net/forum?id=Siv3nHYHheI.

Xu, Q., Li, Y., Shen, J., Liu, J. K., Tang, H., and Pan, G. Constructing deep spiking neural networks from artificial neural networks with knowledge distillation. arXiv preprint arXiv:2304.05627, 2023.

Yang, Q., Wu, J., Zhang, M., Chua, Y., Wang, X., and Li, H. Training spiking neural networks with local tandem learning. arXiv preprint arXiv:2210.04532, 2022.

Yao, M., Hu, J., Zhou, Z., Yuan, L., Tian, Y., Xu, B., and Li, G. Spike-driven transformer. Advances in neural information processing systems, 36:64043–64058, 2023.

Yao, X., Li, F., Mo, Z., and Cheng, J. Glif: A unified gated leaky integrate-and-fire neuron for spiking neural networks. arXiv preprint arXiv:2210.13768, 2022.

Yu, K., Yu, C., Zhang, T., Zhao, X., Yang, S., Wang, H., Zhang, Q., and Xu, Q. Temporal separation with entropy regularization for knowledge distillation in spiking neural networks. arXiv preprint arXiv:2503.03144, 2025.

Yu, Q., Ma, C., Song, S., Zhang, G., and Tan, K. C. Constructing accurate and efficient deep spiking neural networks with double-threshold and augmented schemes. IEEE Transactions on Neural Networks and Learning Systems, PP(99):1–13, 2021.

Zhang, L., Bao, C., and Ma, K. Self-distillation: Towards efficient and compact neural networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(8):4388–4403, 2022. doi: 10.1109/TPAMI.2021.3067100.

Zhang, W. and Li, P. Temporal spike sequence learning via backpropagation for deep spiking neural networks. Advances in Neural Information Processing Systems, 33:12022–12033, 2020.

Zhang, Y., Liu, X., Chen, Y., Peng, W., Guo, Y., Huang, X., and Ma, Z. Enhancing representation of spiking neural networks via similarity-sensitive contrastive learning. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pp. 16926–16934, 2024.

Zheng, H., Wu, Y., Deng, L., Hu, Y., and Li, G. Going deeper with directly-trained larger spiking neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pp. 11062–11070, 2021.

Zhou, Z., Zhu, Y., He, C., Wang, Y., YAN, S., Tian, Y., and Yuan, L. Spikformer: When spiking neural network meets transformer. In The Eleventh International Conference on Learning Representations, 2023. URL https://openreview.net/forum?id=frE4fUwz_h.