

论文“脉冲深度学习梯度替代算法研究综述(2024/8/29-013640)”的修改说明

感谢诸位审稿人提供的精彩意见，这些意见中肯地指出了本文目前的问题。我们根据这些宝贵意见进行了修改，进一步提升了文章质量。在本修改说明中，对于从正文摘录来的文字，原有的内容用蓝色标注，修改的内容用红色标注。下面是对每一个审稿意见的单独回复：

意见 1：第 7 页 3.3 节中“PLIF 神经元”和“CLIF 神经元”的空格需调整。

对意见 1 的修改说明：感谢您对排版的细致检查。该问题是英文单词不能中断导致的，我们已经修改了相关内容，避开了这一排版问题，修改后的内容截图如下：

最早的神经动态可学习的神经元模型之一是 PLIF 神经元 (Parametric Leaky Integrate-and-Fire Neuron) 模型^[75]，其将 LIF 神经元的膜时间常数 τ_m 参数化并设置为可学习，阈下神经动态为：

LIF 神经元的漏电行为可能会导致长期梯度衰减，为解决这一问题，CLIF 神经元 (Complementary Leaky Integrate-and-Fire Neuron)^[96] 通过增加补充电位 (Complementary Potential) 实现跨多个时间步的稳

意见 2：第 5 页“常用的替代函数包括 Rectangular^[46]、SuperSpike^[47]、ArcTan^[75]、Sigmoid 等，这些函数大多是单调递增的偶函数”，应该是单调递增的奇函数。

对意见 2 的修改说明：感谢您仔细的审查，我们的表述存在错误。现已修改为如下内容：

常用的替代函数包括 Rectangular^[46]、SuperSpike^[47]、ArcTan^[75]、Sigmoid 等，这些函数大多单调递增且关于 $(x, y) = (0, 0.5)$ 中心对称，值域为 $(0, 1)$ ，可以视作 $\Theta(x)$ 的光滑近似。

意见 3：综合对比实验中，目标检测任务只有 Gen1 数据集，建议在文章中再进行补充。

对意见 3 的修改说明：感谢您关于充实文章内容的意见。考虑到之前的研究大多也在 COCO 数据集上进行了实验，我们增加了这一数据集的实验结果，相关内容如下：

4.3 Gen1 和 COCO 目标检测性能

本文使用神经形态的 Gen1^[135] 数据集和静态的 COCO 数据集进行目标检测任务训练。Gen1 数据集由事件相机采集的驾驶场景组成，COCO 数据集则是最常用的静态目标检测数据集，两者已经被大量 SNN 目标检测相关研究使用。对于 Gen1 数据集，本文使用 Fan 等^[141] 的开源代码和网络结构，使用 Spiking DenseNet121-16^[137] 作为检测的骨干网络，首先在神经形态的 NCAR^[138] 数据集上预训练分类任务，然后在 Gen1 数据集上进行目标检测任务训练并记录最终的 mAP。对于 COCO 数据集，本文使用 Su 等^[140] 提出的全脉冲结构的 Energy-efficient Membrane-Shortcut (EMS) ResNet-10 并直接训练。

.....

对于 COCO 数据集，以 mAP@0.5:0.95 作为主要指标，性能排序是 CLIF > IF > TEBN > LIF > Sliding PSN > OSR；BlockALIF 神经元训练速度太慢，耗时约为其他方法的 8 倍，不

具备完成训练的可行性，因而在表格中的性能是空缺值.整体来看，除 BlockALIF 神经元训练太慢、OSR 方法性能较差，其余方法的性能较为接近.突触操作数排序是 $TEBN < IF < LIF < CLIF < OSR < Sliding PSN$. 值得注意之处在于，CLIF 神经元达到了最优性能，远强于在 Gen1 数据集的表现，但突触操作数明显高于 LIF 神经元；TEBN 方法则相较于使用普通 BN、作为基准的 IF 神经元性能略微下降，而其在 Gen1 数据集则表现最好，但该方法在两种数据集上的突触操作数都最低；Sliding PSN 的突触操作数约为其他方法的 2 倍，表明去除重置带来脉冲发放率增加的负面效果较为严重.需要指出的是，在 Gen1 数据集的目标检测任务中，本文使用的 Fan 等^[141]的网络结构和训练流程，将 SNN 作为检测骨干(Backbone)网络，而检测头则是由 ANN 实现，这也是绝大多数已有研究使用的方式.而对于 COCO 数据集的目标检测任务，本文沿用的 Su 等^[140]的方法，则是使用全脉冲的流程，检测骨干网络和检测头均由 SNN 实现；且输入图片的分辨率较高，为 640×640 ；由于上述原因，在 COCO 数据集上的突触操作数要远高于 Gen1 数据集，因而在表 4 中两者的单位分别是 G 和 M.

表 4 对比各类代表性方法 SHD 分类和 Gen1、COCO 目标检测任务性能

评估指标\方法	IF	LIF	CLIF	Sliding PSN	TEBN	OSR	BlockALIF
SHD 分类正确率(%)	78.33	66.98	66.1	69.23	74.54	40.93	82.36
SHD 分类突触操作数(M)	0.0251	0.0262	0.0263	0.0264	0.0262	0.0228	0.0239
NCAR 预训练分类正确率							
(%)	81.00	91.69	85.34	91.41	90.96	56.23	68.14
Gen1 目标检测 mAP@0.5	0.0013	0.4399	0.2388	0.4781	0.5213	0.0007	0.1347
Gen1 目标检测							
mAP@0.5:0.95	0.0002	0.2106	0.090204	0.2391	0.2656	0.0003	0.0457
Gen1 目标检测突触操作							
数(M)	385.86	296.33	353.03	293.53	214.72	305.93	232.69
COCO 目标检测 mAP@0.5	0.409	0.397	0.410	0.389	0.408	0.247	—
COCO 目标检测							
mAP@0.5:0.95	0.225	0.216	0.228	0.213	0.223	0.120	—
COCO 目标检测突触操作							
数(G)	363.79	444.67	537.37	1004.68	360.62	561.75	—

此外，相关实验的代码和训练日志，我们也更新在了网盘文件中，下载链接地址不变。相关的超参数在补充材料中进行了说明：

2.5 COCO 目标检测实验细节

本文使用 Su 等^[140]提出的 EMS ResNet 进行直接训练.用于比较的各类方法，除方法本身的超参数外和训练轮数外，其余超参数均与 Su 等^[140]保持一致，具体为：使用 ResNet-10 结构，批量大小为 16，图片尺寸为 640×640 ，使用动量为 0.937 的 SGD 优化器、OneCycle 学习率调节器^[215]、初始学习率为 0.01、最终学习率为 0.1、权重衰减 0.0005，预热训练轮数为 3，预热训练时动量为 0.8、学习率为 0.1，启用混合精度训练.由于 COCO 数据集图片分辨率较大，训练速度较慢，且本文需要对比多种方法，为减少实验成本，本文将训练轮数从 Su 等^[140]设置的 300 轮减少为 100 轮.

对于不同方法，需要额外调节的超参数如下：

Sliding PSN：使用 $k = 3$.

TEBN: 实验发现 IF 神经元好于 LIF 神经元, 故使用 LIF 神经元配合 TEBN.

意见 4: 对于“脉冲 Transformer”的介绍, 建议可以将 Transformer 的关键机制(如多头注意力机制)与脉冲神经网络结合。

对意见 4 的修改说明:感谢您关于修改完善内容的意见, 我们已经对在 SNN 中对 Transformer 关键机制进行修改的动机进行了说明:

Transformer^[106]是继 ResNet 之后的影响力最大的网络结构, 自提出以来便在多个领域刷新了性能指标, 成为目前人工智能领域最常用的网络架构之一, 其核心机制包括多头自注意力(Multi-Head Self Attention)和位置编码(Positional Encoding)等.传统的卷积神经网络结构中, 除去神经元自身的运算, 其余的矩阵运算发生在突触和激活值之间.在网络结构脉冲化后, 参与矩阵运算的一方为脉冲, 可以实现事件驱动、无乘法器的计算.而在 Transformer 中, 除突触和激活值的运算外, 自注意力机制中的矩阵乘法使用的是未经过激活函数作用的全连接层的原始输出, 且其中还使用 Softmax 函数; 前者涉及稠密浮点值的矩阵乘法, 后者则需要指数运算, 这些特性难以与神经形态计算芯片兼容.此外, 位置编码通常需要浮点值, 也与 SNN 的二值特性违背.因而如何解决上述问题, 有效结合 Transformer 架构的高性能和 SNN 的低功耗, 引起了脉冲深度学习领域内学者的广泛兴趣.

此外, 我们还增加了少量关于脉冲 Transformer 中其他机制改进的研究论述:

除最为核心的自注意力机制外, 也有少量研究者探讨了 Transformer 中其他机制的改进.Lv 等^[124]受到人脑中中枢模式发生器工作原理的启发, 设计了基于正弦函数的位置编码方式, 其产生二值的编码值, 避免了传统位置编码带来的浮点计算, 并提升了网络的时间序列处理性能. Zhou 等^[125]对脉冲 Transformer 的混合专家模型进行了研究, 提出了适用于 SNN 的脉冲专家混合机制(Spiking Experts Mixture Mechanism, SEMM).该研究指出, ANN 中由于混合专家模型采用 Softmax 计算路由权重和 Top-*K*, 硬稀疏地选取专家, 并不适用于 SNN 事件驱动的计算和动态稀疏激活的特性.为了解决该问题, SEMM 将每个注意力头视为独立专家, 并使用脉冲路由模块对其进行稀疏激活, 在集成多头注意力的同时达到了动态稀疏激活的效果.

意见 5: 请仔细检查全文, 修改其中的语法及文字错误。

对意见 5 的修改说明:感谢您关于修改错误的意见, 我们已经对全文进行了仔细检查, 修正了错别字、排版、语序、连词、指代不明等问题, 修改内容罗列在下表中。由于该表明确标注了原文和修改后内容, 故在该表中, 我们不再以蓝色标注修改前的内容:

编号	原文内容	修改后内容
1	接收多个输入并输出布尔值	接收多个输入并输出布尔(Bool)值
2	引发了第一次神经网络热潮.感知机不能处理非线性的异或(Exclusive OR, XOR)问题,	引发了第一次神经网络热潮.但感知机不能处理非线性的异或(Exclusive OR, XOR)问题,
3	训练 SNN 比 ANN 更为困难	训练 SNN 比训练 ANN 更为困难
4	1990 年 Mead 提出的神经形态计算概念	1990 年 Mead 提出神经形态计算的概念
5	时识科技(SynSense)的异步神经形态	时识科技(SynSense)的异步神经形态感算

	感算一体 Speck 芯片	一体芯片 Speck
6	具有较为复杂的神经动态.来自其他神经元的输入电信号通过树突(Dendrite)传递到神经元的胞体	具有较为复杂的神经动态.生物神经网络中,来自其他神经元的输入电信号通过树突(Dendrite)传递到神经元的胞体
7	则释放脉冲,使用 Heaviside 阶跃函数 $\Theta(x)$ 描述这一过程	则释放脉冲.使用 Heaviside 阶跃函数 $\Theta(x)$ 描述这一放电过程:
8	拥有数值正常的导	拥有数值正常的导数
9	对于替代函数梯度的另一种解释基于概率性发放脉冲的期望的梯度 ^[48, 76] , 这一思想也被量化神经网络用于解释直通估计器	对于替代函数梯度的另一种解释基于概率性发放脉冲的期望的梯度 ^[48, 76] , 先前这一思想也被量化神经网络用于解释直通估计器
10	则需要非常精细地调整学习率才能达到较好性能	则需要非常精细地调整学习率才能达到目标性能
11	目前研究者们通常将替代函数的梯度缩放到最大值为 1 来避免梯度爆炸问题	目前研究者们通常将替代函数的梯度缩放到最大值为 1 来缓解梯度爆炸问题
12	而 a 是真正的可学习参数	a 是可学习参数
13	较于传统神经元	相较于传统神经元
14	但神经元层输出的不再是纯二值脉冲	但神经元层的输出不再是纯二值脉冲
15	因而网络的任务性能也进一步提升,但这通常也会导致计算代价的提升	因而网络的任务性能也进一步提升,但这通常也会导致计算代价的增加
16	因此无法有效加深 SNN	因此无法有效加深 SNN 以获取性能增益
17	但同时也引入了新的问题.SEW ResNet 主要使用实验性能最好的加法	但同时也引发了新的问题.具体而言, SEW ResNet 主要使用性能最好的加法
18	$score[t] = K[t] \cdot SN(\sum_n Q[t]), (37)$	$score[t] = K[t] \cdot SN(\sum_d Q[t]), (37)$
19	$score'[t] = K'[t] \cdot SN(\sum_d Q'[t]), (38)$	$score'[t] = K'[t] \cdot SN(\sum_n Q'[t]), (38)$
20	Shen 等 ^[127] 更关注于搜索的生物原理性	Shen 等 ^[129] 更关注于搜索的生物可解释性
21	整体来看,网络结构搜索类方法本身训练开销较大,与需要多个时间步运行的 SNN 结合后问题更为明显,已有的研究也多聚焦于此问题.	整体来看,网络结构搜索类方法本身训练开销较大,与需要多个时间步运行的 SNN 结合后问题更为明显,已有的研究也多聚焦于解决计算成本问题.
22	其中 T_i 是经过边缘 ReLU 处理以抑制负信息影响后的教师 ANN 模型的中间层特征	其中 T_i 是经过边缘 ReLU 处理后以抑制负信息影响的教师 ANN 模型的中间层特征
23	基于频率的损失函数同样适用于时序的事件驱动学习方案	基于频率的损失函数同样适用于时序的事件驱动学习方法
24	于是在前向传播中断开了除最后一层外的时序传播过程	于是在前向传播中断开了除最后一层外的时序传播计算图
25	使用可逆模块解决了使用 $\mathcal{O}(1)$ 存储空	使用可逆模块实现了使用 $\mathcal{O}(1)$ 存储空间

	间记录 T 步信息的问题，其关键是	记录 T 步信息的 效果 ，其 核心方法 是
26	这一近似相当于认为当前时刻的输出脉冲，对未来时刻的损失不会产生影响，其实并不符合实际情况.	这一近似其实并不符合实际情况， 例如分类任务中每个时间步发放的脉冲都会参与发放频率的计算，并影响分类结果.
27	将满足 $ H[t]-V_{th} \geq B_{th}$ 的神经元视作不活跃的神经元	将满足 $ H[t]-V_{th} \geq B_{th}$ 的神经元视作不活跃的神经元， 其中 B_{th} 是梯度阈值超参数
28	蒸馏类方法一方面需要 ANN 计算，另一方面引入了额外损失，故内存消耗量比直接训练的方法更大，且特征蒸馏的损失项更多，故内存消耗量也显著高于响应蒸馏方法.	蒸馏类方法一方面需要 ANN 计算，另一方面引入了额外损失，故内存消耗量比直接训练的方法更大； 特征蒸馏的损失项比响应蒸馏更多，故内存消耗量进一步提升.
29	因而这些变量在 CUDA 执行完之后就自动释放了	因而这些变量在 CUDA 内核 执行完之后就自动释放了
30	但目前研究还处在初级阶段，实际性能较为一般，且对超参数敏感、稳定性差、任务正确率较低	但目前研究还处在初级阶段，实际性能较为 落后 ，且对超参数敏感、稳定性差
31	但将其结合梯度替代法其用于改善大规模深度 SNN 的学习	但将其结合梯度替代法 并 用于改善大规模深度 SNN 的学习

此外，原文的表 5 中有一列数据有误。表 5 记录了加速比，而耗时的原始数据在补充材料的表 6 中。表 6 的数据是正确的，表 5 是根据表 6 的原始数据进行计算而得到的。但我们在排版时将表 5 中第 5 列的数据也粘贴到了第 6 列，导致两列数据相同。该错误不影响正文的讨论内容和结论。现在已经修复了错误：

表 5 对比加速方法性能

T	相较于 LIF 神经元的加速比						LIF 耗时 (ms)
	SpikingJelly	PSN	BlockALIF 分组大小				
			2	4	8	16	
2	1.03	2.20	0.20				1.44
4	1.48	4.07	0.17	0.38			3.02
8	2.72	6.81	0.15	0.29	0.60		4.79
16	6.19	12.60	0.22	0.29	0.56	1.29	9.48
32	16.61	17.76	0.25	0.40	0.59	1.01	17.14
64	14.83	43.75	0.24	0.45	0.72	0.98	30.60