

类脑智能大作业报告

刘翰文

522030910109

摘要

脉冲神经网络 (Spiking Neural Networks, SNNs) 因其独特的脉冲驱动范式和显著的能效优势, 在类脑计算中引起了广泛的关注。在本次大作业中, 我们实现了作业要求的基础、中级和高级部分。在基础部分中, 我们基于传统的 VGG 和 ResNet 神经网络, 实现了 VGG 和 ResNet 神经网络到脉冲神经网络的迁移; 在中级部分, 我们提出了一种基于脉冲序列相似性的脉冲神经网络分布外泛化算法; 在高级部分, 我们基于近几年提出的 GhostNet, 实现了它的脉冲神经网络版本, 同时对网络提出了注意力方面的优化, 为神经网络在移动设备上的应用带来了新的前景。

1 引言

脉冲神经网络 (Spiking Neural Networks, SNNs) 是目前在脑启发智能领域中最具前景的工作之一, 凭借其出色的生物学可解释性和显著的能耗优势, 有望成为人工神经网络 (Artificial Neural Networks, ANNs) 的替代方案。近年来, 已有许多工作聚焦于脉冲神经网络。例如, [6] 首次提出了一种用于训练深度脉冲神经网络的无监督方法。[11] 将经典的 YOLO[14] 架构转化成脉冲神经网络, 实现了实时目标检测。[7] 基于传统的残差架构 [9], 提出了一种适用于脉冲神经网络的残差学习方法。[18] 基于 [19] 的工作, 提出了高能效的脉冲 Transformer。

鉴于近年来脉冲神经网络展现的出色表现, 我们结合了此次大作业的三个部分, 对其进行了充分的研究。我们将此次大作业三个部分的完成内容总结如下:

- 在基础部分中, 我们参照传统的 VGG 神经网络 [15] 和 ResNet 神经网络 [9], 实现了 VGG 神经网络和 ResNet 神经网络到脉冲神经网络的转换, 并在 CIFAR-10 数据集 [12] 上进行训练和测试。在训练过程中, 我们发现基于传统 ResNet 架构的 Vanilla Shortcut(VS)[20] 的脉冲 ResNet 无法进行学习, 因此我们搭建了基于 Spike-Element-Wise Shortcut(SEW)[7] 和 Membrane Shortcut(MS)[10] 的脉冲 ResNet 进行实验。
- 在中级部分中, 我们尝试将 IRM 算法 [4] 直接应用于搭建的卷积脉冲神经网络中, 并在 Colored MNIST 数据集 [4] 上进行训练和测试。同时, 我们根据脉冲神经网络脉冲驱动的特点, 提出了一种基于脉冲序列相似性的分布外泛化算法, 测试其在 Colored MNIST 数据集上的表现。
- 在高级部分中, 我们尝试将脉冲神经网络的高能效优势引入可移植的高效神经网络中。为此, 我们实现了 GhostNet[8] 的脉冲神经网络版本, 并引入了 DFC 注意力机制 [16]。同时, 我们参考 [18] 提出的网络, 将通道注意力机制引入 GhostNet。最后, 我们将上述所有网络在 CIFAR-10 数据集上进行训练和测试, 并对结果进行了比较分析。

2 原理

在本节中，我们简单地介绍了本次大作业涉及到的原理和方法。在2.1节中，我们简单介绍了脉冲神经网络的基本计算单元——脉冲神经元。到了2.2节，我们描述了在脉冲残差结构中的三种Shortcut。在2.3节中，我们描述了我们在中级部分中使用的算法。最后，在2.4节中，我们对高级部分涉及的 GhostNet 做了一个简单的概述。

2.1 脉冲神经元

脉冲神经元通过膜电位的变化表示时空信息，将它转换为 0 和 1 的脉冲，以供下一层进行脉冲驱动计算。我们所搭建的所有网络均采用了 Leaky Intergrate-and-Fire(LIF)[13] 脉冲神经元，它的积分、泄露和发射过程可以用如下公式表示：

$$U[t] = H[t - 1] + X[t], \quad (1)$$

$$S[t] = \text{Hea}(U[t] - u_{th}), \quad (2)$$

$$H[t] = V_{reset}S[t] + (\beta U[t])(1 - S[t]), \quad (3)$$

其中， $X[t]$ 是在 t 时刻的空间输入电流， $U[t]$ 表示膜电位，是 $X[t]$ 和时间输入 $H[t - 1]$ 之和。 $\text{Hea}()$ 表示阶跃函数，当 $x \geq 0$ 时输出 1，否则为 0。当 $U[t]$ 超过发放阈值 u_{th} 时，脉冲神经元将发放一个脉冲 $S[t]$ ，并将时间输出 $H[t]$ 重置为 V_{reset} 。否则， $U[t]$ 将会衰减到 $H[t]$ ，其中 $\beta < 1$ 是衰减因子。

2.2 残差连接

在脉冲神经网络中，主要存在着三种残差连接结构，分别为：Vanilla Shortcut、Spike-Element-Wise Shortcut 和 Membrane Shortcut，如图1所示。用 U' 来表示输入， $SN(\cdot)$ 表示脉冲神经元，

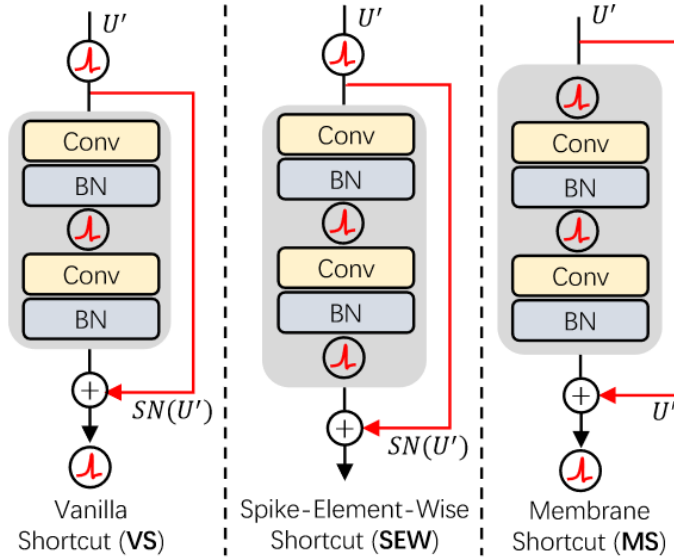


图 1: 脉冲神经网络的三种残差连接

O' 表示输出, 将 $BN(Conv(\cdot))$ 简写为 $F(\cdot)$ 。三种残差连接的输出可以依次用公式4、5、6表示。

$$O' = SN(F(SN(F(U')) + U') \quad (4)$$

$$O' = SN(F(SN(F(U')))) + U' \quad (5)$$

$$O' = F(SN(F(SN(U')))) + U' \quad (6)$$

2.3 脉冲序列相似性

脉冲神经网络的一个最主要的特征是其脉冲驱动的范式, 信息被表示在一个个脉冲序列中。例如, 在使用脉冲神经网络对图像进行分类中, 一个类别对应的序列中拥有的脉冲数量越多, 往往代表这该图像属于此类的概率越大。而序列之间的相似度越高, 其中蕴含的信息也往往越相似。在这次大作业中, 我们使用了余弦相似度来衡量两个序列之间的相似度, 可以用如下公式表示:

$$Similarity(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n (A_i \times B_i)}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}} \quad (7)$$

其中 A 和 B 表示两个 n 维序列, 脉冲序列的相似度 $Similarity(A, B) \in [0, 1]$, 越高的相似度代表两个序列越相似。

2.4 GhostNet

本次大作业的高级部分实现了 GhostNet 到脉冲神经网络的转换, GhostNet 的设计目的是为了减轻可移植神经网络中的特征图冗余问题, 它主要由 Ghost bottleneck 堆叠而成, Ghost bottleneck 的结构如图2所示。其中 SE Module(Squeeze-and-Excitation) 为注意力机制模块, Ghost

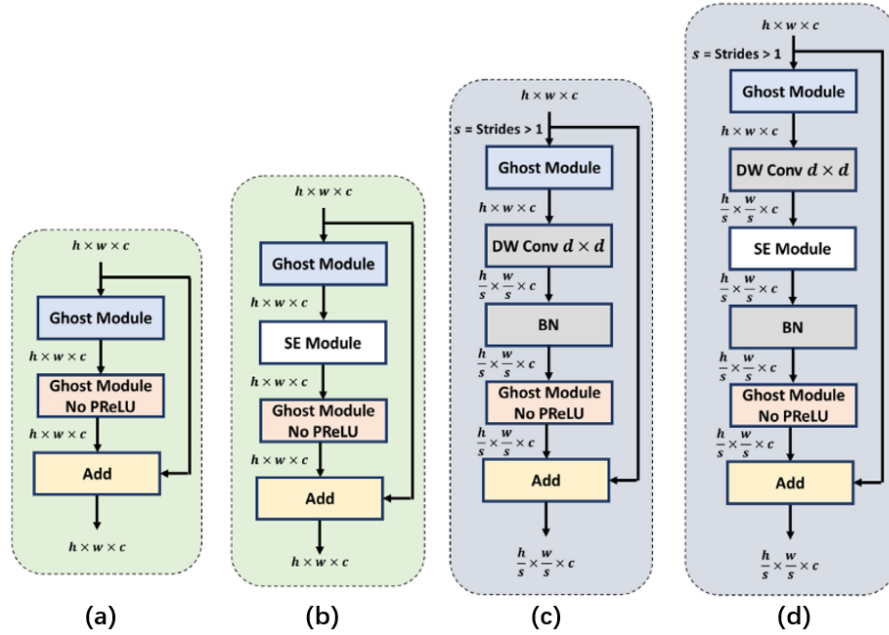


图 2: (a) 无下采样且不带 SE 模块的 Ghost bottleneck (b) 无下采样但带 SE 模块的 Ghost bottleneck (c) 带下采样和 SE 模块的 Ghost bottleneck (d) 带下采样但不带 SE 模块的 Ghost bottleneck

Module 结构如图3所示。不同于传统的卷积操作，Ghost Module 首先对输入进行轻量卷积，得到一半的特征图，然后将得到的特征图经过逐层卷积后与第一步结果进行拼接，最终输出与卷积操作数量相同但冗余程度减小的特征图，同时大大减小了计算量。

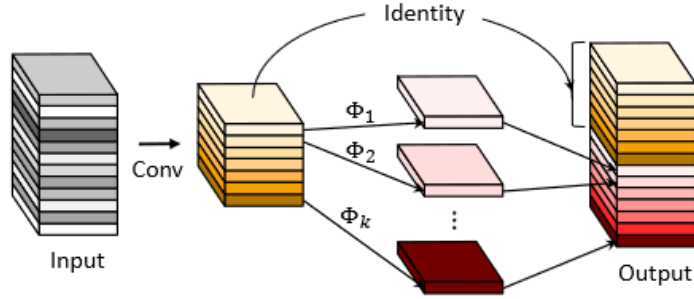


图 3: Ghost Module 结构图

3 实现

在本节中，我们将会具体阐述我们是如何进行大作业以及得到的结果。在3.1节中，我们首先阐明了部分我们进行实验所用的参数、设置等。在3.2、3.3、3.4节中，我们分别展示了我们是如何完成大作业的基础、中级和高级部分的以及得到的对应结果。

3.1 实验设置

本次大作业我们全程使用 Nvidia RTX 4060 GPU 进行，对涉及到的所有脉冲神经网络取时间步长 $T = 6$ ，在基础和高级部分使用 SGD 优化器，初始学习率为 0.1 并设置学习率下降，权重衰减系数为 $1e-4$ ，在 CIFAR-10 数据集上训练 190 个 epochs。在中级部分，我们使用 Adam 优化器并设置学习率为 $1e-3$ ，在 Colored MNIST 数据集上训练 30-50 个 epochs。

3.2 基础部分

对于 VGG 神经网络，由于原始网络被设计用来对 ImageNet 数据集 [5] 做图像分类，无法直接适用于 CIFAR-10。所以我们首先修改了 VGG 神经网络的架构使之适配 CIFAR-10 数据集，然后选择 VGG-16 网络作为基础网络，采取直接替换的方法构建对应的脉冲神经网络，即将 VGG 网络中的激活函数替换成 LIF 脉冲神经元。将输入图片在时间维度上复制 $T = 6$ 份，在最后的线性层输出沿着时间维度计算每个类别的平均脉冲数作为最终输出。

对于 ResNet 神经网络，我们选择了 ResNet-34 作为基础网络，与 VGG 神经网络一样首先修改网络参数，随后采用直接替换的方法构建对应的脉冲神经网络。但是在实验过程中，我们发现搭建的 ResNet 脉冲神经网络在整个训练过程中的准确率维持在 10%，无法通过训练进行提升。在通过阅读文献 [7, 18] 和实验验证后，我们发现问题的来源在于脉冲神经元和传统激活函数的差异导致传统 ResNet 的残差结构 Vanilla Shortcut 无法实现恒等映射，即 $O' = U'$ ，导致网络出现

了退化问题。具体来说，在人工神经网络和脉冲神经网络中，输出 O'_A, O'_S 分别可以表示为

$$O'_A = \text{ReLU}(F(\text{ReLU}(F(U'_A))) + U'_A) \quad (8)$$

$$O'_S = \text{SN}(F(\text{SN}(F(U'_S))) + U'_S) \quad (9)$$

当 $F(\text{ReLU}(F(U'_A))) \equiv 0$, $F(\text{SN}(F(U'_S))) \equiv 0$ 时, $O'_A = \text{ReLU}(U'_A)$, $O'_S = \text{SN}(U'_S)$ 。而在大多数情况下, U'_A 是上一层激活函数的输出结果, 故在经过一层 ReLU 函数后保持不变, 即 $O'_A = \text{ReLU}(U'_A) = U'_A$, 可以实现恒等映射。但到了脉冲神经网络, 由于脉冲神经元的特性, $O'_S = \text{SN}(U'_S) \neq U'_S$, 无法实现恒等映射。

基于以上原因以及我们的测试结果, 我们分别将原网络的 Vanilla shortcut 修改为 SEW shortcut 和 Membrane shortcut 的脉冲 ResNet, 成功的摆脱了原网络无法训练的困境, 进而证实了我们的想法。我们的实验结果如表1所示。结果显示, 脉冲神经网络版本的 VGG-16 和 SEW-ResNet-34 在准确率上表现相当, 且都优于 MS-ResNet-34。但在模型参数、FLOPs 和能耗上, SEW-ResNet-34 对于 VGG-16 有较大的优势。同时, 为了实现大作业要求, 我们还在 MNIST 数据集上进行了测试, 结果如表2所示。可见在 MNIST 数据集上的表现远优于在 CIFAR-10 数据集上的表现。

| 网络架构 | 模型参数 (M) | FLOPs(M) | 能耗 (μJ) | 时间步长 | 准确率 (%) |
|---------------|----------|----------|----------------|------|--------------|
| VGG-16 | 14.79 | 317.0 | 64.70 | 6 | 90.18 |
| SEW-ResNet-34 | 0.47 | 72.5 | 14.80 | 6 | 89.03 |
| MS-ResNet-34 | 0.47 | 72.5 | 8.30 | 6 | 82.39 |

表 1: 网络在 CIFAR-10 上的实验结果

| 网络架构 | 模型参数 (M) | FLOPs(M) | 能耗 (μJ) | 时间步长 | 准确率 (%) |
|---------------|----------|----------|----------------|------|--------------|
| VGG-16 | 14.79 | 315.8 | 64.63 | 6 | 99.03 |
| SEW-ResNet-34 | 0.46 | 72.2 | 14.97 | 6 | 99.18 |
| MS-ResNet-34 | 0.46 | 72.2 | 8.61 | 6 | 99.13 |

表 2: 网络在 MNIST 上的实验结果

3.3 中级部分

神经网络的泛化能力一直以来都是对网络进行评估的重要标准之一, 然而网络在有监督的学习下泛化性通常较差, 解决这个问题关键是要学习到具有因果关系的特征。[4] 面对这个问题, 提出了 IRM 算法和一个专门用来评测模型分布外泛化能力的数据集 Colored MNIST, 将目标描述为一个优化问题:

$$\min_{\Phi: \mathcal{X} \rightarrow \mathcal{Y}} \sum_{e \in \mathcal{E}_{tr}} \mathcal{R}^e(\Phi) + \lambda \cdot \|\nabla_{\omega|_{\omega=1.0}} \mathcal{R}^e(\omega \cdot \Phi)\|^2 \quad (10)$$

其中, 左式表示经验风险最小项, 右式表示不变风险最小项, λ 为一个权重参数。其主要原理是使模型在两个训练环境中的训练损失都尽可能达到最优。

IRM 算法主要建立在人工神经网络的基础上, 无法判断其在脉冲神经网络上的效果。为此, 我们搭建了时间步长 $T = 8$ 的卷积脉冲神经网络, 使用 IRM 算法在 Colored MNIST 数据集上

进行训练。我们将 Colored MNIST 的训练数据集分成两个部分，将第一部分 10% 的背景颜色进行翻转，将第二部分 20% 的背景颜色进行翻转，作为两个训练环境，最后将测试集 90% 的背景颜色进行翻转。设置权重 λ 随训练过程从 0 线性增长，训练 50 个 epochs，结果如图5a所示。从训练过程的准确率曲线来看，结果一直在随机猜测和高训练准确率、低测试准确率之间徘徊，无法找到优化目标较优的解。由此判断单纯地将 IRM 算法应用到脉冲神经网络上无法得到较好的结果。

考虑到脉冲神经网络独特的脉冲驱动的模式，我们提出了一种基于脉冲序列相似性的分布外泛化算法，其主要原理是通过减小序列相似性损失，使得网络在两个训练环境上的序列差异尽可能的小，从而学习到在两个环境上保持一致的具有因果关系的特征。

具体来说，我们将训练损失 \mathcal{L}_{total} 分为两个部分：优化分类准确率的交叉熵损失项 $\mathcal{L}_{crossentropy}$ 和优化两个训练环境上脉冲序列差异的相似性损失项 $\mathcal{L}_{similarity}$ 。其中对于相似性损失，我们分别取网络在两个训练环境中经过最后一个线性层前的结果 O_1, O_2 ，按批次维度计算它们的余弦相似度 $Similarity(O_1, O_2)$ ，使得：

$$\mathcal{L}_{similarity} = 1 - Similarity(O_1, O_2) \quad (11)$$

$$\mathcal{L}_{total} = \mathcal{L}_{crossentropy} + \lambda \cdot \mathcal{L}_{similarity} \quad (12)$$

其中， λ 是权重系数，随 epoch 线性增长。我们在 Colored MNIST 数据集上对提出的算法进行测试，结果如图5b和表3所示。表3展示的结果为在训练集上的平均准确率高于 70% 时在测试集上最高的准确率的结果。与 IRM 算法的结果相比，我们的算法在测试集上的表现稍优于 IRM 算法，且没有在训练过程中出现随机猜测的结果。在同样的数据集上，我们还尝试了将 O_1, O_2 从最后一个线性层前换成从倒数第二个线性层前，即进入所有线性层前的结果，如图4所示，我们将 Similarity1 替换为 Similarity2 计算损失。结果如图5c所示，通过比较5b和5c的结果，取第一个线性层的输出作为序列能够使训练过程更稳定，训练和测试的准确率更高。

除此之外，我们还测试了不同的训练环境对测试结果的影响。具体而言，我们采用倒数第一层线性层前的序列计算损失，设置第一个训练环境被翻转颜色的比例分别为 5%、2.5%、0%，设置第二个训练环境被翻转颜色的比例分别为 25%、27.5%、30%。在保持训练环境平均翻转比例不变的同时，改变翻转比例之差，最后在原测试集上进行测试，我们的结果分别如图5d、5e、5f所示，发现两条训练曲线与翻转比例之间存在着一定关系，比例差距越高，曲线整体距离越远，这也表面我们的算法实质上仍未完全学习到因果特征，否则在两个训练集上效果应该相当。在表3中也展示了我们所有的结果。

| 算法/数据集 | 训练集一 (%) | 训练集二 (%) | 测试集 (%) |
|--------------|----------|----------|-------------|
| IRM 算法 | 73.2 | 68.1 | 41.2 |
| 相似性算法 | 75.8 | 65.0 | 42.1 |
| 线性层前 | 73.4 | 73.2 | 32.5 |
| 5% 和 25% | 75.4 | 67.9 | 35.1 |
| 2.5% 和 27.5% | 87.2 | 67.9 | 23.7 |
| 0% 和 30% | 77.4 | 64.5 | 33.6 |

表 3: 中级部分结果

结合表3，我们可以得出结论：IRM 算法直接应用于脉冲神经网络无法实现在人工神经网络

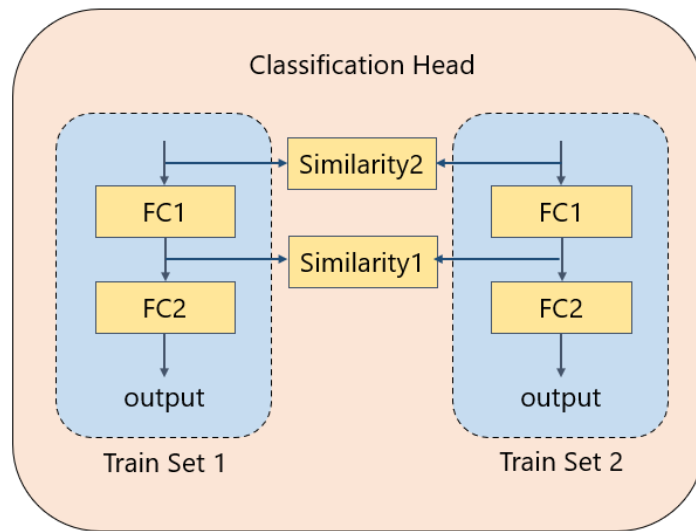
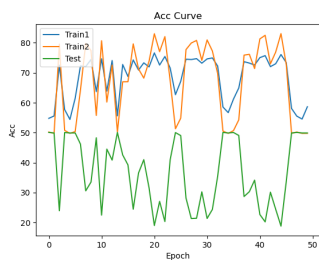
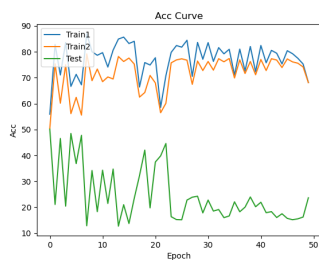


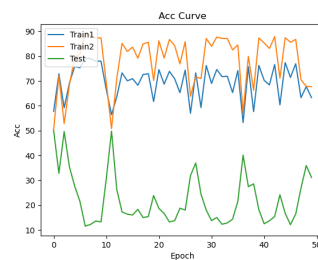
图 4: 不同序列的选择



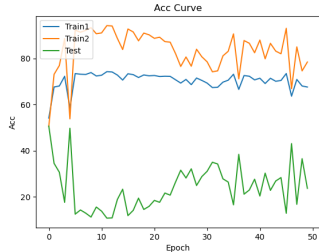
(a) IRM 算法结果



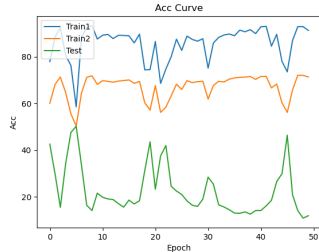
(b) 相似性算法结果



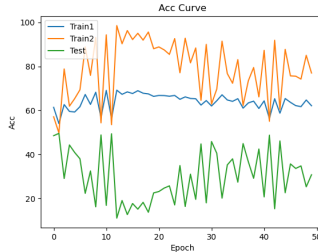
(c) 线性层前序列结果



(d) 5% 和 25% 结果



(e) 2.5% 和 27.5% 结果



(f) 0% 和 30% 结果

图 5: 中级部分结果图

上的结果，且在训练过程中容易多次出现随机猜测的情况；我们提出的序列相似性算法在效果上和 IRM 算法相当，但在稳定性上明显优于 IRM 算法；通过不同序列的对比，我们发现使用最后一层线性层前的序列计算相似性的效果明显优于使用线性层前的序列计算相似性；不同的翻转比例对两个训练集的结果的影响不同，说明我们的算法仍未较好学习到因果特征。在训练过程中，在 λ 的作用下， $\mathcal{L}_{similarity}$ 逐渐减小为 0，如图6所示。但我们观察到测试集的准确率仍处于较低水平，同样说明了我们的算法不具备较好的泛化学习能力。

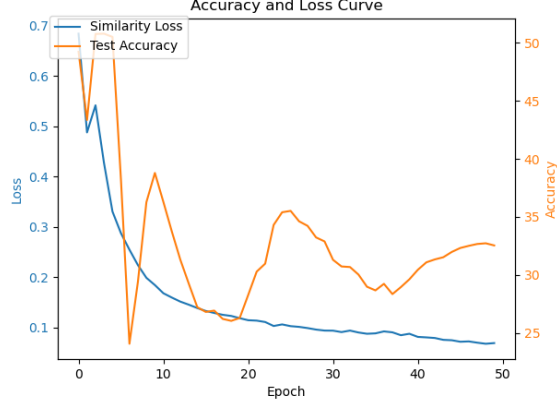


图 6: $\mathcal{L}_{similarity}$ 和 Test Acc 变化曲线

3.4 高级部分

在本节中，我们将介绍大作业中高级部分的实现过程以及我们对它的改进。在3.4.1节中，我们实现了原始 GhostNet 到脉冲神经网络的转换，在3.4.2节中，我们将 DFC 注意力机制添加到原始的 GhostNet 中，在3.4.3节中，我们再次对网络进行了改进，引入了通道注意力机制。

3.4.1 脉冲 GhostNet

我们首先按照人工神经网络中 GhostNet 的架构转换为脉冲神经网络的形式，按图7所示的架构进行搭建，并对网络参数进行修改使之可以运行在 CIFAR-10 数据集上。随后，我们在 CIFAR-10 数据集上对网络进行训练测试，所有的结果均呈现在表3.4中。从结果不难看出，直接转换的 GhostNet 在 CIFAR-10 数据集上的表现不如初级部分搭建的 VGG16 和 ResNet34，但不管是模型参数、FLOPs 还是能耗都远远优于上述网络，很好地符合其可移植属性。

3.4.2 DFC 注意力机制

我们对 GhostNet 在 CIFAR-10 上的表现进行了分析，认为其中的 GhostModule 在脉冲神经网络中无法实现在人工神经网络中的效果。在原论文 [8] 中，作者认为 GhostModule 是一个轻量化即插即用的模块，可以代替其他神经网络中的卷积层，做到在不损失较多精度的同时大大减少计算量。为了验证我们的猜想，我们将脉冲形式的 GhostModule 作为一个即插即用的模块，代替了我们在基础部分搭建的 SEW-ResNet 的卷积层。结果表明，替换 GhostModule 后的 SEW-ResNet 在 CIFAR-10 数据集上仅有 60% 左右的准确率，因此证实了我们的猜想。

| Input | Operator | #exp | #out | SE | Stride |
|-------------------|----------------------|------|------|----|--------|
| $224^2 \times 3$ | Conv2d 3×3 | - | 16 | - | 2 |
| $112^2 \times 16$ | G-bneck | 16 | 16 | - | 1 |
| $112^2 \times 16$ | G-bneck | 48 | 24 | - | 2 |
| $56^2 \times 24$ | G-bneck | 72 | 24 | - | 1 |
| $56^2 \times 24$ | G-bneck | 72 | 40 | 1 | 2 |
| $28^2 \times 40$ | G-bneck | 120 | 40 | 1 | 1 |
| $28^2 \times 40$ | G-bneck | 240 | 80 | - | 2 |
| $14^2 \times 80$ | G-bneck | 200 | 80 | - | 1 |
| $14^2 \times 80$ | G-bneck | 184 | 80 | - | 1 |
| $14^2 \times 80$ | G-bneck | 184 | 80 | - | 1 |
| $14^2 \times 80$ | G-bneck | 480 | 112 | 1 | 1 |
| $14^2 \times 112$ | G-bneck | 672 | 112 | 1 | 1 |
| $14^2 \times 112$ | G-bneck | 672 | 160 | 1 | 2 |
| $7^2 \times 160$ | G-bneck | 960 | 160 | - | 1 |
| $7^2 \times 160$ | G-bneck | 960 | 160 | 1 | 1 |
| $7^2 \times 160$ | G-bneck | 960 | 160 | - | 1 |
| $7^2 \times 160$ | G-bneck | 960 | 160 | 1 | 1 |
| $7^2 \times 160$ | Conv2d 1×1 | - | 960 | - | 1 |
| $7^2 \times 960$ | AvgPool 7×7 | - | - | - | - |
| $1^2 \times 960$ | Conv2d 1×1 | - | 1280 | - | 1 |
| $1^2 \times 1280$ | FC | - | 1000 | - | - |

图 7: GhostNet 架构

在保留 GhostModule 的计算量优势的前提下，我们尝试对 GhostModule 的输出进行改进。相较于卷积层，GhostModule 将卷积操作的结果分成了两部分结果的拼接，为了减小这两部分在脉冲网络和人工网络中的差异同时增强模型对图片空间位置的感知能力，我们参考 [16] 在人工神经网络中的应用，引入了脉冲版本的具有低计算复杂度的 DFC 注意力机制并对其做了部分修改，其原理如图8所示，具体的计算方法不在此处详细展开。为了适配 CIFAR-10 数据集较小的特点，我们删除了原有的降采样和升采样操作，并将 DFC 注意力机制与 GhostModule 结果逐元素相乘，新的 Ghost bottleneck 结构如图9中的 (a) 所示。对比于 DFC 自注意力机制，在 Transformer[17] 和 Spiking Transformer[18] 中使用的自注意力机制因其大量的参数和计算量在便于部署和轻量化上的表现逊于 DFC 注意力机制。我们在 CIFAR-10 上的测试结果如表4所示，在测试集上的准确率稍高于原始的网络结构，同时增加了少量的参数和计算量。

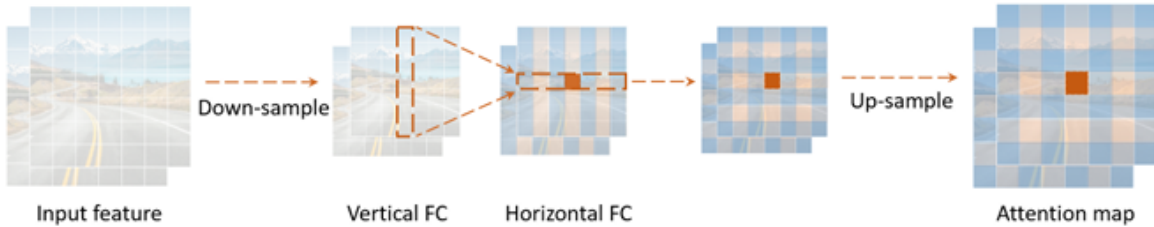


图 8: DFC 注意力机制

3.4.3 通道注意力机制

基于 DFC 注意力在 GhostNet 上的表现，我们参考了 [18] 所使用的 Meta Transformer Block 架构 (见图10)，将 GhostModule 视作 TokenMixer，将每个 GhostModule 与一个 Channel MLP

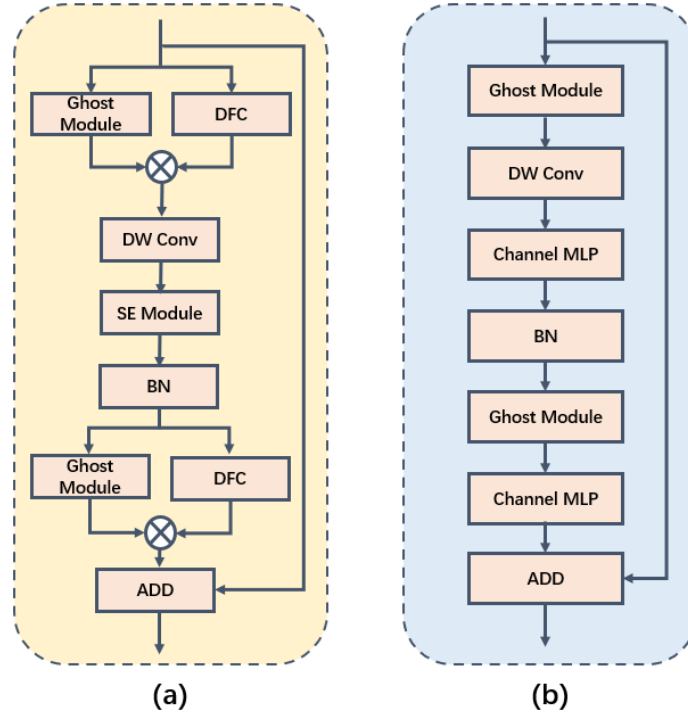


图 9: (a) 引入 DFC 注意力的 Ghost bottleneck (b) 使用通道注意力的 Ghost bottleneck

相连，引入了通道注意力机制，改进后的 Ghost bottleneck 架构如图9中的 (b) 所示。

我们使用的 Channel MLP 由两层全连接层和一层脉冲神经元组成，负责将 Ghost Module 的输出进行通道层次重新加权分配。值得一提的是，我们并没有采用图10中的残差连接，因为我们在 Ghost bottleneck 中已经实现了从输入到输出的残差连接。且在实际过程中，我们对尝试将 Ghost bottleneck 的残差结构更改为图10所示的多个小型残差结构，但得不到较好的结果。我们同样对新的网络架构在 CIFAR-10 数据集上进行了实验，结果如表4所示。结果显示，我们在 CIFAR-10 数据集上取得了表中三个网络的最高准确率，并且能耗同时达到最优。

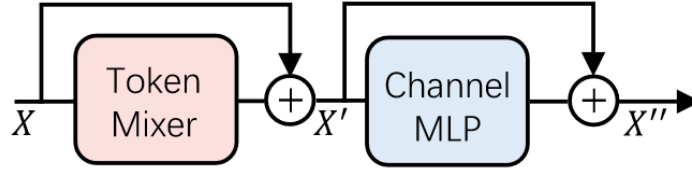


图 10: Meta Transformer Block 架构

| 网络架构 | 模型参数 (M) | FLOPs(M) | 能耗 (μJ) | 时间步长 | 准确率 (%) |
|---------------------|--------------|--------------|----------------|------|--------------|
| GhostNet-SNN | 3.9 | 6.4 | 1.04 | 6 | 80.12 |
| GhostNet-DFC | 4.9 | 9.7 | 1.53 | 6 | 80.42 |
| GhostNet-ChannelMLP | 4.7 | 7.1 | 0.96 | 6 | 81.30 |

表 4: GhostNet 在 CIFAR-10 上的实验结果

4 总结

在本次大作业中,我们完成了初级、中级和高级的任务。在初级部分中,我们测试了自己搭建了 VGG16 的脉冲版本,并在分析脉冲 ResNet34 的缺陷后搭建了 SEW-ResNet34 和 MS-ResNet34。在中级部分中,我们提出了一种基于序列相似性的分布外泛化算法并研究了序列出处、训练环境差异对结果的影响。在高级部分中,我们实现了 GhostNet 到脉冲神经网络的迁移并对其提出了两种有效的改进,我们在附录中对人员等进行了说明并展示了对初级和高级部分搭建的脉冲神经网络的可视化结果和训练过程图。我们相信脉冲神经网络在今后的发展中一定会达到崭新的高度。

参考文献

- [1] ghostnet_cifar10. https://github.com/TingsongYu/ghostnet_cifar10.
- [2] invariant-risk-minimization. <https://github.com/reiinakano/invariant-risk-minimization>.
- [3] Invariantriskminimization. <https://github.com/facebookresearch/InvariantRiskMinimization>.
- [4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [6] Peter U Diehl and Matthew Cook. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Frontiers in computational neuroscience*, 9:99, 2015.
- [7] Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34:21056–21069, 2021.
- [8] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1580–1589, 2020.
- [9] 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*, pages 770–778, 2016.
- [10] Yifan Hu, Lei Deng, Yujie Wu, Man Yao, and Guoqi Li. Advancing spiking neural networks toward deep residual learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

- [11] 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, pages 11270–11277, 2020.
- [12] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [13] Wolfgang Maass. Networks of spiking neurons: the third generation of neural network models. *Neural networks*, 10(9):1659–1671, 1997.
- [14] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
- [15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [16] Yehui Tang, Kai Han, Jianyuan Guo, Chang Xu, Chao Xu, and Yunhe Wang. Ghostnetv2: Enhance cheap operation with long-range attention. *Advances in Neural Information Processing Systems*, 35:9969–9982, 2022.
- [17] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- [18] Man Yao, Jiakui Hu, Tianxiang Hu, Yifan Xu, Zhaokun Zhou, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips. *arXiv preprint arXiv:2404.03663*, 2024.
- [19] Man Yao, Jiakui Hu, Zhaokun Zhou, Li Yuan, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer. *Advances in neural information processing systems*, 36, 2024.
- [20] Hanle Zheng, Yujie Wu, Lei Deng, Yifan Hu, and Guoqi Li. Going deeper with directly-trained larger spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11062–11070, 2021.

附录

A 说明

我们使用了.py 文件运行本次大作业的代码,结果文件已与代码一同打包,详细信息见 README 文件。本小组成员: 刘翰文, 参考代码 [1, 2, 3]。

B 输出脉冲可视化

我们在图11中展示了在所有网络上首个 Conv2d-BatchNorm2d-LIFNode 后的脉冲可视化的结果。

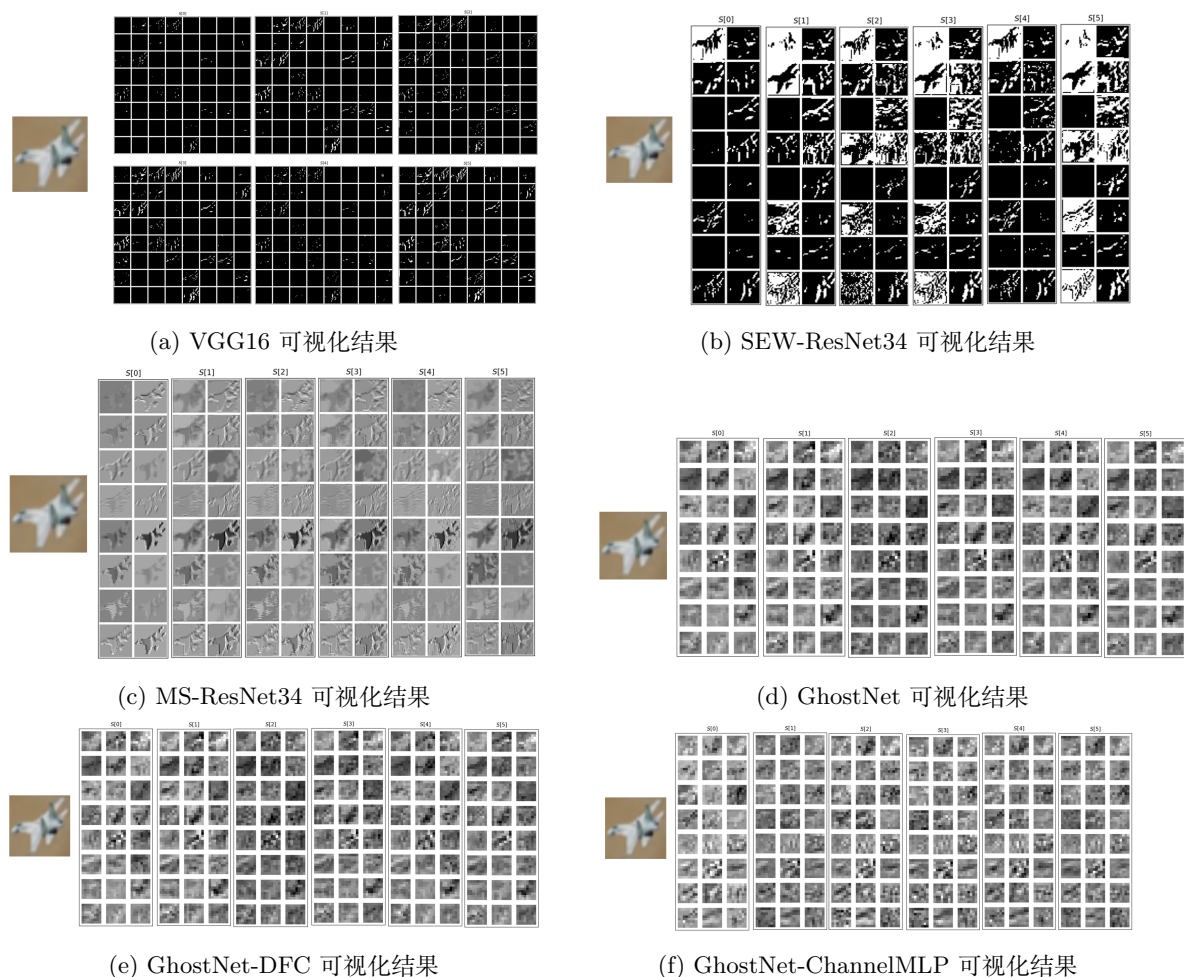
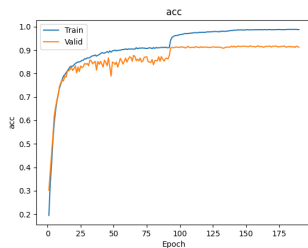


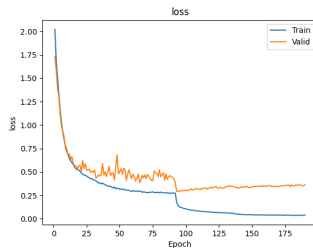
图 11: 输出脉冲可视化

C 训练准确率及损失

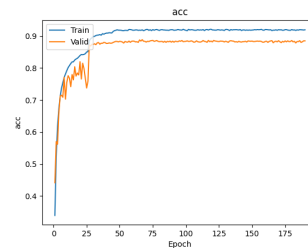
图12展示了我们训练的六个网络在 CIFAR-10 训练集和测试集上准确率和损失的变化结果。



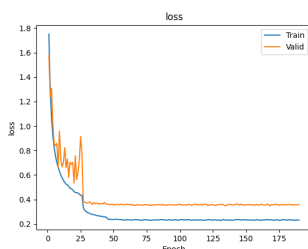
(a) VGG16 准确率曲线



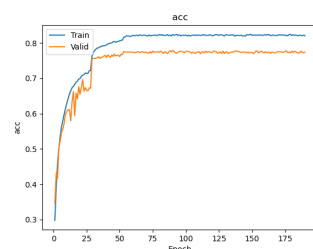
(b) VGG16 损失曲线



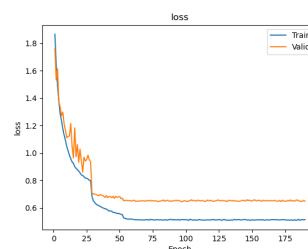
(c) SEW-ResNet34 准确率曲线



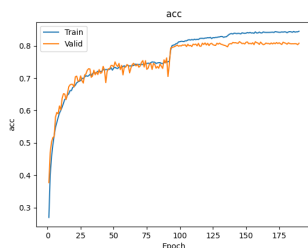
(d) SEW-ResNet34 损失曲线



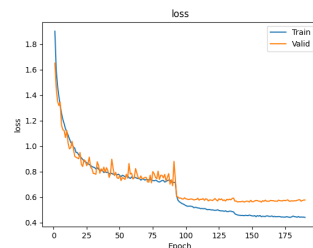
(e) MS-ResNet34 准确率曲线



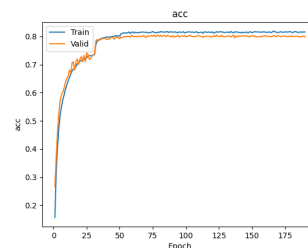
(f) MS-ResNet34 损失曲线



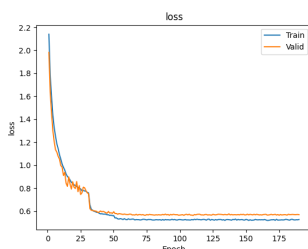
(g) GhostNet 准确率曲线



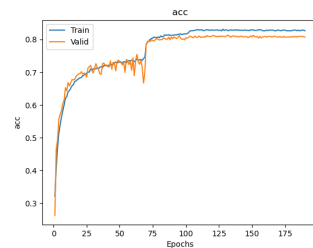
(h) GhostNet 损失曲线



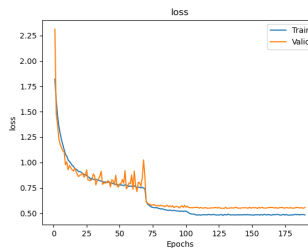
(i) GhostNet-DFC 准确率曲线



(j) GhostNet-DFC 损失曲线



(k) GhostNet-ChannelMLP 准确率曲线



(l) GhostNet-ChannelMLP 损失曲线

图 12: 脉冲神经网络训练结果 (CIFAR-10)