# 脉冲神经网络权重量化方法与对抗鲁棒性分析

李 莹[①] 李艳杰[①②] 崔小欣[③] 倪庆龙[①②] 周鉴灏[*①]

[①](中国科学院微电子研究所 北京 100029)

[②](中国科学院大学集成电路学院 北京 100049)

[③](北京大学集成电路学院 北京 100087)

**摘 要**：类脑芯片中的脉冲神经网络(SNNs)具有高稀疏性和低功耗的特点，在视觉分类任务中存在应用优势，但仍面临对抗攻击的威胁。现有研究缺乏对网络部署到硬件的量化过程中鲁棒性损失的度量方法。该文研究硬件映射阶段的SNN权重量化方法及其对抗鲁棒性。建立基于反向传播和替代梯度的监督训练算法，并在CIFAR-10数据集上生成快速梯度符号法(FGSM)对抗攻击样本。创新性地提出一种感知量化的权重量化方法，并建立与对抗攻击的训练与推理相融合的评估框架。实验结果表明，在VGG9网络下，直接编码对抗鲁棒性最差。在权重量化前后，4种编码和4种结构参数组合方式下，推理精度损失差与层间脉冲活动的平均变化幅度分别增大73.23%和51.5%。该文指出稀疏性因素对鲁棒性的影响相关度为：阈值增加大于权重量化bit降低大于稀疏编码，所提对抗鲁棒性分析框架与权重量化方法在PIcore类脑芯片中得到了硬件验证。

**关键词**：脉冲神经网络；权重量化；对抗鲁棒性；稀疏性；对抗攻击

**中图分类号**：TN918; TP183 **文献标识码**：A **文章编号**：1009-5896(2023)09-3218-10

**DOI**: 10.11999/JEIT230300

# Weight Quantization Method for Spiking Neural Networks and Analysis of Adversarial Robustness

LI Ying[①] LI Yanjie[①②] CUI Xiaoxin[③] NI Qinglong[①②] ZHOU Yinhao[①]

[①](*Institute of Microelectronics*, *Chinese Academy of Sciences*, *Beijing* 100029, *China*)

[②](*School of Integrated Circuits*, *University of Chinese Academy of Sciences*, *Beijing* 100049, *China*)

[③](*School of Integrated Circuits*, *Peking University*, *Beijing* 100087, *China*)

**Abstract**: Spiking Neural Networks (SNNs) in neuromorphic chips have the advantages of high sparsity and low power consumption, which make them suitable for visual classification tasks. However, they are still vulnerable to adversarial attacks. Existing studies lack robustness metrics for the quantization process when deploying the network into hardware. The weight quantization method of SNNs during hardware mapping is studied and the adversarial robustness is analyzed in this paper. A supervised training algorithm based on backpropagation and alternative gradients is proposed, and one types of adversarial attack samples, Fast Gradient Sign Method (FGSM), on the CIFAR-10 dataset are generated. A perception quantization method and an evaluation framework that integrates adversarial training and inference are provided innovatively. Experimental results show that direct encoding leads to the worst adversarial robustness in the VGG9 network. The difference between the accuracy loss and inter-layer pulse activity change before and after weight quantization increases by 73.23% and 51.5%, respectively, for four encoding and four structural parameter combinations. The impact of sparsity factors on robustness is: threshold increase more than bit reduction in weight quantization more than sparse coding. The proposed analysis framework and weight quantization method have been proved on the PIcore neuromorphic chip.

**Key words**: Spiking Neural Network (SNN); Weight quantization; Adversarial robustness; Sparsity; Adversarial attack

收稿日期：2023-04-19；改回日期：2023-08-17；网络出版：2023-08-23

*通信作者： 周鉴灏 zhouyinhao@ime.ac.cn

基金项目：科技创新2030重大项目(2022ZD0208700)

Foundation Item: STI 2030-Major Projects (2022ZD0208700)

## 1　引言

人工智能正成为推动人类进入智能时代的决定性力量[1]，在自动驾驶、医疗救助、政府管理等安全攸关应用中，人工神经网络(Artificial Neural Networks, ANN)面临着以对抗攻击为代表的各种攻击威胁。对抗攻击来源于人脑与机器智能的识别差异，可定义为通过对输入注入人类难于辨识的扰动来欺骗神经网络模型，使其达到高概率的错误输出。愚弄模型[2]、操纵语音助手[3]、错误过滤垃圾信息[4]等案例层出不穷，可能造成泄露关键数据，甚至决策错误[5]，严重影响系统应用的准确性、机密性和完整性。对抗攻击已经成为影响深度学习模型成功的最大挑战之一。

类脑芯片在结构上多采用脉冲化的神经网络(Spiking Neural Networks, SNN)，具有高稀疏性、低功耗等特点，已成为第3代神经网络的代表[6]，有望大规模地用于视觉分类等低延迟目标识别任务。SNN的固有结构使层间数据具备天然的稀疏性[7]，使其比ANN有显著的对抗鲁棒性提升。国内外相关研究主要聚焦在算法模型分析层面，Sharmin等人[8]提出了一个简单的对抗攻击框架，并指出通过泊松编码的输入离散化和泄漏-积累-发放(Leaky Integrate-and-Fire, LIF)神经元的非线性激活是SNN鲁棒性的来源。El-allami等人[7]研究了SNN对不同神经元放电电压阈值和时间窗边界值条件下的攻击鲁棒性。Kundu等人[9]从权重、泄漏、阈值和时间步长等方面评估了视觉几何组(Visual Geometry Group, VGG)网络结构的鲁棒性。Kim等人[10]在低延迟训练中比较了两种编码的鲁棒性、能量效率、准确性等特性。上述研究中均没有考虑算法到电路的映射，而类脑芯片在实际应用中，必然要通过重要参数的量化来配置电路单元，进而完成网络推理。

本文通过在对抗攻击前向推理的加载顺序之前加入感知量化函数，从而模拟量化压缩后的权重在推理过程中的精度损失，将感知量化框架与对抗攻击训练/推理过程相融合，能够对网络部署到硬件的量化过程中鲁棒性损失进行度量。主要贡献如下：

(1) 针对基于梯度攻击的对抗攻击算法，在不同网络拓扑上构建了适用于对抗攻击训练的基于替代梯度的SNN模型和对抗样本。

(2) 创新性地提出一种权重感知量化的方法，建立感知量化与对抗攻击训练/推理相融合的评估框架，在不同脉冲编码和网络参数组合条件下，通过引入攻击前后推理精度损失差、层间脉冲活动和脉冲信噪比等度量依据，解释量化稀疏性与对抗鲁棒性的联系。

(3) 基于类脑芯片模拟器完成部署和对抗推理，对SNN对抗鲁棒性实现了全栈评估。

具体结构安排如下：第2节介绍支持对抗鲁棒性分析的SNN算法与对抗样本生成，第3节介绍本文提出的权重量化方法和评估框架，第4节通过实验分析了量化前后不同对抗鲁棒性评估依据的差异，并在实际类脑芯片中进行了硬件验证，第5节总结全文。

## 2　支持对抗鲁棒性分析的SNN算法与攻击建模

### 2.1　基于反向传播和替代梯度的SNN算法

SNN训练常用到的算法有3种：(1)ANN转换SNN方法[11]；(2)基于反向传播和替代梯度的方法(Spatio-Temporal Back-Propagation, STBP)[12]；(3)无监督脉冲时间依赖可塑性 (Spike-Timing-Dependent Plasticity, STDP)及其变体方法[13]。由于转换SNN的算法需要较大的计算资源和调试优化时间，而STDP没有梯度反向传播的过程，且损失函数不可导，不易生成对抗样本，因此，使用STBP训练算法，将替代梯度算法与时间反向传播算法融合，解决了SNN层间脉冲反向传播不可导的问题。反向传播主要利用链式求导法则计算损失函数相对于网络参数的梯度。导数的链式法则是微积分的基本规则，允许通过应用其组成函数的导数来计算复合函数的导数。SNN时空展开的链式求导结果如式(1)

$$\frac{\partial L}{\partial W_l}=\begin{cases}\displaystyle\sum_t\left(\frac{\partial L}{\partial O_l^t}\frac{\partial O_l^t}{\partial U_l^t}+\frac{\partial L}{\partial U_l^{t+1}}\frac{\partial U_l^{t+1}}{\partial U_l^t}\right)\frac{\partial U_l^t}{\partial W_l}, & l:\text{hidden}\\ \dfrac{\partial L}{\partial U_l^T}\dfrac{\partial U_l^T}{\partial W_l}, & l:\text{output}\end{cases}\tag{1}$$

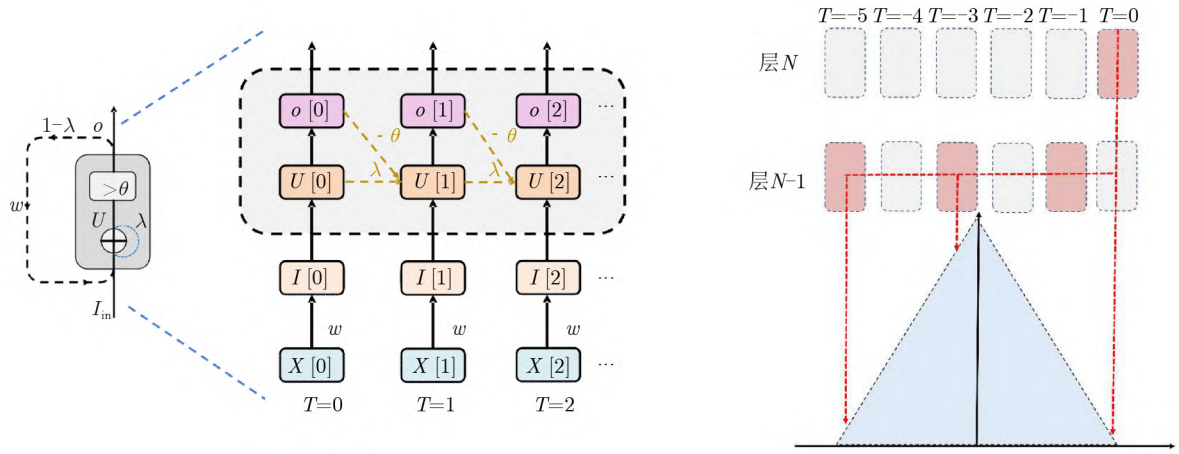
$L$表loss函数，$O$代表神经元输出，$U$代表神经元电压，$t$代表当前的第$t$个单位时间，$l$代表网络的第$l$层，$W$代表对应节点上的权重。

由于SNN中隐藏层的LIF神经元只有在膜电位超过放电阈值时才产生脉冲输出，导致不可微。为了利用标准的基于反向传播的优化程序，本文使用了替代梯度技术。参考利用导数中峰值时间信息的梯度逼近函数进行替代梯度计算的方法[12]。线性函数替代梯度的数学公式如式(2)

$$\frac{\partial o^t}{\partial u^t}=\max\left\{0,1-\left|\frac{u^t}{\theta}-1\right|\right\}\tag{2}$$

$o^t$和$u^t$为时间步长$t$时的输出峰值和膜电位。$\theta$为神经元阈值电压。

图1(a)为同一神经元在时间上展开后的链式求




(a) 链式求导 (b) 反向传播

图 1 链式求导法则在时间和空间上展开后的求导原理

导法则原理，在一轮训练权重更新的过程中，对应每个单位时间$T$上的权重信息累计得到这一轮训练权重更新的最终数据。图1(b)为不同层神经元在时间上展开后的求导原理。

### 2.2 对抗攻击建模

Szegedy等人[14]于 2013 年发表了第1篇深度神经网络的对抗性攻击论文，并创造了术语“对抗性示例”。随后Goodfellow等人[15]也提出了具有单个梯度步长的对抗性示例：快速梯度符号方法(Fast Gradient Sign Method, FGSM)。后续出现更多的对抗算法，如R + FGSM通过随机化步骤来增强攻击[16]，投影梯度下降(Projected Gradient Descent, PGD)则采取多个较小的步骤迭代对FGSM进行了改进[17]。SNN对抗攻击包括：(1)对抗样本的生成；(2)对抗样本的注入。首先将原始样本叠加上权重和扰动因子的乘积后生成新的样本，然后将样本重新注入网络进行训练，更改要攻击的标签后进行反向传播和梯度下降，训练多轮直到最终分类结果修改为攻击目标效果后，导致SNN做出不正确的预测。

SNN对抗攻击的建模首先需要确定一个数据集$(x, y_{\text{true}})$个分类模型$h$，其中$x$为干净的图像，$y_{\text{true}}$为正确标签，对抗攻击的概念是找到一个输入$x_{\text{adv}}$，使得$x$和$x_{\text{adv}}$无法区分开，但分类模型$h$错误地区分了$x_{\text{adv}}$，即在错误标签上产生高概率输出。在本文中，主要考虑FGSM生成$x_{\text{adv}}$的方法，如图2。其中的Loss为SNN替代梯度训练过程中前向传播后的损失函数，Grad为训练过程中反向传播后获得的梯度信息，Adv_Grad为保存中间梯度信息的节点，Adv_Loss为原始样本推理精度减去对抗样本推理精度的差值，Embedding Weight是指将训练最后一轮的权重信息重新加载到推理过程中。

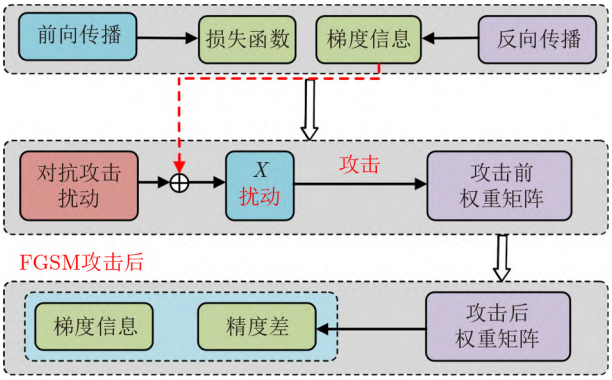



图 2 FGSM攻击原理

#### 2.2.1 FGSM攻击

FGSM是产生对抗性干扰的最基本和最广泛的方法，分为非目标和目标两种方式。

非目标FGSM如式(3)

$$x^{\text{adv}} = x + \epsilon \text{sign}\left(\nabla_x J\left(x, y_{\text{true}}\right)\right) \tag{3}$$

FGSM如式(4)

$$x^{\text{adv}} = x - \epsilon \text{sign}\left(\nabla_x J\left(x, y_{\text{random}}\right)\right) \tag{4}$$

这里$\epsilon$指扰动量，通常$\epsilon$远小于$x$，$\nabla_x J$是损失函数相对于原始干净数据的梯度。

#### 2.2.2 对抗样本生成

假设$\boldsymbol{X}$是32×32矩阵，卷积的核大小$\boldsymbol{W}$是3×3。卷积是以1的填充和1的步幅执行。然后输出$\boldsymbol{Y}$将保持32×32的大小($\boldsymbol{Y}$维度=$\boldsymbol{X}$维度 + (2×padding)−$\boldsymbol{W}$维度+ 1)。当$\boldsymbol{Y} = \boldsymbol{X} \circledast \boldsymbol{W}$时可以得到：$\dfrac{\partial \boldsymbol{J}}{\partial \boldsymbol{X}} = \dfrac{\partial \boldsymbol{J}}{\partial \boldsymbol{Y}} \circledast \boldsymbol{W}^{180\text{rotated}}$。由于编码方式不同，$\boldsymbol{X}$和$\boldsymbol{X}_{\text{encode}}$的差异也不同，导致最终不同编码方式下求导得到的$\partial \boldsymbol{J}/\partial \boldsymbol{X}$也存在很大差异，这会影响到最终对抗攻击的效果，因此本文作了近似假设，如式(5)和式(6)

$$\boldsymbol{X}=\boldsymbol{X}_{\text{encode}}\begin{cases}=\boldsymbol{X}_{\text{direct}}\\ \approx\boldsymbol{X}_{\text{rate}}\\ \approx\boldsymbol{X}_{\text{latency}}\\ \approx\boldsymbol{X}_{\text{phase}}\end{cases}\tag{5}$$

$$\boldsymbol{X}=\begin{pmatrix}X_{1,1}&X_{1,2}&\cdots&X_{1,32}\\X_{2,1}&X_{2,2}&\cdots&X_{2,32}\\\vdots&\vdots&\ddots&\vdots\\X_{32,1}&X_{32,2}&\cdots&X_{32,32}\end{pmatrix},$$

$$\boldsymbol{W}=\begin{pmatrix}W_{1,1}&W_{1,2}&W_{1,3}\\W_{2,1}&W_{2,2}&W_{2,3}\\W_{3,1}&W_{3,2}&W_{3,3}\end{pmatrix}\tag{6}$$

经过补丁操作以后的结果如式(7)

$$\begin{aligned}Y_{1,1}&=W_{2,2}X_{1,1}+W_{2,3}X_{1,2}+W_{3,2}X_{2,1}+W_{3,3}X_{2,2}\\Y_{1,2}&=W_{2,1}X_{1,1}+W_{2,2}X_{1,2}+W_{2,3}X_{1,3}+W_{3,1}X_{2,1}\\&\quad+W_{3,2}X_{2,2}+W_{3,3}X_{2,3}\\&\vdots\\Y_{32,32}&=W_{1,1}X_{31,31}+W_{1,2}X_{31,32}+W_{2,1}X_{32,31}\\&\quad+W_{2,2}X_{32,32}\end{aligned}\tag{7}$$

损失函数用$\boldsymbol{J}$表示，$\boldsymbol{J}$对$\boldsymbol{X}$的梯度也是一个32 × 32的矩阵，每个元素用链式法则描述，并表示为矩阵的形式为式(8)。假设输入图像表示为$\boldsymbol{X},\boldsymbol{W}$ 表示卷积层的权重矩阵，$\boldsymbol{Y}$表示卷积运算的输出。合理进行以下近似假设如式(9)和式(10)

$$\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}}=\begin{pmatrix}0&0&0&\cdots&0&0\\0&\dfrac{\partial\boldsymbol{J}}{\partial Y_{1,1}}&\dfrac{\partial\boldsymbol{J}}{\partial Y_{1,2}}&\cdots&\dfrac{\partial\boldsymbol{J}}{\partial Y_{1,32}}&0\\0&\dfrac{\partial\boldsymbol{J}}{\partial Y_{2,1}}&\dfrac{\partial\boldsymbol{J}}{\partial Y_{2,2}}&\cdots&\dfrac{\partial\boldsymbol{J}}{\partial Y_{2,32}}&0\\\vdots&\vdots&\vdots&\ddots&\vdots&\vdots\\0&\dfrac{\partial\boldsymbol{J}}{\partial Y_{32,1}}&\dfrac{\partial\boldsymbol{J}}{\partial Y_{32,2}}&\cdots&\dfrac{\partial\boldsymbol{J}}{\partial Y_{32,32}}&0\\0&0&0&\cdots&0&0\end{pmatrix}\circledast\begin{pmatrix}W_{3,3}&W_{3,2}&W_{3,1}\\W_{2,3}&W_{2,2}&W_{2,1}\\W_{1,3}&W_{1,2}&W_{1,1}\end{pmatrix}\tag{8}$$

$$\boldsymbol{X}_{\text{conv1}}\approx\boldsymbol{X}_{\text{encode}}\circledast\boldsymbol{W}_{\text{conv1}}\tag{9}$$

$$\frac{\partial\boldsymbol{X}_{\text{conv1}}}{\partial\boldsymbol{X}_{\text{encode}}}=\boldsymbol{W}_{\text{conv1}}^{180\text{rotated}}\tag{10}$$

利用替代梯度技术通过网络反向传播损失而得到的。然后用式(9)求得对抗扰动，可以得到

$$\begin{aligned}\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}}&\approx\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}_{\text{encode}}}=\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}_{\text{conv1}}}\times\frac{\partial\boldsymbol{X}_{\text{conv1}}}{\partial\boldsymbol{X}_{\text{encode}}}\\&=\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}_{\text{conv1}}}\circledast\boldsymbol{W}_{\text{conv1}}^{180\text{rotated}}\end{aligned}\tag{11}$$

$\boldsymbol{W}_{\text{conv1}}^{180\text{rotated}}$是$\boldsymbol{W}_{\text{conv1}}$旋转180°。将式(11)的结果代入式(3)和式(4)中，即可得到FGSM攻击的对抗扰动值。如式(12)

$$\begin{aligned}\epsilon\times\text{sign}\left(\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}}\right)&\approx\epsilon\times\text{sign}\left(\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}_{\text{encode}}}\right)\\&=\epsilon\times\text{sign}\left(\frac{\partial\boldsymbol{J}}{\partial\boldsymbol{X}_{\text{conv1}}}\circledast\boldsymbol{W}_{\text{conv1}}^{180\text{rotated}}\right)\end{aligned}\tag{12}$$

具体攻击算法的迭代次数即可得到最终的对抗样本。由于对抗攻击过程只在前向推理过程中进行，但是对抗样本的生成过程，需要利用反向传播的权重数据。因此利用权重数据和FGSM算法生成的对抗样本注入网络来完成攻击。更多算法建模和分析可参考文献[18]。

## 3　权重量化方法与评估框架

### 3.1　量化稀疏

量化是将神经网络中的浮点数参数和激活值转换为低位整数或定点数的过程，通常会牺牲一定的模型精度以换取更小的模型尺寸和更低的计算量，从而使得神经网络可以在资源有限的场景中运行。深度神经网络模型中，通常权重由float32浮点数量化为int8, int4等定点数，减少模型的可表示空间大小。当 SNN 的权重被量化时，由于权重变得非常小，在某些情况下甚至为0。因此，连接到这些小或零值权重的神经元将接收到非常少的输入，从而造成网络输出的稀疏性。量化稀疏性有助于防止过度拟合训练数据。但过度的量化稀疏会导致大多数神经元沉默，影响网络推理精度和训练效果。因此，权重量化方法需要综合考虑量化比特选择、量化函数，以及量化和对抗框架的执行顺序等要素。

### 3.2　权重量化方法

在机器学习中，模型量化通常包括两种方式：后训练量化和感知量化。后训练量化是在模型训练完成后对模型参数进行量化，可将原本使用浮点数表示的参数转换为使用更少的比特位表示。感知量化是在模型训练期间对模型参数进行量化，可在不降低模型精度的情况下，显著降低模型参数的存储和计算开销。

为了使量化比特宽度范围更广泛，本文使用K-bit量化函数来调节量化的bit位宽。如式 (13)，[.]代表round操作，将输入张量的每个元素四舍五入到最近的整数

$$q_k\left(x\right)=2\left(\frac{\left[\left(2^k-1\right)\left(\dfrac{x+1}{2}\right)\right]}{2^k-1}-\frac{1}{2}\right)\tag{13}$$

### 3.3　量化函数

根据2.1节的描述，SNN中每一层输出脉冲不可导，故而在训练过程中引入了替代梯度。量化的

过程是在训练的每一轮中插入伪节点，在训练过程中对权重信息进行量化并存储到权重伪节点。在训练的前向传播中利用伪节点的权重信息进行推理，然后求得该轮的损失值loss后，进行反向传播并更新权重信息。在下一轮训练中重复这种执行顺序，直到达到适合的推理精度，并保存此时的权重信息。在定义SNN前向推理网络时，也要将感知量化函数插入到同样的地方，需加载训练过程最后一轮保存的权重信息，在推理时对该权重进行裁剪量化和存储，利用伪节点权重做前向推理即可得到真实的量化后的网络推理结果。

感知量化函数主要包括以下4个内容：

(1) 初始化网络：定义变量，定义网络节点用于索引网络中每一层参数，网络初始化；

(2) 构建参数张量节点(伪量化节点)：用于保存前向推理过程中量化参数；

(3) 裁剪函数：将浮点权重按层进行裁剪，利用式(17)完成裁剪，用于k-bit量化；

(4) restore函数：用于存储将反向过程中原始权重参数值还原到原始节点。

### 3.4 融合对抗攻击与量化的训练/推理框架

在具体模型的实现中，需要将对抗攻击与量化的训练和推理框架进行融合，算法1为加入了感知量化算法的训练框架，算法2为感知量化与对抗攻击融合的推理框架。

**算法1 加入感知量化算法的训练框架**

(1) 输入：数据集$T$，epoch，攻击强度参数($\varepsilon$,$k$)Timestep，权重量化比特$K$，编码式，网络架构，攻击类型
(2) 加载：感知训练好的模型.pt文件
(3) 初始化：网络初始化，参数初始化，攻击模型初始化，量化初始化
(4) 损失函数，优化器选择
(5) 执行量化函数
(6) for $T$ do:
(7) 采样 batch ($x$, $y$)
(8) 目标模型的对抗攻击前向传播
(9) 累计adv_loss
(10) 攻击模型的对抗攻击前向传播
(11) 累计adv_loss
(12)restore函数
(13)计算最终的adv_loss

## 4 实验结果与分析

### 4.1 量化后的对抗鲁棒性分析

神经网络算法部署在类脑硬件上时都要进行权重裁剪，因此量化权重必然会引起推理过程的稀疏性变化，进而影响对抗鲁棒性。本节通过3.2节的方法对VGG5和VGG9进行了4 bit的权重量化，并使用对抗攻击融合的推理框架，分析对抗鲁棒性的差异。(1)VGG5推理精度损失和对抗鲁棒性差异。表1展示了VGG5在最优推理精度时(参数为(0.4,1.0,0.5))，量化前后不同攻击强度下的推理精度损失，损失越大表明对抗鲁棒性越差。Q1代表权重量化前，Q2代表权重量化后。量化前后推理精度损失差在可接受范围内。组合参数的3个数值分别代表conv层阈值电压，fc层阈值电压和泄露因子λ。I代

**算法2 感知量化与对抗攻击融合的推理框架**

(1) 输入：数据集训练集$D$，测试集$T$, epoch, Timestep，权重量化比特$K$，编码式，网络架构
(2) 初始化：网络初始化，参数初始化，量化初始化
(3) 损失函数，优化器选择
(4) for epoch do
(5) for $D$ do:
(6) 执行量化函数
(7) 从$D$中采样 batch ($x$, $y$)
(8) 网络前向传播
(9) 求单轮loss
(10) 训练loss更新
(11) 网络反向传播
(12) restore函数
(13) 调整学习率
(14) 执行量化函数
(15) for $T$ do:
(16) 从$T$中采样 batch ($x$, $y$)
(17) 网络前向传播
(18) restore函数
(19) 测量推理精度
(20) 保存训练模型到pt文件

**表 1 VGG5量化前后不同攻击强度下的推理精度损失(%)**

| 组合参数 | | 直接编码 | | | 速率编码 | | | 相位编码 | | | 延迟编码 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | I | II | III | I | II | III | I | II | III | I | II | III |
| 0.4,1.0,0.5 | $Q_1$ | 7.41 | 13,42 | 34,52 | 3.21 | 8.04 | 13,81 | 2.32 | 5.56 | 16.33 | 1.82 | 7.81 | 25.88 |
| | $Q_2$ | 6.37 | 11.65 | 22.44 | 0.89 | 3.24 | 5.37 | 0.89 | 3.24 | 5.37 | 1.73 | 7.74 | 22.10 |

表FGSM_2攻击，II代表FGSM_8攻击，III代表FGSM_16攻击。4种编码的对抗鲁棒性在量化后都大于量化前。直接编码的鲁棒性最差。为了更深入分析量化前后的对抗鲁棒性规律，在更深层的VGG9网络上针对更多组合参数进行了实验。(2)VGG9推理精度损失表2为在FGSM攻击下，不同组合方式在权重量化前后推理精度损失的对比变化幅度表征了不同参数下的结果波动范围。

由表2中结果可以看到：

**现象1**　在相同($\theta$,$\lambda$)组合参数的情况下，无论哪种攻击强度，直接编码在4种编码方法中都显现出最差的鲁棒性。

**现象2**　对于3种稀疏编码(速率编码、相位编码和延迟编码)在低阈值电压($\theta$=0. 4)时，量化后对抗鲁棒都高于量化前的对抗鲁棒性。升高阈值电压($\theta$=1, $\theta$ =0.6和$\theta$=0. 8)时，对抗鲁棒性分布无规律性。各种编码方式都在($\theta$=0. 4)时表现出最好的鲁棒性，量化前只有局部最佳阈值点。

**现象3**　量化后随着组合参数的变化，在低强度攻击下直接编码的adv_loss最大差异为1.65%，在高强度攻击下为21.21%。速率编码对低强度攻击最大差异为2.01%，对于高强度攻击为9.34%。相位编码对低强度攻击最大差异为1.46%，对于高强度攻击为4.94%。延迟编码对低强度攻击最大差异为1.36%，对于高强度攻击为9.4%。较于量化前对固有参数变化时的推理精度损失变化幅度均有所增大，均值为5.79%，涨幅为73.23%。

(1) VGG9层间脉冲活动。图3(a)—图3(d)分别代表了在没有对抗攻击的情况下，量化前后不同编码方式和参数下的脉冲活动，可以看到量化前后脉冲活动基本保持在同一数量级。

以FGSM-2攻击为例，表3为不同组合参数和编码方式在权重量化前后平均层间脉冲活动的差异(个别数值由于训练失败缺失)。可以看到，直接编码、相位编码和延迟编码结的层间脉冲活动均值的在量化前后变化幅度明显变大，平均涨幅为51.6%，而速率编码则稍有减少。

(2)脉冲信噪比。对抗样本和原始样本会因攻击强度的不同存在不同的噪声强度。根据输入数据传递的时空特性，以及输入编码后数据积累用于脉冲发放的特性，将原始样本和对抗样本在推理过程中层间的原始样本脉冲活动数SA1与SA1-SA2的差值的比值定义为对抗样本与原始样本的脉冲信噪比(Signal to Noise Ratio, SNR)，其倒数为噪信比(Noise to Signal Ratio, NSR)，如式(14)

$$\mathrm{SNR}=\frac{1}{\mathrm{NSR}}=\frac{\mathrm{SA1}}{\mathrm{SA1}-\mathrm{SA2}} \tag{14}$$

在分析中使用脉冲信噪比SNR(或NSR)，可以将影响层间脉冲稀疏性的因素包含在内，更好地反映加入干扰后网络是如何受影响而做出错误判断的，同时还考虑时间步$T$的影响。攻击强度 $\varepsilon/255$ 代表对抗样本相对于原始样本在每个图像通道上被改变的幅度，一旦数据被编码送到每个神经元后，对抗样本的附加干扰就会被随机加到每个神经元上参与神经的积累电压与发放脉冲过程，从而影响最终的分类结果。

图4为加入对抗攻击后，量化后不同编码方式和组合参数作用下的NSR的结果。可以看到：

(1)在相同编码方式下，随着阈值变化，NSR的波动趋势一致。

(2)4种编码方式都存在NSR波动幅度相对最大的阈值点或波动幅度最小的阈值点，直接编码、速率编码和延迟编码都是$\theta$=(0.4,1)NSR波动幅度都最

**表 2　VGG9量化前后不同攻击强度下的推理精度损失(%)**

| 组合参数 | | 直接编码 | | | 速率编码 | | | 相位编码 | | | 延迟编码 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | I | II | III | I | II | III | I | II | III | I | II | III |
| 0.4,1.0,0.5 | $Q_1$ | 5.84 | 21.61 | 46.63 | 2.08 | 6.3 | 12.16 | 1.71 | 7.48 | 16.10 | 0.77 | 8.07 | 26.71 |
| | $Q_2$ | 3.76 | 7.46 | 23.21 | 0.41 | 2.92 | 6.94 | 0.62 | 4.70 | 12.73 | 0.44 | 5.27 | 19.59 |
| 0.6,1.0,0.5 | $Q_1$ | 5.24 | 19.31 | 41.23 | 2.38 | 8.43 | 12.01 | 2.1 | 7.71 | 16.75 | 2 | 7.23 | 24.8 |
| | $Q_2$ | 5.41 | 11.31 | 26.69 | 1.94 | 7.81 | 16.28 | 1.07 | 5.93 | 13.94 | 0.72 | 3.27 | 12.32 |
| 0.8,1.0,0.5 | $Q_1$ | 5.56 | 19.07 | 38.87 | 2.13 | 7.93 | 11.44 | 2.7 | 8.15 | 15.61 | 1.89 | 6.21 | 19.55 |
| | $Q_2$ | 4.45 | 13.41 | 42.92 | 2.42 | 6.96 | 13.07 | 2.08 | 7.87 | 17.67 | 1.80 | 7.43 | 21.72 |
| 1,0,1.0,0.5 | $Q_1$ | 5.03 | 17.14 | 30.14 | 3.01 | 8.23 | 12.75 | 1.86 | 7.05 | 15.15 | – | – | – |
| | $Q_2$ | 4.60 | 13.37 | 44.31 | 1.96 | 6.09 | 12.44 | 2.08 | 7.86 | 17.26 | 1.06 | 5.43 | 17.63 |
| $Q_1$变化幅度 | | 0.81 | 4.47 | 16.49 | 0.93 | 2.13 | 1.31 | 0.99 | 1.10 | 1.60 | 1.23 | 1.86 | 7.16 |
| $Q_2$变化幅度 | | 1.65 | 5.95 | 21.10 | 2.01 | 4.89 | 9.34 | 1.46 | 3.17 | 4.94 | 1.36 | 4.16 | 9.40 |

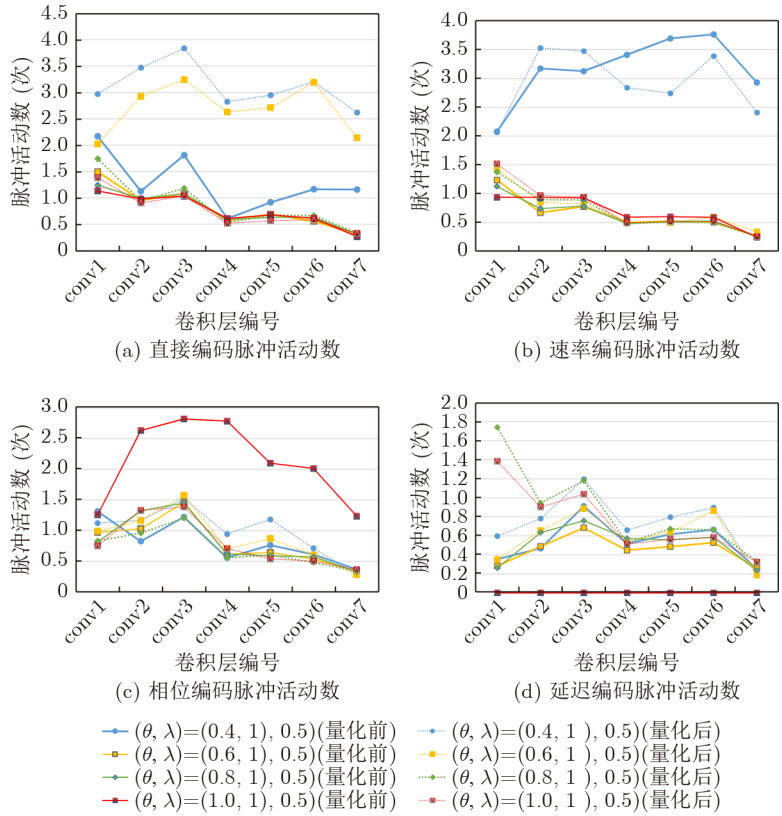



图 3 量化前后不同组合方式下的脉冲活动

**表 3 量化前后不同攻击强度下的层间脉冲活动均值**

| 组合参数 | | 直接编码 | 速率编码 | 相位编码 | 延迟编码 |
|---|---|---|---|---|---|
| 0.4,1.0,0.5 | $Q_1$ | 1.2758 | 3.1738 | – | 0.5347 |
| | $Q_2$ | 3.1236 | 2.8196 | 0.9885 | 0.7321 |
| 0.6,1.0,0.5 | $Q_1$ | 0.8081 | 0.6398 | 0.7925 | 0.4854 |
| | $Q_2$ | 2.6948 | 0.7227 | 0.8795 | 0.5869 |
| 0.8,1.0,0.5 | $Q_1$ | 0.7804 | 0.6343 | 0.8082 | 0.5144 |
| | $Q_2$ | 0.8659 | 0.7018 | 0.7042 | 0.4158 |
| 1,0,1.0,0.5 | $Q_1$ | 0.7578 | 0.6913 | 2.1092 | – |
| | $Q_2$ | 0.7585 | 0.7463 | 0.7939 | 0.4784 |
| $Q_1$变化幅度 | | 0.5180 | 2.5395 | 0.1752 | 0.0493 |
| $Q_2$变化幅度 | | 2.3651 | 2.1178 | 0.2843 | 0.3162 |

小，延迟编码在$\theta$=(0.8,1)和$\theta$=(1,1)上NSR波动幅度更大，相位编码不同阈值下幅度相当。

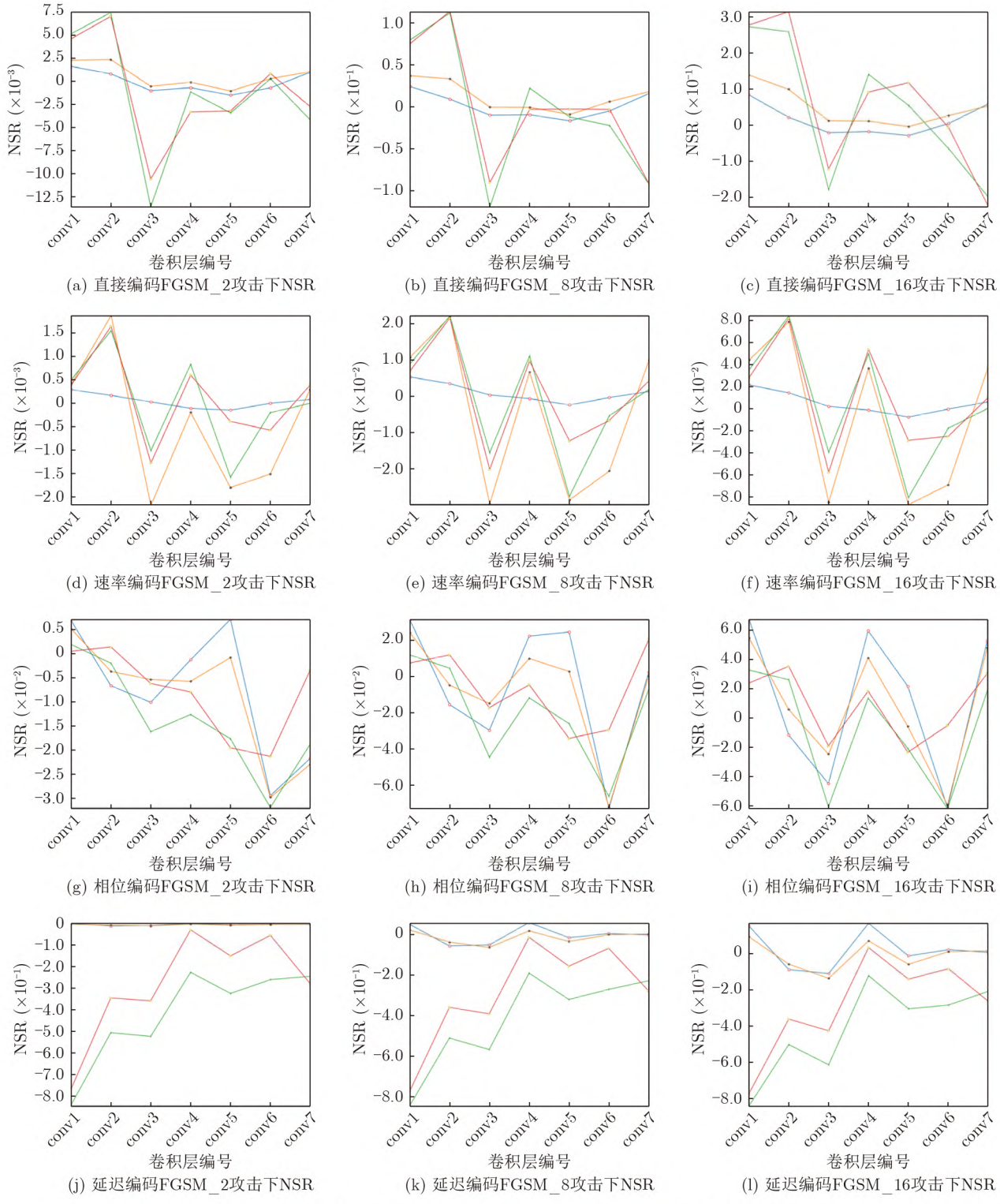
(3)同一种组合参数，不同编码方式NSR的差别数量级不同，本文以1, 0.1, 0.01, 0.001和0.000 1为数量级的衡量依据来描述NSR的差异，结果如表4。无论在何种攻击强度下延迟编码NSR值都最大，这说明对抗样本和原始样本影响层间脉冲的发放程度更大。

**分析1** 在同一种编码方式下，对抗鲁棒性存在最佳阈值点，且最佳阈值点的NSR波动相对其他阈值点低。当调整单一参数时，总会存在一个极点使对抗鲁棒性达到相对最佳，而低于或者超过这个设定值时，就会出现鲁棒性的下降的趋势。且NSR越低对抗样本对层间脉冲改变量越小，此时鲁棒性越好。

**分析2** 不同编码方式的$\boldsymbol{X}_{\text{clean}}$影响了$\boldsymbol{W}$的分布，中间存在替代梯度和权重量化两个近似过程，导致了$\boldsymbol{W}$是模拟混淆值，所以编码越稀疏，模型越具有较高的鲁棒性。

由上述实验结果的综合分析可以得出如下结论：

(1)阈值增加，稀疏编码，权重量化都可以作为层间脉冲稀疏化的因素。其加剧稀疏化的趋势




(a) 直接编码FGSM_2攻击下NSR　(b) 直接编码FGSM_8攻击下NSR　(c) 直接编码FGSM_16攻击下NSR

(d) 速率编码FGSM_2攻击下NSR　(e) 速率编码FGSM_8攻击下NSR　(f) 速率编码FGSM_16攻击下NSR

(g) 相位编码FGSM_2攻击下NSR　(h) 相位编码FGSM_8攻击下NSR　(i) 相位编码FGSM_16攻击下NSR

(j) 延迟编码FGSM_2攻击下NSR　(k) 延迟编码FGSM_8攻击下NSR　(l) 延迟编码FGSM_16攻击下NSR

图 4 量化后不同编码方式下的脉冲NSR

是，阈值增加大于权重量化bit降低大于稀疏编码。当稀疏因素单一时，对抗鲁棒性趋势是单调的，而当稀疏因素达到3种时，训练的模型有很大的脆弱性，导致对抗鲁棒性下降。

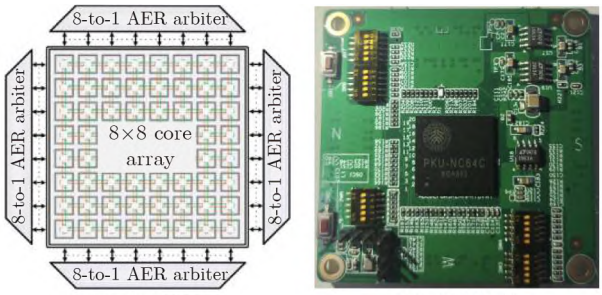
(2)单一的参数因素与对抗鲁棒性之间并不是线性关系。关于模型对抗鲁棒性的比较必须在固定系统模型参数的前提下。当调整单一参数时，总会存在一个极点使对抗鲁棒性达到相对最佳，而低于或者超过这个设定值时，就会出现鲁棒性的下降的趋势。

### 4.2 类脑芯片硬件验证

为了验证本文的权重量化方法与实际硬件上的兼容性，将提出的SNN框架在权重量化后部署到类脑芯片的硬件中。实验平台为北京大学PKU-NC64C[19]类脑芯片，如图5。该芯片是一款多核脉冲神经网络类脑芯片，芯片采用2D Mesh NOC架构，集成了64k个随机LIF神经元和64M神经突触，可实现脉冲神经网络模型的高效推断功能。芯片配套设计工具链包括完整的芯片行为硬件模拟器，基于硬件模拟器进行了攻击验证实验。因芯片可部署网络的规模限制，芯片部署实验采用模型为VGG5，数据集CIFAR-10，编码为直接编码，在不加攻击的情况下，推理精度为83%，加入不同程度的攻击后推理精度损失如表5，由于训练方法的差异，映射过程存在一定转换误差，但是不同攻击强度下的趋势没有大的改变。这也验证了本研究算法的可用性和迁移性。

## 5 结束语

针对脉冲神经网络对抗鲁棒性分析方法缺乏对网络部署到硬件中的量化过程的损失度量的局限性，本文提出一种将对抗攻击与权重感知量化相结合的训练与推理框架，通过将推理精度损失差、层间脉冲活动和脉冲信噪比作为度量依据，从而衡量量化前后的鲁棒性水平和变化趋势。实验结果表明直接编码的对抗鲁棒性最差，且提出稀疏化的因素对鲁棒性影响相关度为：阈值增加大于权重量化bit降低大于稀疏编码，及单一稀疏性因素对于对抗鲁棒性存在局部最优解的结论，量化方法和评估框架在实际的类脑芯片硬件中得到验证。研究揭示了SNN在量化过程中的部分对抗脆弱性来源和机理，能够帮助研究人员构建更安全可靠的神经形态系统。




(a) 内部阵列 (b) 芯片外观

图 5 PKU-NC64C 芯片

**表 4 NSR数量级差异**

| | 直接编码 | 速率编码 | 相位编码 | 延迟编码 |
|---|---|---|---|---|
| 低强度攻击 | 0.001 | 0.0001 | 0.001 | 1 |
| 中强度攻击 | 0.01 | 0.01 | 0.01 | 1 |
| 高强度攻击 | 0.1 | 0.001 | 0.01 | 1 |

**表 5 软件和PKU-NC64C硬件映射adv_loss对比(%)**

| 验证类型 | FGSM_2 | FGSM_8 | FGSM_16 |
|---|---|---|---|
| 算法级 | 5.42 | 12 | 23.27 |
| 硬件级 | 7.02 | 10.0 | 24.52 |

## 参 考 文 献

[1] 谭铁牛: 人工智能的历史、现状和未来[EB/OL]. https://www.cas.cn/zjs/201902/t20190218_4679625.shtml, 2019.
Tan Tieniu. The historyk, present and future of artificial intelligence. Chinese Academy of Sciences[EB/OL]. https://www.cas.cn/zjs/201902/t20190218_4679625.shtml, 2019.

[2] LIU Aishan, LIU Xianglong, FAN Jiaxin, *et al*. Perceptual-sensitive GAN for generating adversarial patches[C]. The 33rd AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, Honolulu, USA, 2019: 127. doi: 10.1609/aaai.v33i01.33011028.

[3] ZHANG Guoming, YAN Chen, JI Xiaoyu, *et al*. DolphinAttack: Inaudible voice commands[C]. The 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, USA, 2017: 103–117. doi: 10.1145/3133956.3134052.

[4] WARREN T. Microsoft's Outlook spam email filters are broken for many right now[EB/OL]. https://www.theverge.com/2023/2/20/23607056/microsoft-outlook-spam-email-filters-not-working-broken, 2023.

[5] 董庆宽, 何浚霖. 基于信息瓶颈的深度学习模型鲁棒性增强方法[J]. 电子与信息学报, 2023, 45(6): 2197–2204. doi: 10.11999/JEIT220603.
DONG Qingkuan and HE Junlin. Robustness enhancement method of deep learning model based on information bottleneck[J]. *Journal of Electronics & Information Technology*, 2023, 45(6): 2197–2204. doi: 10.11999/JEIT220603.

[6] WEI Mingliang, YAYLA M, HO S Y, *et al*. Binarized SNNs: Efficient and error-resilient spiking neural networks through binarization[C]. 2021 IEEE/ACM International Conference on Computer Aided Design, Munich, Germany, 2021: 1–9. doi: 10.1109/ICCAD51958.2021.9643463.

[7] EL-ALLAMI R, MARCHISIO A, SHAFIQUE M, *et al*. Securing deep spiking neural networks against adversarial attacks through inherent structural parameters[C]. 2021 Design, Automation & Test in Europe Conference & Exhibition, Grenoble, France, 2021: 774–779. doi: 10.23919/DATE51398.2021.9473981.

[8] SHARMIN S, RATHI N, PANDA P, *et al*. Inherent adversarial robustness of deep spiking neural networks: Effects of discrete input encoding and non-linear activations[C]. The 16th European Conference, Glasgow, UK, 2020: 399–414. doi: 10.1007/978-3-030-58526-6_24.

[9] KUNDU S, PEDRAM M, and BEEREL P A. HIRE-SNN: Harnessing the inherent robustness of energy-efficient deep spiking neural networks by training with crafted input noise[C]. 2021 IEEE/CVF International Conference on Computer Vision, Montreal, Canada, 2021: 5209–5218. doi: 10.1109/ICCV48922.2021.00516.

[10] KIM Y, PARK H, MOITRA A, *et al*. Rate coding or direct coding: Which one is better for accurate, robust, and energy-efficient spiking neural networks?[C]. 2022 IEEE International Conference on Acoustics, Speech and Signal Processing, Singapore, 2022: 71–75. doi: 10.1109/ICASSP43922.2022.9747906.

[11] O'CONNOR P and WELLING M. Deep spiking networks[J]. arXiv preprint arXiv: 1602.08323, 2016.

[12] RATHI N, SRINIVASAN G, PANDA P, *et al*. Enabling deep spiking neural networks with hybrid conversion and spike timing dependent backpropagation[C]. The 8th International Conference on Learning Representations, Addis Ababa, Ethiopia, 2020.

[13] TAVANAEI A and MAIDA A. BP-STDP: Approximating backpropagation using spike timing dependent plasticity[J]. *Neurocomputing*, 2019, 330: 39–47. doi: 10.1016/j.neucom.2018.11.014.

[14] SZEGEDY C, ZAREMBA W, SUTSKEVER I, *et al*. Intriguing properties of neural networks[C]. The 2nd International Conference on Learning Representations, Banff, Canada, 2014.

[15] GOODFELLOW I J, SHLENS J, and SZEGEDY C. Explaining and harnessing adversarial examples[C]. The 3rd International Conference on Learning Representations, San Diego, USA, 2015.

[16] SHAFAHI A, NAJIBI M, GHIASI A, *et al*. Adversarial training for free![C]. The 32nd International Conference on Neural Information Processing Systems, Vancouver, Canada, 2019.

[17] MADRY A, MAKELOV A, SCHMIDT L, *et al*. Towards deep learning models resistant to adversarial attacks[C]. The 6th International Conference on Learning Representations, Vancouver, Canada, 2018.

[18] LI Yanjie, CUI Xiaoxin, ZHOU Yihao, *et al*. A comparative study on the performance and security evaluation of spiking neural networks[J]. *IEEE Access*, 2022, 10: 117572–117581. doi: 10.1109/ACCESS.2022.3220367.

[19] KUANG Yisong, CUI Xiaoxin, ZHONG Yi, *et al*. A 64K-neuron 64M-1b-synapse 2.64 pJ/SOP neuromorphic chip with all memory on chip for spike-based models in 65nm CMOS[J]. *IEEE Transactions on Circuits and Systems II:Express Briefs*, 2021, 68(7): 2655–2659. doi: 10.1109/TCSII.2021.3052172.

李　莹：女，博士，副研究员，研究方向为集成电路设计与验证、硬件安全.

李艳杰：女，硕士生，研究方向为神经网络算法与安全.

崔小欣：女，博士，研究员，研究方向为类脑芯片、信息处理、硬件安全.

倪庆龙：男，硕士生，研究方向为神经网络算法与电路设计.

周崟灏：男，硕士，高级工程师，研究方向为集成电路设计、硬件安全.

责任编辑：余　蓉