CasKDNet:基于改进 DenseNet 的恶意代码分类方法

刘 强[1,2],王 坚[1],路艳丽[1],王艺菲[1]
(1.空军工程大学防空反导学院,西安,710051;2.空军工程大学研究生院,西安,710051)

摘要 针对现有恶意代码可视化分类模型在精度和鲁棒性方面的不足,提出一种基于改进 DenseNet 的恶意代码可视化分类方法 CasKDNet,通过 3 项关键技术实现精度和鲁棒性的提升。首先,构建级联分类器结构,增强纹理相似家族的特征区分能力;其次,采用 KAN 结构替代 DenseNet 网络中的多层感知机,优化特征提取过程的非线性表达能力,提升模型整体精度;最后,基于 FFM 图像修复算法对训练集进行数据增强,提高模型鲁棒性。在恶意代码数据集 Malimg 上的实验结果显示,CasKDNet 模型取得 99.69%的分类准确率,与现有研究方法相比具有明显性能优势。此外,在白盒攻击背景下,FGSM 和 I-FGSM 算法对 CasKDNet 的攻击成功率仅为 12.7%和 37.5%,进一步证实了模型在防范对抗性攻击方面的有效性。

关键词 恶意代码;级联分类器;KAN;FFM 算法;对抗性攻击
DOI 10.3969/j.issn.2097-1915.2025.04.013
中图分类号 TP309 **文献标志码** A **文章编号** 2097-1915(2025)04-0110-10

CasKDNet: A Malware Classification Method Based on Improved DenseNet

LIU Qiang[1,2], WANG Jian[1], LU Yanli[1], WANG Yifei[1]
(1. Air Defense and Antimissile School, Air Force Engineering University, Xi'an 710051, China;
2. Graduate School, Air Force Engineering University, Xi'an 710051, China)

Abstract In existing malware visualization classification models, there are inadequate accuracy and robustness. For this reason, this paper proposes a malicious code visualization classification method CasKDNet (Cascade DenseNet with KAN) based on an improved DenseNet. The CasKDNet is to realize the improvements in accuracy and robustness by three key technologies. Firstly, a cascaded classifier structure is constructed to enhance the feature discrimination ability of texture similar families. Secondly, the KAN structure is used to replace the multi-layer perceptron in the DenseNet network, optimizing the non-linear expression ability of the feature extraction process and improving the overall accuracy of the model. Finally, the FFM image restoration algorithm is used to enhance the training set and improve the robustness of the model. It appears from the experimental results on the malicious code dataset Malimg that the CasKDNet model achieves 99.69% of classification accuracy, and is superior to the existing research methods. Furthermore, in the context of white box attacks, the success rate of FGSM and I-FGSM algorithms attack against the CasKDNet only serves as 12.7% and 37.5% respectively, and the model is valid in pre-

收稿日期:2025-01-08
基金项目:国家自然科学基金(61806219,61703426,61876189);陕西省高校科协青年人才托举计划(20190108,20220106);陕西省创新能力支撑计划(2020KJXX-065)
作者简介:刘 强(1993—),男,陕西白水人,硕士生,研究方向为网络空间安全和恶意代码检测。E-mail:dugugongsui@163.com
通信作者:路艳丽(1979—),女,陕西大荔人,副教授,研究方向为智能信息处理和网络空间安全。E-mail:luyanlihgh@163.com

引用格式:*刘强,王坚,路艳丽,等. CasKDNet:基于改进 DenseNet 的恶意代码分类方法[J]. 空军工程大学学报, 2025, 26(4):110-119. LIU Qiang, WANG Jian, LU Yanli, et al. CasKDNet: A Malware Classification Method Based on Improved DenseNet[J]. Journal of Air Force Engineering University, 2025, 26(4):110-119.*

venting adversarial attacks.

Key words　malware；cascade classifier；KAN；FFM algorithm；adversarial attack

随着信息技术的快速发展，网络安全面临的挑战日益严峻，其中恶意代码问题尤为突出，由于其变种繁多、传播迅速且危害巨大，给网络安全带来了严重的威胁，因此如何高效准确地检测恶意代码，是目前网络安全领域的研究热点[1]。恶意代码分析技术按照是否执行文件分为动态分析技术和静态分析技术[2]，由于动态分析需要占用大量计算和存储资源，分析的难度和成本较高，因此基于静态分析的恶意代码检测方法仍是主流。传统的静态分析技术基于签名和特征库，通过构建已知恶意代码特征库，将待检测的样本特征码与特征库进行相似性对比判断是否恶意，然而随着反检测技术的不断发展，恶意代码采用各种加壳、混淆等对抗技术进化出更具威胁性的变种[3]，传统的静态分析方法已暴露出其局限性，无法满足当前对恶意代码检测的需求。在这种背景下，恶意代码可视化检测技术应运而生，由于同一类别恶意代码的可视化图像通常具有相似性，而不同家族的可视化图像之间则有较大的差异[4]，这种方法能够直观地展示恶意代码的特征，提高检测的准确性和效率。

卢喜东等[5]基于灰度图编码与方向梯度直方图特征，采用深度森林算法实现分类；王伟等[6]针对图像中小目标物体所占像素少容易出现漏检的问题，提出了一种结合注意力机制和特征融合的小目标检测方法。轩勃娜等[7]构建包含二进制码、汇编指令和 API 调用特征的三通道 RGB 图，实现比灰度图更好的分类效果；孙松等[8]提出 RGBA 四通道编码方法，得到的不同家族恶意代码图像具有明显结构差异；黄保华等[9]引入部分卷积与可分离注意力机制，有效提升 ViT 模型的局部特征提取能力，并降低模型参数量。Deng 等[10]构建汇编指令马尔可夫转移矩阵，在微软公共恶意软件数据集上准确率达 99.44%；Vasan 等[11]通过截断奇异值分解（singular value decomposition，SVD）降低特征向量维度，缓解模型过拟合问题；Duraibi 等[12]将 Snake 优化算法与 ShuffleNet 模型深度融合，显著提升图像特征表征能力；Bavishi 等[13]构建级联检测系统，先判别恶意属性再进行家族分类。Liu 等[14]集成数据可视化、平衡采样与增强技术，在保证检测精度的同时有效提升模型泛化能力；Salas 等[15]采用双三次插值等技术建立微调的 MobileNet 模型，在多个数据集上取得较高准确率。

尽管恶意代码可视化检测方法取得了较好的效果，但仍面临着以下不可忽视的问题：一是将恶意代码可视化为图像后，不同恶意代码家族之间也存在纹理相似的情况，模型难以准确分类。比如 Malimg 数据集中 Autorun.K 和 Yuner.A 两个家族的图像纹理高度相似，文献[14]中的方法将 Autorun.K 中的样本全部分类为 Yuner.A。二是现有基于深度学习的恶意代码分类模型鲁棒性不佳，容易受到对抗样本的攻击，攻击者只需修改恶意代码图像的少量字节，便可导致模型误分类。比如文献[16]通过在 PE 文件的资源段上引入扰动，来攻击基于图像的可视化恶意软件分类系统，攻击成功率高达 87%。

针对上述问题，本文提出了一种基于改进 Densenet 的恶意代码分类方法 CasKDNet（Cascade DenseNet with KAN），主要创新点如下：

1）针对个别恶意代码家族之间图像纹理相似，导致模型难以分类的问题，设计了一个级联分类器，提高了模型对纹理相似家族的分类能力。

2）采用 Kolmogorov-Arnold 网络代替 Densenet 网络中的多层感知机，提高了模型精度。

3）基于 FFM 图像修复算法对训练集进行数据增强，提高了模型鲁棒性，可以有效防范白盒背景下的对抗性攻击。

1　基础理论和技术

1.1　Densenet 模型

DenseNet[17]是一种卷积神经网络架构，设计上具有密集连接的特点。其核心思想是每一层的输入不仅来自上一层，还来自前面所有的层，这种密集连接方式使得网络中每一层都能充分利用前面层的特征，改善了梯度流，缓解了梯度消失问题，且使用了较少的参数以提高计算效率。

DenseNet 模型通常包括多个 DenseBlock，DenseBlock 是 DenseNet 的核心组件，在 1 个结构为 L 层的 DenseBlock 中，第 l 层接收前面所有层的特征图作为输入：

$$x_l = H_l([x_0, x_1, \cdots, x_{l-1}]) \tag{1}$$

式中：x_0，x_1，…，x_{l-1} 分别为第 0～$l-1$ 层对应的特征图；$H_l(\cdot)$为包批量归一化、激活函数和卷积 3 种操作的复合函数。

1.2　Kolmogorov-Arnold 网络

Kolmogorov-Arnold Networks（KAN）[18]是一

种新型神经网络架构，与传统使用固定激活函数的神经网络不同，KAN 在网络的边缘采用可学习的激活函数。在 KAN 中，传统的权重参数在网络的边缘被单变量函数参数所取代，每个节点汇总这些函数的输出时不进行任何非线性变换，这与多层感知机(multilayer perceptron，MLP)中的做法形成了鲜明对比，KAN 的具体实现将在 2.2.2 节中详细介绍。

1.3 FFM 图像修复算法

FFM(fast marching method)图像修复算法[19]是一种基于偏微分方程(PDE)的高效图像修复技术，主要用于填补图像中的缺失区域，使其与周围背景自然融合。其核心思想是通过模拟“波前传播”的过程，逐步从已知区域向未知区域扩散信息，优先修复结构简单的区域，再处理复杂区域。对每个待修复像素，根据其周围有效像素的梯度、颜色等信息进行插值，确保修复后的区域在结构和纹理上与周围一致，并使用加权平均或偏微分方程避免模糊边缘。FFM 图像修复算法的具体实现将在 2.3 节中介绍。

2 CasKDNet 模型概述

CasKDNet 模型整体框架如图 1 所示，由 2 个改进的 Densenet 模型构成 1 个级联分类器，以实现更好的恶意代码分类精度，并基于 FFM 图像修复算法进行数据增强，以提高模型鲁棒性。

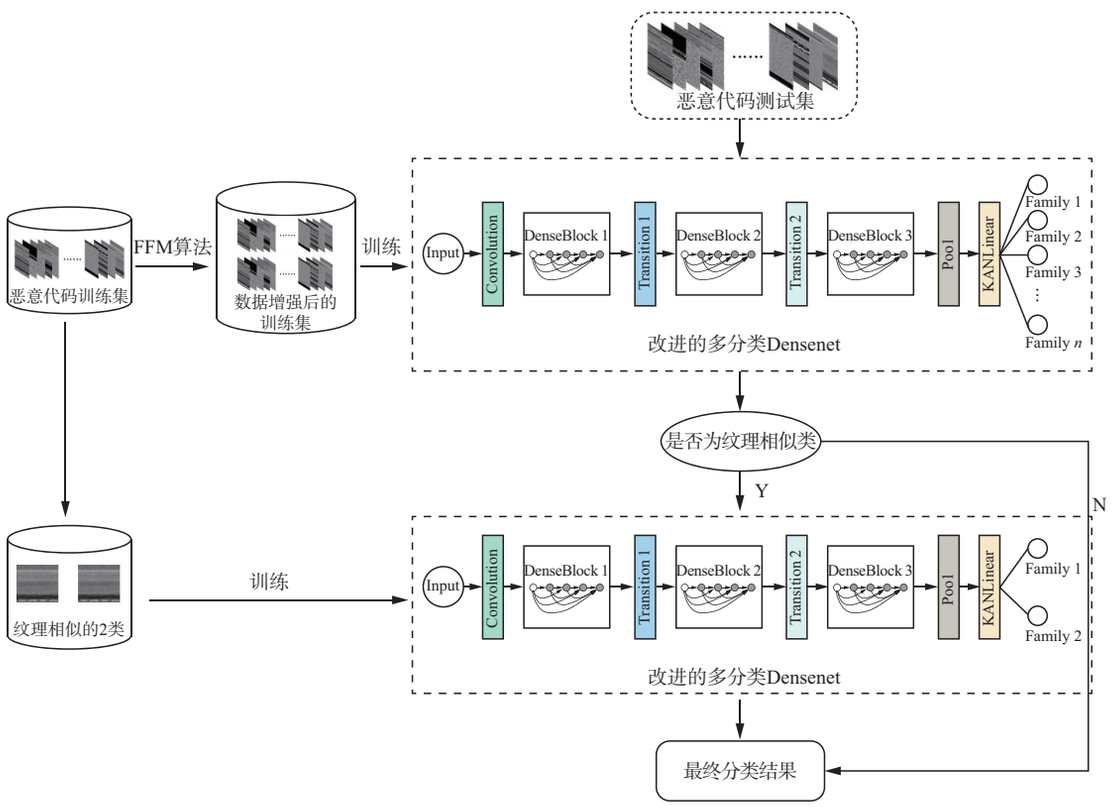

图 1 CasKDNet 模型整体框架

Fig. 1 Overall framework of CasKDNet

2.1 级联分类器运行机制

采用级联分类器实现恶意代码分类，具体实施步骤如下：

步骤 1 采用 FFM 图像修复算法对恶意代码训练集图像进行数据增强，随后用增强后的训练集训练一个改进的多分类 Densenet 模型。

步骤 2 对训练集中纹理相似的 2 类恶意代码(Autorun.K 和 Yuner.A)，训练一个改进的二分类 Densenet 模型。

步骤 3 将恶意代码测试集图像输入到多分类 Densenet 模型，得到初步分类结果。

步骤 4 若初步分类结果不属于纹理相似的 2 类，则直接输出为最终结果；若属于纹理相似的 2 类，则将原始图像再次输入到二分类 Densenet 模型进行分类，得到最终结果。

2.2 改进的 Densenet 模型

2.2.1 模型网络结构

改进的 Densenet 模型参考了 DenseNet121 网络结构的设计，和 DenseNet121 不同之处在于，DenseNet121 将图像输入尺寸统一为 224×224，包含 4 个 DenseBlock 和 3 个 Transition 层，本文结合恶意代码分类任务的实际，参考文献[20]，将图像输入尺寸固定为 64×64，有利于在提高恶意代码分类性能的同时减少参数量，防止模型过拟合，由于输入

图像尺寸的改变，仅需使用 3 个 DenseBlock 和 2 个 Transition 层，此外，在全连接层采用 KAN 结构代替 MLP 结构，以提高模型的拟合能力。改进的 Densenet 模型网络结构和参数如表 1 所示。

表 1　改进的 Densenet 模型网络结构和参数

Tab. 1　Network structure and parameters of improved densenet

层类型	配置参数	输出尺寸（$H\times W\times C$）
输入层	输入：64×64×3	64×64×3
卷积层	卷积核大小：7×7 步长：2 通道数：64	32×32×64
最大池化层	池化窗口：3×3 步长：2	16×16×64
DenseBlock 1	卷积层数：6 通道增长率：32	16×16×256
Transition 1	1×1 卷积 步长：1 池化窗口：2×2	8×8×128
DenseBlock 2	卷积层数：12 通道增长率：32	8×8×512
Transition 2	1×1 卷积 步长：1 池化窗口：2×2	4×4×256
DenseBlock 3	卷积层数：24 通道增长率：32	4×4×1024
全局平均池化		1×1×1024
KANLinear	类别数：n	1×1×n

2.2.2　KANLinear 模块

传统的全连接层采用 MLP 结构，可表示为：

$$\mathrm{MLP}(x)=(W_{l-1}\circ\sigma\circ W_{l-2}\circ\sigma\circ\cdots\circ W_1\circ\sigma\circ W_0\circ\sigma)x \tag{2}$$

式中：W_0，W_1，…，W_{l-2}，W_{l-1} 分别为第 0～$l-1$ 层的线性权重参数；σ 为非线性激活函数；$\circ$ 表示函数组合。

KANLinear 模块采用 KAN 结构，可表示为：

$$\mathrm{KAN}(x)=(\Phi_{l-1}\circ\Phi_{l-2}\circ\cdots\circ\Phi_1\circ\Phi_0)x \tag{3}$$

式中：$\boldsymbol{\Phi}_l$ 为激活函数矩阵。

$$\boldsymbol{\Phi}_l=\begin{pmatrix}\varphi_{l,1,1}(\cdot) & \varphi_{l,1,2}(\cdot) & \cdots & \varphi_{l,1,n_l}(\cdot)\\ \varphi_{l,2,1}(\cdot) & \varphi_{l,2,2}(\cdot) & \cdots & \varphi_{l,2,n_l}(\cdot)\\ \vdots & \vdots & & \vdots\\ \varphi_{l,n_{l+1},1}(\cdot) & \varphi_{l,n_{l+1},2}(\cdot) & \cdots & \varphi_{l,n_{l+1},n_l}(\cdot)\end{pmatrix} \tag{4}$$

激活函数 $\varphi_{l,j,i}(\cdot)$ 连接第 l 层的第 i 个神经元与第 $l+1$ 层的第 j 个神经元，为基函数 $b(x)$ 和样条函数 spline(x) 的加权组合：

$$\varphi_x=w_b b(x)+w_s\,\mathrm{spline}(x) \tag{5}$$

$$b(x)=\mathrm{silu}(x)=x/(1+\mathrm{e}^{-x}) \tag{6}$$

$$\mathrm{spline}(x)=\sum_i c_i B_i(x) \tag{7}$$

式中：c_i 为训练期间优化的系数；$B_i(x)$ 为定义在网络上的 B 样条基函数，样条的灵活性使其能够通过调整形状来适应性地建模数据中的复杂关系，从而最小化近似误差，增强了网络从高维数据中学习细微模式的能力。

2.3　基于 FFM 图像修复算法的数据增强方法

在 FFM 图像修复算法中，对于图像 I，设 Ω 为待修复区域，p 为 Ω 边界上任意一点，则点 p 的像素值估计为：

$$I(p)=\frac{\sum_{q\in B_\varepsilon(p)} w(p,q)[I(q)+\nabla I(q)(p-q)]}{\sum_{q\in B_\varepsilon(p)} w(p,q)} \tag{8}$$

式中：$B_\varepsilon(p)$ 为以点 p 为中心、ε 为半径的已知图像内的 1 个邻域；q 为该邻域中任意一点；$\nabla I(q)$ 为 q 点的梯度。

为了使填充更加精确，增加已知像素点 q 对待填充区域的影响，添加 1 个权重函数 $w(p,q)$：

$$w(p,q)=\mathrm{dir}(p,q)\cdot\mathrm{dst}(p,q)\cdot\mathrm{lev}(p,q) \tag{9}$$

式中：dst(p，q) 为几何距离因子；lev(p，q) 为水平集距离因子；dir(p，q) 为方向因子。这些因子共同决定了已知像素点对待修复像素点的影响程度。对边界填充完后，需要不断迭代上述步骤，逐渐收缩边界直至空洞区域修复完毕。

采用 FFM 图像修复算法对恶意代码图像进行数据增强的过程如图 2 所示，具体步骤如表 2 所示。将原始图像划分为 $n\times n$ 个尺寸相同的正方形块，依次向每个块内随机位置生成大小为 $k\times k$ 的掩码区域，得到待修复的图像，遍历每个掩码区域，采用 FFM 算法对像素值进行恢复，便可得到新的图像。为了达到防范对抗性攻击的目的，同时避免数据增强后训练集样本数量过大而影响模型训练速度，合理确定新生成图像的数量十分关键。本文在 Malimg 数据集上进行了大量实验，得出在 $n=16$、$k=2$ 的情况下，为每个原始图像生成 5 个新的图像时模型能够较好地防范对抗性攻击，且每一轮的训练时间为 72.4 s，处于可接受范围。

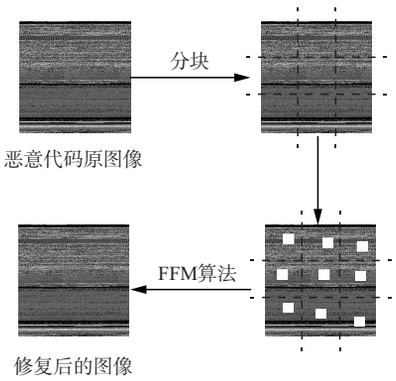

图 2 基于 FFM 图像修复算法的数据增强方法过程

Fig. 2 Data augmentation method based on FFM image restoration algorithm process

表 2 基于 FFM 图像修复算法的数据增强方法步骤

Tab. 2 Data augmentation method based on FFm image restoration algorithm process

算法 1 基于 FFM 图像修复算法的数据增强方法

输入：恶意代码原图像 I

输出：生成的新图像 I'

1. 初始化参数：图像划分的块数 $n\times n$，掩码区域的尺寸 $k\times k$，FFM 算法的邻域半径参数 ε

2. $w\leftarrow$getwidth(I)

3. $h\leftarrow$getheight(I)

4. mask[w, h]$\leftarrow$0

5. for $i\leftarrow$1 to n

6. for $j\leftarrow$1 to n

7. $x\leftarrow$rand($j\times w/n$, $(j+1)\times w/n-k$)

8. $y\leftarrow$rand($i\times h/n$, $(i+1)\times h/n-k$)

9. mask[y:$y+k$, x:$x+k$]$\leftarrow$255

10. $I'\leftarrow$FFM(I, mask, ε)

本文之所以考虑基于 FFM 的图像修复算法进行数据增强，以防范对抗性攻击，主要基于 2 个假设：一是为原始图像设置多个较小的掩码区域，并采用图像修复算法进行修复，则掩码区域像素值会被改变，新生成的图像本质上是一个对抗样本，可以达到对抗性训练的效果；二是采用图像修复算法修复后的图像，和原图像的纹理仍然高度相似，从而不会影响模型的分类精度。第 3.3 节的实验结果验证了以上假设的合理性。

3 实验与分析

3.1 数据集与实验环境

本文选取恶意代码公开数据集 Malimg 进行实验，该数据集包括 25 个不同家族的恶意样本共 9 339 个，已按 8∶1∶1的比例划分为训练集、验证集和测试集，具体样本种类和数量分布如图 3 所示，实验环境如表 3 所示。

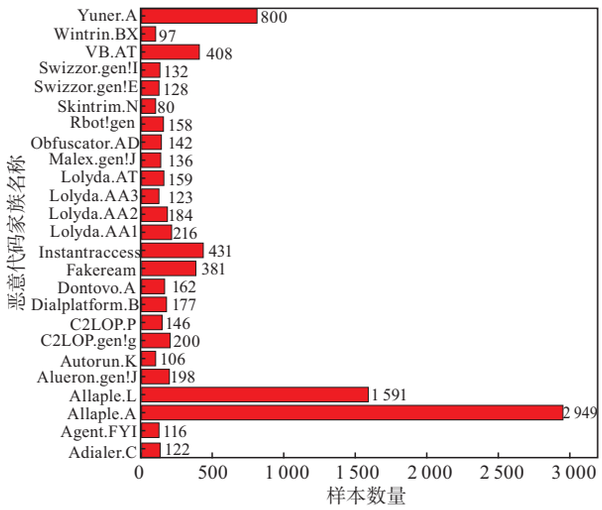

图 3 Malimg 数据集样本分布

Fig. 3 Sample distribution of Malimg

表 3 实验环境配置

Tab. 3 Configuration of experimental environment

实验环境	具体配置
操作系统	Windows 11
CPU	Intel(R) Core(TM) i7-13620H CPU @ 2.40 GHz 2.40 GHz
内存	16 GB
硬盘	1 TB
显卡	NVIDIA GeForce RTX 4050
开发框架	Pytorch
开发语言	Python 3.10

3.2 评价指标

实验评价选用了准确率 Accuracy、精确率 Precision、召回率 Recall 和 F1-score 4 个指标。

$$\text{Accuracy}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \tag{10}$$

$$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}} \tag{11}$$

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}} \tag{12}$$

$$\text{F1-score}=2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{13}$$

式中：TP 和 FP 分别为对正样本的正确预测和错误预测；TN 和 FN 分别为对负样本的正确预测和错误预测。

3.3 实验结果与分析

3.3.1 CasKDNet 模型性能分析

超参数的选择直接影响着模型训练效果，为了使模型性能得到充分体现，经过严格的参数调优，最

终确定模型各项超参数的值如表 4 所示。

表 4　CasKDNet 模型超参数设置

Tab. 4　Hyperparameter setting of CasKDNet

超参数	具体设置
学习率	0.002
权重衰减参数	0.001
批量大小	64
epoch	100
损失函数	交叉熵损失函数
优化器	AdamW
图像归一化尺寸	(64, 64)

图 4 显示了 CasKDNet 模型在训练集和测试上准确率随着训练轮次的变化情况，图 5 显示了损失值随着训练轮次的变化情况。由图 4 可以观察出，在前 10 个训练轮次里，模型已基本达到收敛状态，说明具有较快的收敛速度。随着训练轮次的增加，模型在训练集上准确率最高可达 100%，在测试集上准确率最高可达 99.69%。由图 5 可以观察出，训练集和测试集上损失值最小都接近于 0。

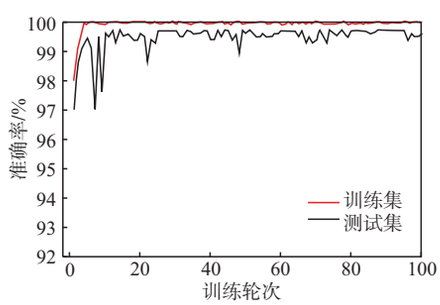

图 4　准确率随训练轮次变化曲线

Fig. 4　Accuracy variation with training epochs

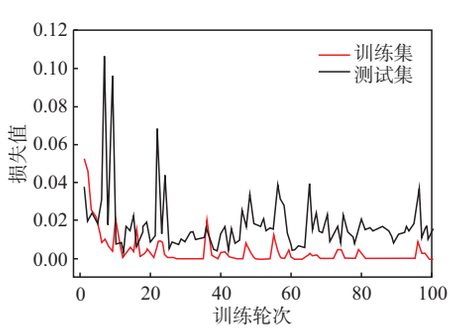

图 5　损失值随训练轮次变化曲线

Fig. 5　Loss variation with training epochs

为了更好地观察模型对每一类恶意代码的分类细节，绘制了混淆矩阵，如图 6 所示。可以看出，模型对大多数恶意代码家族的分类准确率高达 100%，仅有 Swizzor.gen!E 家族分类准确率低于 90%，说明模型较好地学习到了每个家族恶意代码图像的纹理特征，能够做出科学的分类预测。

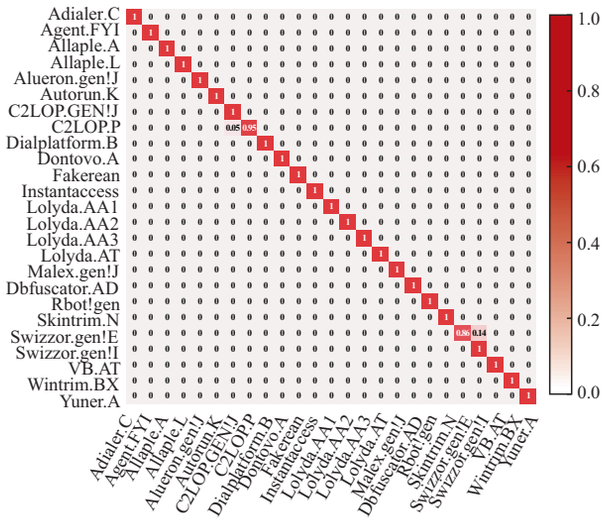

图 6　CasKDNet 模型对应混淆矩阵

Fig. 6　Confusion matrix of CasKDNet

3.3.2　消融实验

为了探究不同模块对 CasKDNet 模型性能的影响，设置了 3 种 CasKDNet 的变体进行消融实验。CasKDNet-1 表示不采用级联分类器，CasKDNet-2 表示不采用 KAN 结构，CasKDNet-3 表示不采用图像增强，各模型在测试集上的混淆矩阵如图 7～图 9 所示，各项指标性能如表 5 所示。

结合图 7～图 9 和表 5 的数据可以看出，不采用级联分类器时，模型的准确率、精确率、召回率和 F1-score 4 个指标平均下降了 1.27%，主要原因是模型将 Autorun.K 家族全部误分类为 Yuner.A。不采用 KAN 时，模型 4 个指标平均下降了 0.22%。不进行图像增强的情况下，模型性能未受到明显影响。

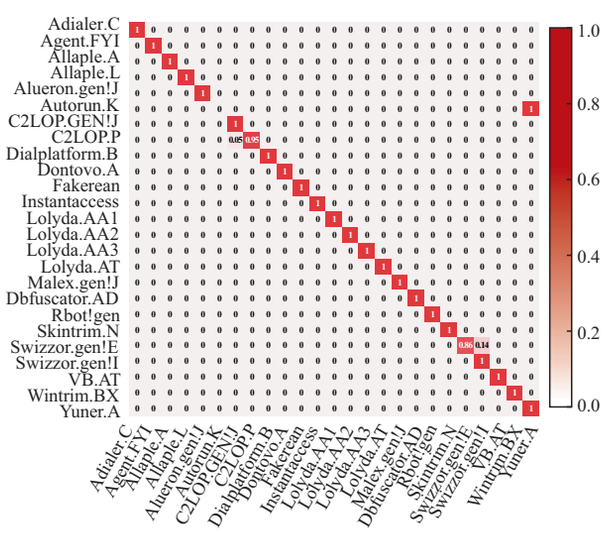

图 7　CasKDNet-1 模型对应混淆矩阵

Fig. 7　Confution matrix of CasKDNet-1

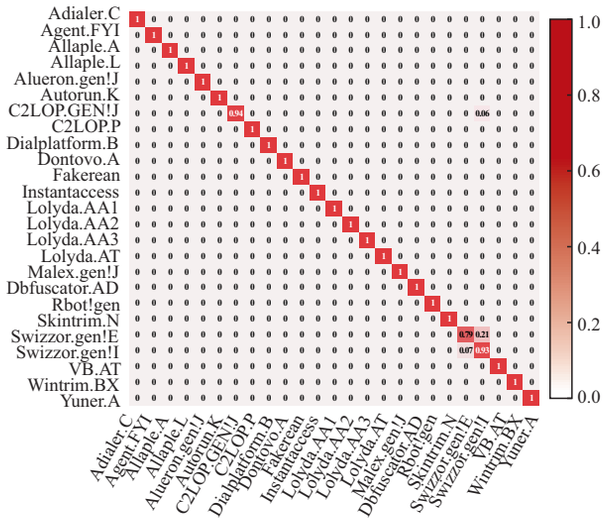

图 8 CasKDNet-2 模型对应混淆矩阵

Fig. 8 Confution matrix of CasKDNet-2

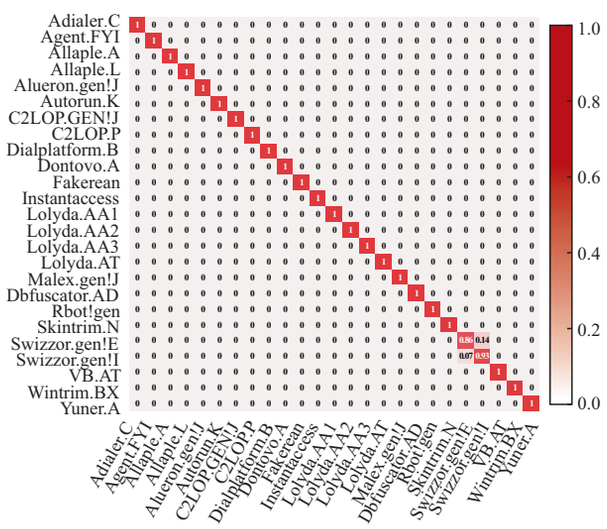

图 9 CasKDNet-3 模型对应混淆矩阵

Fig. 9 Confution matrix of CasKDNet-3

表 5 消融实验结果

Tab. 5 Results of ablation experiment

模型	评价指标/%			
	Accuracy	Precision	Recall	F1-score
CasKDNet-1	98.43	98.44	98.43	98.43
CasKDNet-2	99.48	99.48	99.48	99.47
CasKDNet-3	99.69	99.70	99.69	99.69
CasKDNet	99.69	99.74	99.69	99.68

3.3.3 实验结果对比

为了验证本文所提方法的有效性，将 CasKDNet 模型和近 5 年来在 Malimg 数据集上提出其他的恶意代码分类模型进行对比，结果如表 6 所示。从表中数据可知，和现有研究方法相比，CasKDNet 模型具有更好的效果，各项指标均具有优势。

表 6 CasKDNet 模型与其他研究方法实验结果对比

Tab. 6 Comparison of experimental results between CasKDNet and other methods

方法	年份	评价指标/%			
		Accuracy	Precision	Recall	F1-score
文献[21]	2019	99.03			
文献[22]	2020	98.82	98.85	98.81	98.75
文献[23]	2020	98.79	98.79	98.79	
文献[24]	2021	98.63			96.58
文献[25]	2021	98.90	98.58	98.79	98.89
文献[6]	2021	99.17	99.10	98.80	98.95
文献[26]	2022	98.50	94.23	93.91	93.91
文献[27]	2022	99.03			
文献[20]	2023	99.35	99.37	99.35	99.35
文献[28]	2023	98.81	98.85	98.79	98.83
文献[12]	2024	98.42			
文献[15]	2024	99.08	99.06	99.07	99.06
文献[29]	2024	99.31	99.26	99.23	99.24
本文	2025	99.69	99.74	99.69	99.68

3.3.4 对抗性攻击实验

为验证 CasKDNet 模型鲁棒性，在 Malimg 数据集的测试集中选择所有被模型正确分类的恶意代码图像，在白盒攻击背景下分别采用 FGSM 和 I-FGSM 2 种算法，为原始图像生成对抗样本，对模型进行攻击，观察攻击成功率。

在对抗性攻击中，扰动参数 ε 的设置是一个关键因素，决定了对抗样本与原始样本之间的差异程度。为了在实验中将 ε 的取值限定在一个合理范围，以 Adposhel 家族样本为例，采用 FGSM 和 I-FGSM 2 种算法在不同扰动参数下生成对抗样本图像如图 10 和图 11 所示。

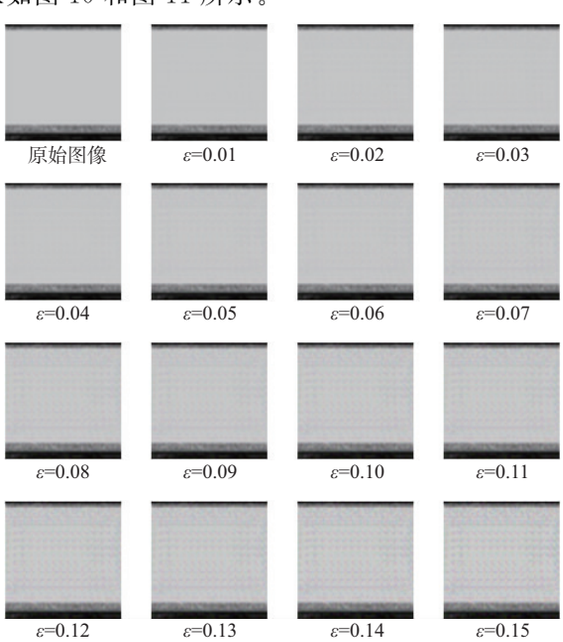

图 10 不同 ε 值下 FGSM 算法生成的对抗样本图像

Fig. 10 Adversarial samples generated by FGSM algorithm under different ε values

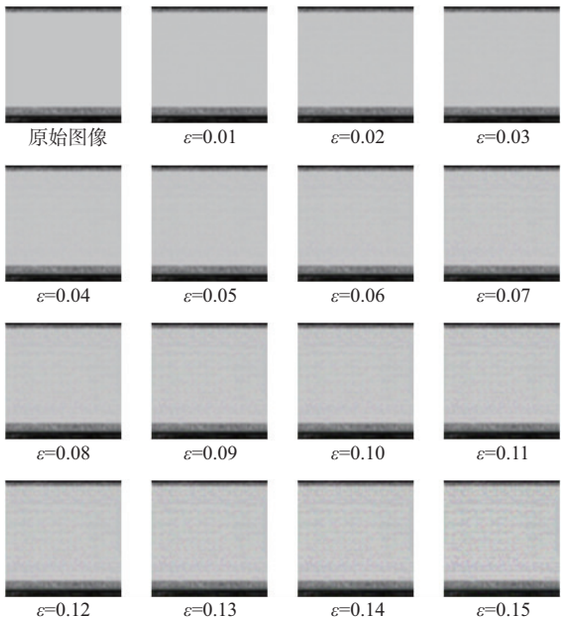

图 11　不同 ε 值下 I-FGSM 算法生成的对抗样本图像

Fig. 11　Adversarial samples generated by I-FGSM algorithm under different ε values

可以观察出，当 ε≤0.12 时，生成的对抗样本图像噪点较少，不易被肉眼察觉，当 ε＞0.12 时，噪点明显，容易被察觉。因此实验中将 ε 的取值范围设置为[0，0.12]，图 12 为不同扰动参数下 2 种算法的攻击成功率。

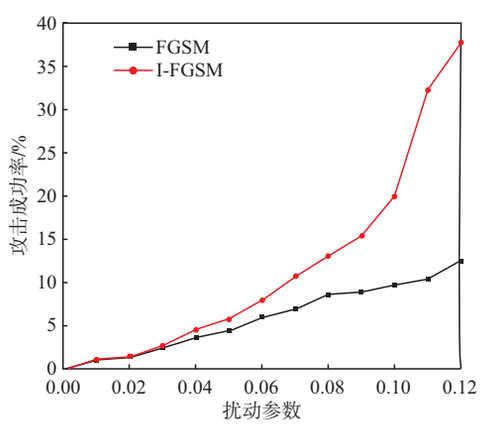

图 12　不同 ε 值下 2 种算法的攻击成功率

Fig. 12　Attack success rates of two algorithms under different ε values

可以观察出，随着 ε 的增大，2 种算法的攻击成功率都呈不断上升趋势。FGSM 算法的攻击成功率最高为 12.7%，说明模型分类准确率为 87.3%，I-FGSM 算法的攻击成功率最高为 37.5%，说明模型分类准确率为 62.5%，可见 CasKDNet 模型具有较好的鲁棒性，能够有效防御对抗性攻击。

为了将 CasKDNet 模型鲁棒性和其他模型进行对比，以文献[15]的开源代码为依据，对其采用的 MobileNet FT 模型进行复现，并采用 FGSM 算法和 I-FGSM 算法对训练好的 MobileNet FT 模型进行攻击，结果如图 13 所示。

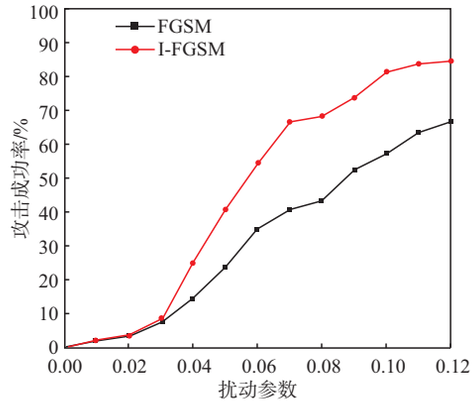

图 13　不同 ε 值下 2 种算法对 MobileNet FT 的攻击成功率

Fig. 13　Attack success rates of two algorithms on MobileNet FT under different ε values

可以观察出，FGSM 算法的攻击成功率最高为 69.8%，I-FGSM 算法的攻击成功率最高为 85.1%，说明 MobileNet FT 模型鲁棒性较差，不能有效防御对抗性攻击。究其原因，在于 MobileNet FT 模型侧重于提高模型精度和泛化能力，而未采取防范对抗性攻击的措施。

此外，为了探究基于 FFM 图像修复算法的数据增强方法对模型鲁棒性的贡献，设置了对比实验，采用数据增强前的训练集进行训练，对训练完毕后的模型进行攻击，结果如图 14 所示。

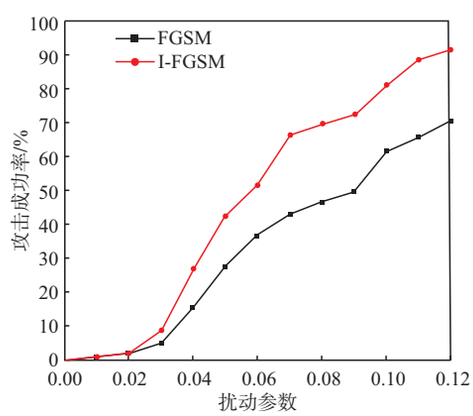

图 14　不同 ε 值下 2 种算法的攻击成功率(数据增强前)

Fig. 14　Attack success rates of two algorithms before data augmentation under different ε values

可以观察出，当不采用数据增强时，2 种算法的攻击成功率都明显提升，FGSM 算法的攻击成功率最高为 70.3%，I-FGSM 算法的攻击成功率最高为 91.4%，这意味着训练好的模型基本失效，无法防御对抗性攻击，可见数据增强是模型鲁棒性提高的直接原因。

需要说明的是，CasKDNet 模型采用级联分类器、KAN 和数据增强 3 种技术，在实现高精度和鲁棒性的同时，不可避免地带来了更多计算和存储资

源的开销。具体而言：

1)采用级联分类器在测试集上进行测试时，每个样本的预测时间为 32.5 ms，比采用单个 DenseNet 模型时增加 11.2 ms。

2)KAN 的计算过程比 MLP 更复杂，这使模型训练时间增加了 16%。

3)数据增强阶段，需耗时 201.4 s 生成额外 283.5 MB 的增强图像，且数据增强后模型每训练一轮耗时 72.4 s，比数据增强前增加 53.6 s，因此完成 100 轮次训练需要额外消耗 5 360 s。

综上所述，在本文采用的 Malimg 数据集的规模下，以上计算和存储资源的开销均处于可控范围，但在实际应用中，若恶意代码数据集规模过大，模型的时间复杂度和空间复杂度将显著增长，CasKDNet 的使用可能会受到计算和存储资源的限制。

4 结语

本文针对恶意代码检测领域面临的挑战，提出了 3 种创新方法以提升模型的性能。首先，通过设计级联分类器解决了个别恶意代码家族之间图像纹理相似导致的分类难题，显著提高了对纹理相似家族的识别能力。其次，结合恶意代码分类任务的需要，对 Densenet 网络结构进行简化，减少了模型参数量，以防止模型过拟合，并引入 KAN 结构替代 Densenet 中的多层感知机，进一步优化了网络架构，提升了模型的整体精度。最后，采用基于 FFM 图像修复算法的数据增强策略，提高了模型的鲁棒性，有效抵御了白盒攻击背景下常见的 FGSM 和 I-FGSM 2 种对抗性攻击手段。然而，本文方法也存在一定局限性：一是 CasKDNet 模型在实现高精度和鲁棒性的同时，产生了更多计算和存储资源的开销，在实际应用中若数据集规模过大，该方法的使用将受到影响；二是随着对抗性攻击算法的不断发展，涌现出各种更为先进的攻击手段，而本文仅在 2 种攻击算法下验证 CasKDNet 模型的鲁棒性，模型能否抵御新的攻击手段仍需验证。未来的工作将进一步优化基于 FFM 图像修复算法的数据增强方法，以减少时间和存储资源的开销，并引入多种近年来提出的对抗性攻击算法，来测试 CasKDNet 模型鲁棒性，从而扩展模型防御能力。

参考文献

[1] 宋亚飞，张丹丹，王坚，等. 基于深度学习的恶意代码检测综述[J]. 空军工程大学学报，2024，25(4)：94-106.
SONG Y F，ZHANG D D，WANG J，et al. Review of Malware Detection Based on Deep Learning[J]. Journal of Air Force Engineering University，2024，25(4)：94-106.(in Chinese)

[2] 李思聪，王坚，宋亚飞，等. 基于扩散模型的恶意代码数据集扩充方法[J]. 空军工程大学学报，2025，26(1)：95-103.
LI S C，WANG J，SONG Y F，et al. A Diffusion Model Approach to Malicious Code Dataset Expansion[J]. Journal of Air Force Engineering University，2025，26(1)：95-103.(in Chinese)

[3] 黄玮，王坚，吴暄，等. 基于 BiTCN-SA 的恶意代码分类方法[J]. 空军工程大学学报，2023，24(4)：77-84.
HUANG W，WANG J，WU X，et al. A Malicious Code Classification Method Based on BiTCN-SA[J]. Journal of Air Force Engineering University，2023，24(4)：77-84.(in Chinese)

[4] 王金伟，陈正嘉，谢雪，等. 恶意软件检测和分类可视化技术综述[J]. 网络与信息安全学报，2023，9(5)：1-20.
WANG J W，CHEN Z J，XIE X，et al. Review of Malware Detection and Classification Visualization Techniques[J]. Chinese Journal of Network and Information Security，2023，9(5)：1-20.(in Chinese)

[5] 卢喜东，段哲民，钱叶魁，等. 一种基于深度森林的恶意代码分类方法[J]. 软件学报，2020，31(5)：1454-1464.
LU X D，DUAN Z M，QIAN Y K，et al. Malicious Code Classification Method Based on Deep Forest[J]. Journal of Software，2020，31(5)：1454-1464.(in Chinese)

[6] 王伟，万晓刚. 结合注意力机制和特征融合的小目标检测方法[J]. 西安工程大学学报，2022，36(6)：115-123.
WANG W，WAN X G. A Small Target Detection Method Combining Attention Mechanism and Feature Fusion[J]. Journal of Xi'an Polytechnic University，2022，36(6)：115-123.(in Chinese)

[7] 轩勃娜，李进. 基于改进 CNN 的恶意软件分类方法[J]. 电子学报，2023，51(5)：1187-1197.
XUAN B N，LI J. Malware Classification Method Based on Improved CNN[J]. Acta Electronica Sinica，2023，51(5)：1187-1197.(in Chinese)

[8] 孙松，高丽婷，范祎宁，等. 恶意代码可视化 RGBA 图像方法研究[J]. 河北建筑工程学院学报，2024，42(3)：229-234，246.
SUN S，GAO L T，FAN Y N，et al. Research on RGBA Image Visualization Method for Malicious Code[J]. Journal of Hebei Institute of Architecture and Civil Engineering，2024，42(3)：229-234，246.(in Chinese)

[9] 黄保华，杨婵娟，熊宇，等. 基于 ViT 的轻量级恶意代码检测架构[J]. 信息网络安全，2024，24(9)：1409-1421.
HUANG B H，YANG C J，XIONG Y，et al. Light Weight Malicious Code Detection Architecture Based on Vision Transformer[J]. Netinfo Security，2024，24(9)：1409-1421. (in Chinese)

[10] DENG H X，GUO C，SHEN G W，et al. MCTVD：A Malware Classification Method Based on Three-Channel Visualization and Deep Learning[J]. Computers & Security，2023，126：103084.

[11] VASAN D，HAMMOUDEH M，ALAZAB M. Broad-Learning：A GPU-Free Image-Based Malware Classification[J]. Applied Soft Computing，2024，154：111401.

[12] DURAIBI S. Enhanced Image-Based Malware Classification Using Snake Optimization Algorithm with Deep Convolutional Neural Network[J]. IEEE Access，2024，12：95047-95057.

[13] BAVISHI S，MODI S. Accelerating Malware Classification：A Vision Transformer Solution[EB/OL]. (2024-09-28)[2025-01-12]. https://arxiv.org/abs/2409.19461.

[14] LIU Y J，FAN H，ZHAO J G，et al. Efficient and Generalized Image-Based CNN Algorithm for Multi-Class Malware Detection[J]. IEEE Access，2024，12：104317-104332.

[15] SALAS M P，DE GEUS P L. Deep Learning Applied to Imbalanced Malware Datasets Classification[J]. Journal of Internet Services and Applications，2024，15(1)：342-359.

[16] VI B N，NGUYEN H N，NGUYEN N T，et al. Adversarial Examples Against Image-Based Malware Classification Systems[C]//2019 11th International Conference on Knowledge and Systems Engineering (KSE). Da Nang，Vietnam：IEEE，2019：1-5.

[17] HUANG G，LIU Z，VAN DER MAATEN L，et al. Densely Connected Convolutional Networks[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu，HI：IEEE，2017：2261-2269.

[18] LIU Z，WANG Y，VAIDYA S，et al. Kan：Kolmogorov-Arnold Networks[EB/OL]. (2024-04-30)[2025-01-12]. http://arxiv.org/abs/2404.19756.

[19] TELEA A. An Image Inpainting Technique Based on the Fast Marching Method[J]. Journal of Graphics Tools，2004，9(1)：23-34.

[20] 张丹丹，宋亚飞，刘曙. MalMKNet：一种用于恶意代码分类的多尺度卷积神经网络[J]. 电子学报，2023，51(5)：1359-1369.
ZHANG D D，SONG Y F，LIU S. MalMKNet：A Multi-Scale Convolutional Neural Network Used for Malware Classification[J]. Acta Electronica Sinica，2023，51(5)：1359-1369. (in Chinese)

[21] LO W W，YANG X，WANG Y P. An Xception Convolutional Neural Network for Malware Classification with Transfer Learning[C]//2019 10th IFIP International Conference on New Technologies，Mobility and Security (NTMS). Canary Islands：IEEE，2019：1-5.

[22] VASAN D，ALAZAB M，WASSAN S，et al. IMCFN：Image-Based Malware Classification Using Fine-Tuned Convolutional Neural Network Architecture[J]. Computer Networks，2020，171：107138.

[23] NAEEM H，ULLAH F，NAEEM M R，et al. Malware Detection in Industrial Internet of Things Based on Hybrid Image Visualization and Deep Learning Model[J]. Ad Hoc Networks，2020，105：102154.

[24] ÇAYIR A，ÜNAL U，DAĞH S. Random CapsNet Forest Model for Imbalanced Malware Type Classification Task[J]. Computers & Security，2021，102：102133.

[25] ALAM M，AKRAM A，SAEED T，et al. DeepMalware：A Deep Learning Based Malware Images Classification[C]//2021 International Conference on Cyber Warfare and Security (ICCWS). Islamabad：IEEE，2021：93-99.

[26] PAARDEKOOPER C，NOMAN N，CHIONG R，et al. Designing Deep Convolutional Neural Networks Using a Genetic Algorithm for Image-Based Malware Classification[C]//2022 IEEE Congress on Evolutionary Computation (CEC). Padua：IEEE，2022：1-8.

[27] QIU L F，WANG S，WANG J，et al. Malware Classification Based on a Light-Weight Architecture of CNN：MalShuffleNet[C]//2022 3rd International Conference on Computer Vision，Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA). Changchun：IEEE，2022：1047-1050.

[28] WU Z P，YANG W Y，GUO L J，et al. MalEdgeNeXt：A Lightweight Malware Family Classification Method[C]//2023 IEEE 5th International Conference on Power，Intelligent Computing and Systems (ICPICS). Shenyang：IEEE，2023：492-496.

[29] MADDALI D. Convnext-Eesnn：An Effective Deep Learning Based Malware Detection in Edge Based IIOT[J]. Journal of Intelligent & Fuzzy Systems，2024，46(4)：10405-10421.

（编辑：杜娟）