基于图像域的轻量级恶意软件分类方法研究

孙敬张[1,2]，程轶男[1]，邹炳慧[1]，乔彤华[1]，符思政[1]，张琪[3]，曹春杰[1,2]
（1.海南大学网络空间安全学院（密码学院），海南 海口 570228；2.密码与跨境数据安全海南省工程研究中心，海南 海口 570228；
3.澳门城市大学数据科学学院，澳门 999078）

摘　要：针对传统恶意软件家族分类方法部署成本高和预测时间长等问题，提出了一种轻量的恶意软件可视化分类方法。首先，提出对比度受限双三次插值与高斯模糊算法，解决恶意软件图像大小不平衡及噪声过多的问题。其次，为应对恶意软件特征间关联捕获困难和现有注意力模块复杂度高的问题，提出轻量通道注意力机制，重点关注信息量更大的特征，结合深度可分离卷积减少模型参数。在MalImg、BIG2015和BODMAS这3个大型数据集上进行实验，该模型对恶意软件家族分类的准确率分别达到99.68%、99.45%和93.12%，模型大小分别为442 KB、414 KB和423 KB，预测时间分别为14.12 ms、11.09 ms和4.11 ms，证明了该方法在准确率、模型大小和推理速度上的先进性。

关键词：恶意软件分类；图像增强；轻量级模型；轻量通道注意力
中图分类号：TP309.5
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025035

Research on lightweight malware classification method based on image domain

SUN Jingzhang[1,2], CHENG Yinan[1], ZOU Binghui[1], QIAO Tonghua[1], FU Sizheng[1], ZHANG Qi[3], CAO Chunjie[1,2]
1. School of Cyberspace Security (School of Cryptology), Hainan University, Haikou 570228, China
2. Hainan Provincial Engineering Research Center of Cryptology and Cross-Border Data Security, Haikou 570228, China
3. Faculty of Data Science, City University of Macau, Macau 999078, China

Abstract: To address the high deployment costs and long prediction times associated with traditional malware classification methods, a lightweight malware visualization classification method was proposed. Firstly, a CBG algorithm was introduced to solve the problems of imbalanced image sizes and excessive noise in malware images. Then, to capture feature relationships effectively and reduce computational complexity, a lightweight channel attention mechanism was implemented. This mechanism guided the model to focus on more informative features, while depthwise separable convolution further reduced the number of model parameters. Experimental results on three large malware datasets, MalImg, BIG2015, and BODMAS, demonstrate that the proposed model achieved classification accuracies of 99.68%, 99.45%, and 93.12%, with model sizes of 442 KB, 414 KB, and 423 KB, and prediction times of 14.12 ms, 11.09 ms, and 4.11 ms per image, respectively. This method demonstrates state-of-the-art performance in accuracy, model size, and inference speed.

Keywords: malware classification, image enhancement, lightweight model, lightweight channel attention

收稿日期：2024–10–23；**修回日期：**2025–02–18
通信作者：曹春杰，caochunjie@hainanu.edu.cn
基金项目：海南省科技人才创新基金资助项目（No.KJRC2023B13, No.KJRC2023D30）
Foundation Items: Hainan Province Science and Technology Talents Innovation Project (No.KJRC2023B13, No.KJRC2023D30)

0 引言

尽管电子邮件已有50多年的历史，但它至今仍是企业、政府机构以及个人主要的通信工具之一。据统计，2023年全球电子邮件用户数量达到了约43.7亿。然而，正是因为其庞大的用户基础，电子邮件成为网络犯罪分子的首选攻击渠道[1]。尤其是通过邮件附件传播的恶意软件，对网络安全构成严重威胁。一旦用户不慎打开这些附件，不仅可能导致个人信息和企业数据的泄露，而且可能引发整个网络系统的瘫痪。据趋势科技（Trend Micro）报告显示，仅2023年就检测并阻止了1 910万个恶意软件文件，与2022年的检测量相比大幅增长了349%；由于电子邮件附件中钓鱼链接的使用增多，其中已知的恶意软件文件数量也大幅增加，共检测到1 600万个，增长了3 079%。

恶意软件通过电子邮件附件传播的机制不仅提高其入侵的隐蔽性，也大大增加了传统安全防御措施的检测难度。为了应对这一问题，在邮件附件端设置恶意软件实时扫描系统极为重要。现有的恶意软件分类模型在处理邮件附件时受限于模型体积和推理速度，导致高计算资源消耗和扫描时延，难以满足在线扫描系统的需求，因此，需要在保持分类准确性的同时减少模型体积并提高推理速度。

随着深度学习等现代技术的兴起，研究者开始探索自动化特征提取和分类的方法，以更有效地识别和应对日益复杂的网络安全威胁[2-4]，恶意电子邮件也不例外[5]。Cohen等[6]提出一种监督分类机器学习算法，通过标记的训练实例进行学习，生成一个能够预测未知恶意邮件附件类别的模型。Qbeitah等[7]提出了一种标准的深度学习分析方法，对网络钓鱼电子邮件中的动态恶意软件进行分析，在受控环境中运行恶意样本以调查其行为。尽管这些深度学习方法在恶意软件识别上表现出色，但未针对在线邮件附件扫描的实时需求优化。在处理数以亿计的邮件交流时，扫描速度是衡量系统性能的关键指标。轻量级模型通过减少复杂度和参数数量，显著加快推理速度，减少处理时延，从而提升邮件系统的响应时间和交流效率。

本文主要的研究工作如下。

1) 本文提出了一种图像预处理方法，旨在为恶意软件图像分类任务提供更有效的输入数据，称为对比度受限双三次插值与高斯模糊算法（CBG, contrast-limited bicubic interpolation and Gaussian-based algorithm）。该方法结合限制对比度直方图自适应均衡（CLAHE , contrast limited adaptive histogram equalization）算法、双三次插值法和高斯模糊，能够显著增强图像对比度，规范图像尺寸，并有效减少噪声，从而提高后续恶意软件分类模型的准确性和鲁棒性。

2) 本文设计了一种基于轻量通道注意力机制的模型，用于高效分类恶意软件图像。通过轻量通道注意力模块引导模型重点关注更具信息量的特征，实现更有效的特征提取；同时使用深度可分离卷积进一步减少模型参数量，使其更适合邮件附件的在线恶意软件扫描场景。

3) 本文在3个大型恶意软件数据集MalImg、BIG2015和BODMAS上进行了全面的实验评估，结果表明，所提方法在准确率、模型大小和推理速度上取得了最先进的性能。具体为准确率分别达到99.68%、99.45%和93.12%，模型大小分别为442 KB、414 KB和423 KB，预测其中一张恶意软件图像分别只需要14.12 ms、11.09 ms和4.11 ms。

1 相关工作

1.1 基于可视化的恶意软件分类方法

基于可视化的恶意软件分类方法将恶意软件以二进制形式转换为像素点并构成图像，不需要特征工程。Kang等[8]将Android应用程序的Dalvik可执行（Dex, dalvik executable）文件转换为图像，该文件作为Android应用的编译字节码的容器，可将其二进制代码转化为图像形式。Huang等[9]将Dex文件三等分后转化为RGB图像，利用多通道卷积运算的优势，但也引入不必要的噪声干扰分类。Yuan等[10]将恶意软件的二进制序列转换为固定大小的马尔可夫图像。恶意软件的特征往往和纹理有关，相较于RGB图像和马尔可夫图像，灰度图像的优势在于无须复杂的数学模型来描述统计特性，因此操作更为简单，且能更好地表示恶意软件的结构特征。

1.2 基于轻量化模型的恶意软件分类方法

面对邮件附件在线扫描的实时性要求，需要开发轻量级模型以降低计算资源的消耗和时延。Gao等[11]提出了一种基于改进轻量级神经网络的分类模型，用于提高Android恶意软件检测性能。通过

基于局部信息熵的图像生成技术构建特征向量，优化 ESPNetV2 模型，并提出 Mal-WGANGP 生成对抗样本增强模型。Ma 等[12]设计了一种名为 MCADS（MLP-CNN unified android malware detection system）的轻量级 Android 恶意软件检测系统，使用增强型多层感知器进行初步分析，提出一种新的轻量级卷积神经网络变体用于进一步分析。Gu 等[13]提出了一种基于图神经网络的 Android 恶意软件检测框架 GSEDroid，利用 CodeBERT 和 TextCNN 构建轻量级嵌入模型，结合 API 调用图和权限特征，将检测任务转化为图分类问题。Zou 等[14]提出一种名为 FACILE 的轻量级胶囊网络，使用更少胶囊的同时降低恶意软件分类错误率。尽管上述轻量级恶意软件分类方法减少了模型参数，但仍存在参数较多、特征提取范围有限的问题。与之前的研究相比，本文提出了一种更加准确和高效的恶意软件分类方法。

2　本文方法

为了保障邮件传输过程中的安全性和稳定性，本文提出了一种轻量级的恶意软件分类方法，专门用于邮件附件的在线扫描系统，以应对恶意软件威胁。该方法结合了恶意软件特征提取与图像分类技术的需求，具有显著的轻量级特性和高效的分类性能，本文模型架构如图 1 所示。

2.1　数据预处理模块

为了避免烦琐且耗时的特征工程，并更好地适应邮件附件在线扫描系统的实时性要求，本文采用将恶意软件二进制数据转换为图像的方式进行处理。通过这种方式，恶意软件样本的每个字节都被映射为灰度图像的一个像素值，这种转换方式能够有效地保留恶意软件的二进制信息，同时将其转化为图像形式以便利用现有图像分类模型进行高效分类。在这一基础上，本文设计了图像增强算法，通过对图像的标准化处理和增强，确保分类模型能更好地识别恶意软件中的重要特征。以下为具体步骤。

首先，将恶意软件的二进制数据转换为灰度图像。通过读取二进制数据流，将每个 8 位无符号整数映射为 0~255 的灰度像素值。图像宽度固定为 256 像素，图像高度根据恶意软件样本的字节数确定，因而不同样本的图像高度不同。

在得到可视化的灰度图像后，接下来采用 CBG 算法规范图像大小，进行图像增强。首先对恶意软件灰度图像应用 CLAHE 算法[15]，增强图像的局部对比度，突出图像中的细节信息。CLAHE 是一种自适应直方图均衡化方法，通过限制对比度增强的幅度，避免传统直方图均衡化方法可能引入的噪声和伪影。CLAHE 的基本原理是将图像分成若干小的矩形区域，每个小区域大小相等且互不相叠，对每个小区域分别进行直方图均衡化，并在均衡化过程中限制对比度的增强幅度，避免全局对比度调整可能引起的图像过增强和噪声放大问题。这一处理步骤能够帮助模型识别图像中的微小变化，尤其是在恶意软件特征的细节部分。具体计算式为

$$H_{\text{clip}}(i)=\min\left(H(i),\frac{\text{clipLimit}}{N}\right) \tag{1}$$

其中，$H(i)$是灰度级 i 的直方图值，clipLimit 是对比度限制参数，N 是图像像素总数。本文将 clipLimit 设置为 2.0，限制对比度增强的幅度。

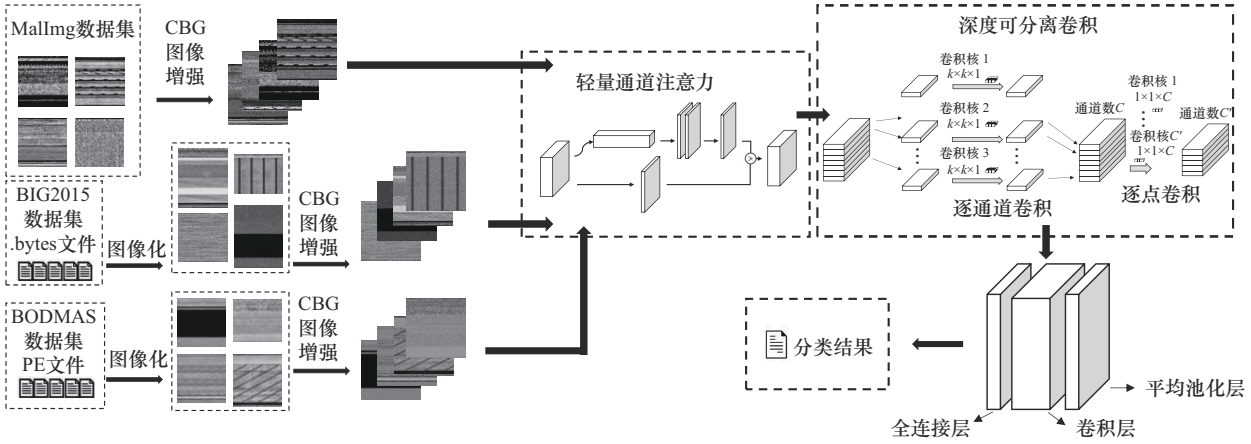


图 1　本文模型架构

然后，通过双三次插值法对图像进行插值处理，使所有图像的尺寸一致，以便对不同恶意软件样本进行比较。该方法通过考虑邻近的 16 个像素点来计算插值结果，使图像在放大或缩小时保持较高的质量。双三次插值的基本计算式为

$$f(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3}a_{ij}x^{i}y^{j} \tag{2}$$

其中，a_{ij}是插值系数，x和y是像素坐标。在本文中，使用 OpenCV 库的 cv2.resize 函数并设置 interpolation=cv2.INTER_CUBIC 来实现双三次插值，以提高图像缩放后的质量。

最后，通过高斯模糊对图像进行平滑处理，减少噪声，保证分类模型专注于有效的恶意软件特征而不受干扰。高斯模糊的基本计算式为

$$G(x,y)=\frac{1}{2\pi\sigma^{2}}\exp\left(-\frac{x^{2}+y^{2}}{2\sigma^{2}}\right) \tag{3}$$

其中，σ是标准差，x和y是像素坐标。本文将高斯核的大小设置为 5 × 5 进行平滑处理，将标准差设置为 0，算法自动计算最合适的模糊程度。

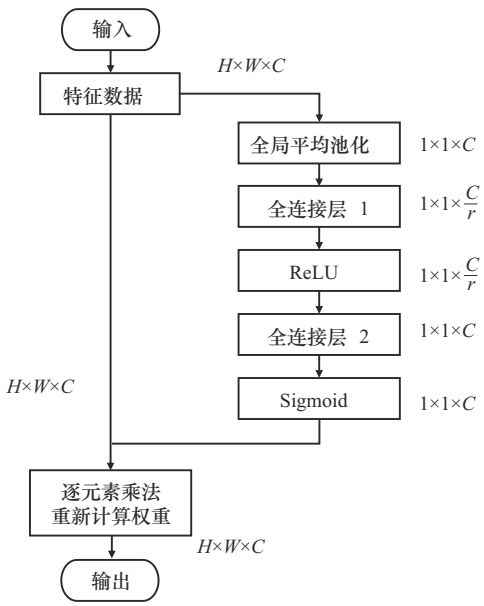
2.2　轻量通道注意力模块

在深度学习模型用于恶意软件分类的方法中，压缩激励注意力模块（SEAM, squeeze and excitation attention module）[16]通过全局池化层计算通道注意力，提升了模型效率和性能。但 SEAM 的降维和全连接层方法容易导致信息丢失并增加计算复杂度。为了确保系统在高并发情况下依然能快速响应和处理大量邮件附件，本文提出了轻量通道注意力（LCA , lightweight channel attention）模块，通过引导模型关注更具信息量的特征，结合稀疏卷积和一维卷积，增强特征表示的同时显著减少计算开销和参数量。通过这种设计，LCA 能够更好地捕捉恶意软件的关键信息，并减少对非恶意特征的干扰，从而提高分类精度。SEAM 与 LCA 对比如图 2 所示。

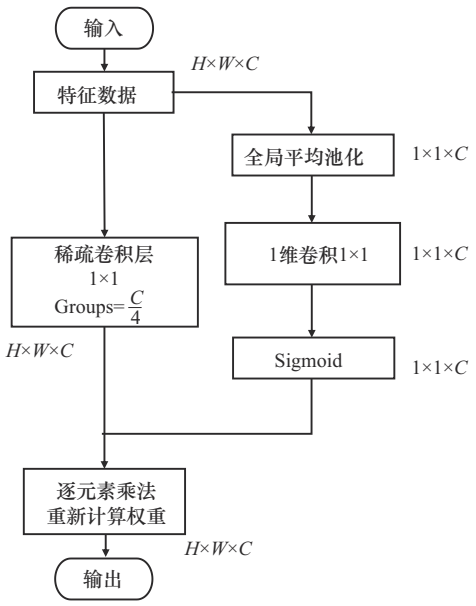
首先，LCA 模块利用全局平均池化提取全局特征，这是为了捕捉输入图像的全局上下文信息；再通过一维卷积进行通道间的轻量级交互，避免了 SEAM 中全连接层带来的高计算成本。随后，模块引入稀疏卷积，通过将通道数减少至$\frac{1}{4}$来减少计算开销，同时保留重要信息交互。最后，通过逐元素乘法重新计算权重，实现全局和局部特征的有效融合，从而有效地重新计算通道注意力权重。这种方法能够引导模型关注更具信息量的特征，同时抑制噪声和冗余信息，从而提高分类精度。

2.3　深度可分离卷积

传统卷积使用多通道卷积核在输入特征图上滑动加权求和，而深度可分离卷积[17]通过将复杂的多通道卷积操作分解为逐通道卷积和逐点卷积，进一步减少了参数数量和计算量，加快卷积操作的执行速度，提高了在线扫描系统的实时性。传统卷积



(a) SEAM



(b) LCA

图2　SEAM 与 LCA 对比

和深度可分离卷积对比如图3所示。

在逐通道卷积中，每个输入通道独立应用一个$k \times k$的卷积核。相对于使用$k \times k \times C$大小的卷积核对所有C个通道同时进行卷积，逐通道卷积大幅减少了参数数量。随后是逐点卷积，它用一个1×1的卷积核整合逐通道卷积的输出，混合各个通道信息。其参数数目为$1 \times 1 \times C \times C'$，其中$C'$是输出特征图的通道数。

相比传统卷积，深度可分离卷积通过以上两步分别对空间特征和通道特征进行处理，进而显著减少整体所需的参数量和计算。总体参数量由传统卷积$k \times k \times C \times C'$减少到$k \times k \times C + 1 \times 1 \times C \times C'$。在保证分类性能的同时，大幅降低恶意软件分类过程中的计算复杂度，使在有限的计算资源下能够快速完成恶意软件分类。

3　实验设计与分析

3.1　数据集与实验设置

3.1.1　恶意软件数据集

MalImg、BIG2015和BODMAS数据集中的恶意软件家族分布如图4所示，其中，外圈英文表示各恶意软件家族的名称，数字表示各家族的恶意软件样本数。

数据集MalImg[18]：包含大量恶意软件样本，这些样本被转换成图像格式组成数据集，无须自行转换，其家族分布情况如图4(a)所示。MalImg数据集包含了来自25个不同家族的9 339个恶意软件样本的字节图。每个样本都是一个可执行的二进制文件，这些文件被转换成灰度图像，其中每个像素的亮度值对应文件内容的字节值。

数据集BIG2015[19]：由微软公司提供，包含多种类型的恶意软件，这些样本来自不同的恶意软件家族，反映了现实世界中恶意软件的多样性，其家族分布情况如图4(b)所示。每个恶意软件样本通常包含一个.asm（反汇编）文件和一个编译后的.bytes（字节码）文件，本文在预处理模块将.bytes文件转换为恶意软件图像。

数据集BODMAS[20]：包含从2019年8月至2020年9月收集的57 293个恶意软件样本，共有581个家族。本文选择了其中家族样本数均超过1 000的14个家族进行实验，其家族分布情况如图4(c)所示。每个恶意软件样本都包括其SHA-256哈希、原始PE二进制文件和预提取的特征向量。本文在预处理模块将原始PE二进制文件转换为恶意软件图像。

3.1.2　实验设置

本文实验均在云服务器上进行，使用GPU型号为Geforce RTX 3090Ti，CPU型号为Intel(R) Xeon(R) CPU E5-2686 v4，在PyTorch 1.13.1环境中使用Python3.8实现。

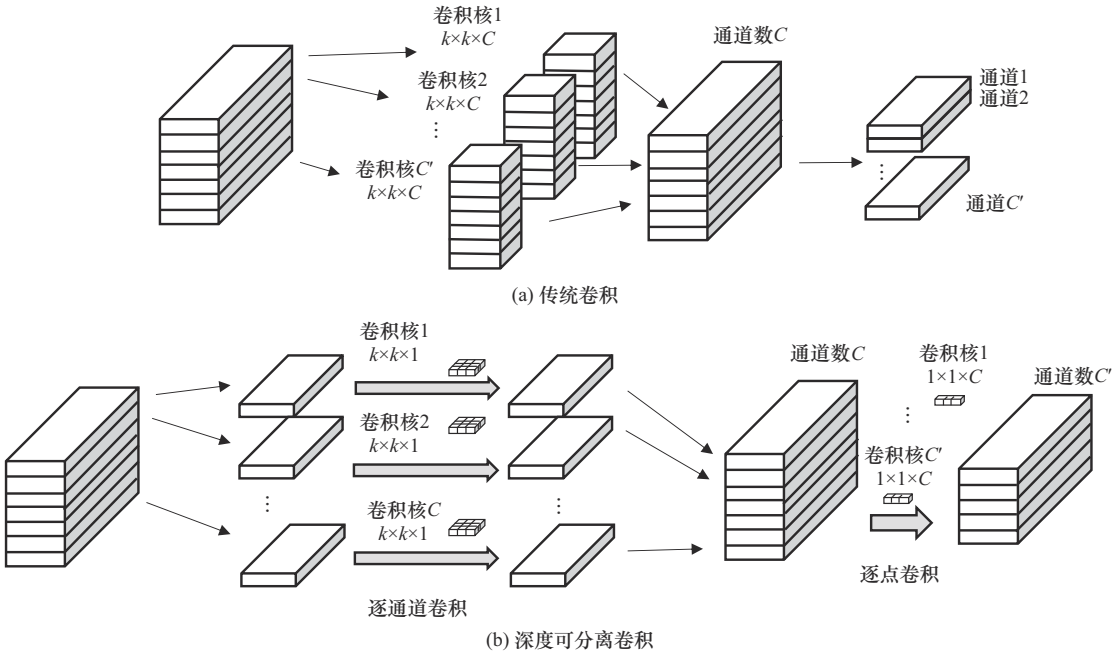
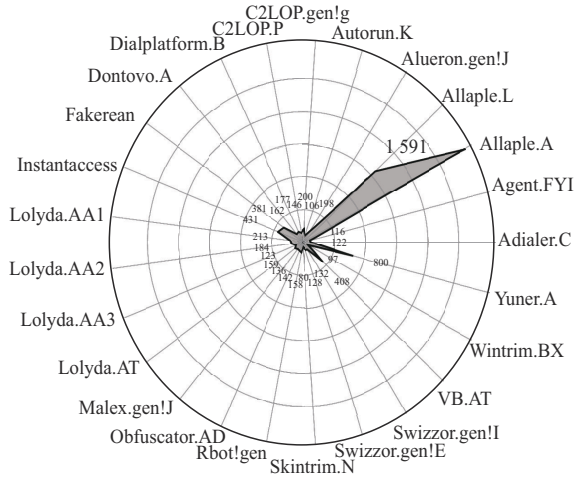
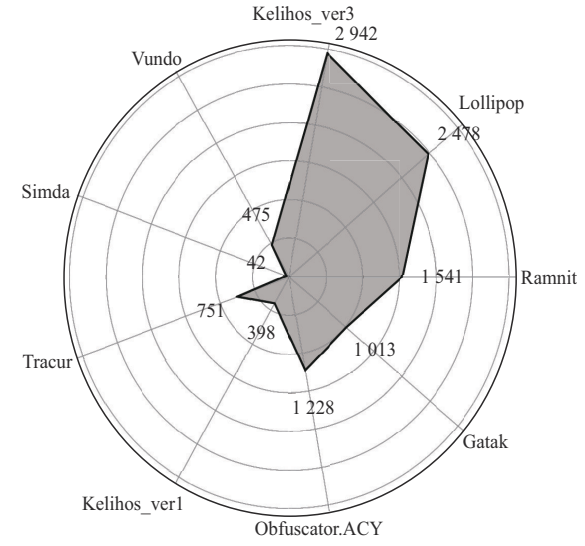


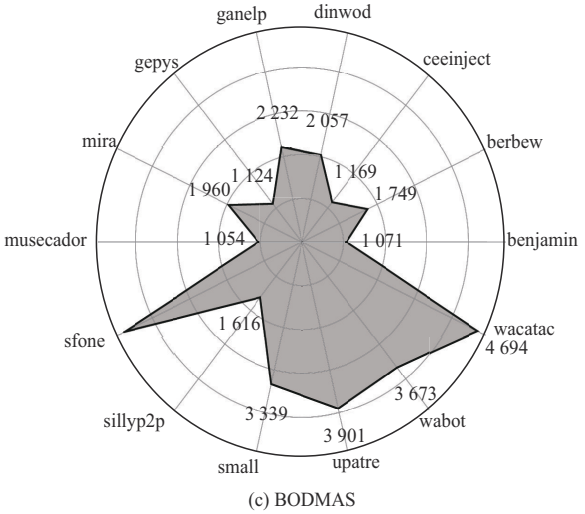
图3　传统卷积和深度可分离卷积对比



(a) MalImg



(b) BIG2015



(c) BODMAS

图4　MalImg、BIG2015和BODMAS数据集中的恶意软件家族分布

由于本文实验使用文献[21]作为基本框架，而Batch Size、Learning Rate、Epochs这3个超参数的配置在先前的研究中已经被证明是有效的，所以沿用了先前研究中的超参数配置。但由于模型架构的修改，最优参数也可能变动，因此在该配置基础上选择了部分范围的数据进行遍历，得到了最优配置结果，超参数的选择范围与最佳值如表1所示。

表1　超参数的选择范围与最佳值

超参数	选择范围	最佳值
Batch Size	{8, 16, 32, 64, 128}	16
Learning Rate	{0.003 0, 0.003 4, 0.003 8, 0.004 2, 0.004 6}	0.003 8
Epochs	{25, 50, 100, 150, 200}	100

为了全面评价模型的泛化能力，本文在部分实验中采用了75%训练集和25%测试集的划分方式，其中训练集用于模型训练，测试集用于评估模型性能。此外，为了进一步提升评估的可靠性并减少数据划分的偶然性，本文在消融实验部分采用了5折交叉验证的方法。将整个数据集等分为5个子集，在每一轮验证过程中选取4个子集作为训练数据，剩下的一个子集则用作测试。这样的设置确保了每个子集都有机会作为测试集被使用，从而提高了评估的准确性和可靠性。值得注意的是，除消融实验外，其他实验均采用了固定的训练集与测试集划分方式，即训练集占75%，测试集占25%，以保持实验条件的一致性。

本文选用了准确率（Accuracy）、精确率（Precision）、召回率（Recall）和F1分数（F1 Score）4个指标来评估不同方法的性能，其定义分别为

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

其中，真阳性（TP）表示模型正确地预测了正类，真阴性（TN）表示模型正确地预测了负类，假阳性（FP）表示模型错误地预测了正类，假阴性（FN）表示模型错误地预测了负类。

3.1.3　基线介绍

本文方法将在数据集MalImg、BIG2015和BODAMS上进行训练，并与以下恶意软件分类方法进行比较。

IMCFN[22]：通过将恶意软件二进制转换为彩色图像并使用微调的卷积神经网络架构，有效分类恶意软件家族。

MalSort[23]：利用带掩码的自监督框架和Swin Transformer模型来识别和分类恶意软件，而无须依赖于复杂的特征工程。

UDA[24]：提出一种高效的深度无监督领域自适应方法，用于未知恶意软件的检测，通过自学习的方式适应不同的数据分布。

MCFT-CNN[25]：利用传统和迁移学习技术对物联网中的恶意软件进行分类，通过微调卷积神经网络来提高检测的准确性和效率。

Malconv[26]：提出了一种从原始字节序列中检测恶意软件的方法。针对处理超过200万时间步的序列问题提出了一个初步的解决方案，该方案对序列长度的线性复杂度具有依赖性，并能够识别二进制文件中的可解释子区域。

Malconv+GCG[27]：提出了一种新的方法来处理极长序列的恶意软件检测问题，开发了一种新的时间最大池化方法，使所需的内存与序列长度无关，从而显著提高了内存效率，并加快了训练速度。

DANN+SGD[28]：提出了一种结合深度学习技术的新方法，通过有效降低数据集的高维性来检测异常，而无须依赖标记的训练样本。

DesNet[29]：一种深度卷积神经网络架构，特点是每一层与前面所有层直接连接，促进特征重用，缓解梯度消失问题，提升训练效率和模型性能，通过密集连接提高网络的表达能力。

ResNet50[30]：一个深度残差网络，包含50层，通过引入残差连接解决了深层网络训练中的梯度消失问题。它使用跳跃连接将输入直接传递到输出，显著提升了模型的训练效率和分类精度。

3.2　数据增强实验评估

将本文模型在数据增强后的恶意软件数据集MalImg、BIG2015和BODMAS上进行实验评估，混淆矩阵如图5所示。为了证明本文数据增强方法的有效性，将数据增强前后的两组恶意软件数据集MalImg和BIG2015分别应用于恶意软件分类常用模型DesNe、ResNet50和本文模型，CBG数据增强前后的恶意软件数据集MalImg应用于各模型对比结果如表2所示。CBG数据增强前后的恶意软件数据集BIG2015应用于各模型对比结果如表3所示。

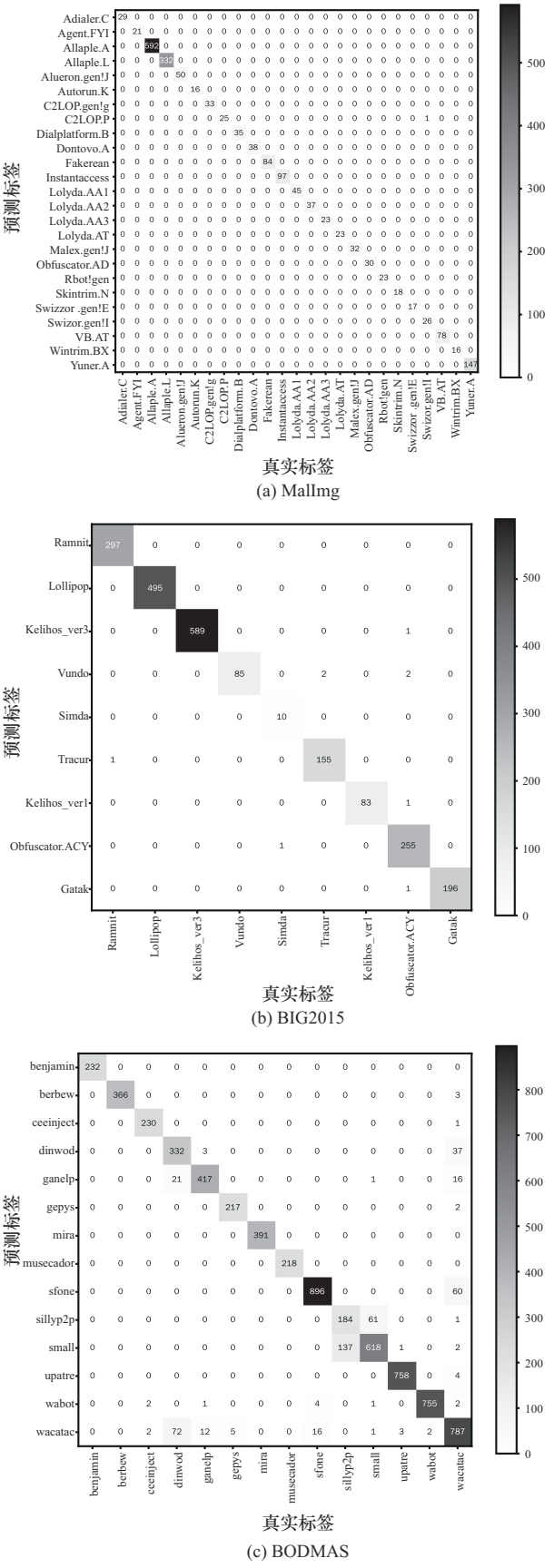


图5　本文模型在数据集MalImg、BIG2015和BODMAS上的混淆矩阵

表 2 CBG 数据增强前后的恶意软件数据集 MalImg 应用于各模型对比结果

数据集	模型	准确率	精确率	召回率	F1 分数
原始 MalImg	DesNet	96.20%	95.10%	96.10%	95.40%
原始 MalImg	ResNet50	98.53%	95.38%	95.68%	95.50%
原始 MalImg	本文方法	98.90%	98.58%	98.49%	98.52%
增强后 MalImg	DesNet	98.50%	96.90%	96.60%	96.70%
增强后 MalImg	ResNet50	99.14%	97.67%	97.69%	97.62%
增强后 MalImg	本文方法	**99.89**%	**99.73**%	**99.72**%	**99.73**%

表 3 CBG 数据增强前后的恶意软件数据集 BIG2015 应用于各模型对比结果

数据集	模型	准确率	精确率	召回率	F1 分数
原始 BIG2015	DesNet	95.80%	95.00%	95.70%	95.80%
原始 BIG2015	ResNet50	97.80%	95.46%	95.05%	95.19%
原始 BIG2015	本文方法	98.64%	96.87%	97.34%	97.10%
增强后 BIG2015	DesNet	97.30%	95.90%	96.60%	96.70%
增强后 BIG2015	ResNet50	98.23%	96.98%	96.70%	96.82%
增强后 BIG2015	本文方法	**99.77**%	**99.71**%	**99.76**%	**99.76**%

本文方法在 MalImg 数据集上的混淆矩阵如图 5(a)所示。从图 5(a)可以看出，本文方法误将 Lolyda.AA2 的部分样本识别为 Lolyda.AA1 家族，以及误将 Swizzor.gen!I 家族的部分样本识别为 Swizzor.gen!E 家族，分类其他家族时在各项评估指标上均达到了理论最优值。原因是误判的这 2 个家族在可视化特征上分别存在高度相似性，使分类模型难以区分。

本文方法在数据集 BIG2015 上的混淆矩阵如图 5(b)所示。从图 5(b)可以看出，本文方法分类 Kelihos_ver3 和 Simda 家族时在各项评估指标上均达到了理论最优值，分类其他家族时各性能指标略差。不同恶意软件家族之间可能存在某些相似特征，这会导致模型在区分这些家族时出现混淆。

本文方法在数据集 BODMAS 上的混淆矩阵如图 5(c)所示。从图 5(c)可以看出，本文方法分类 dinwod、sillyp2p、small 和 wacatac 这 4 个家族时准确率较低，尤其是 sillyp2p 和 small 这 2 个家族容易被互相误判，原因是这 2 个家族虽然在功能上有所不同，但均涉及 P2P 通信和文件操作等行为[20]，这些行为通常使用相同的 API 函数，导致了字节序列在可视化处理后具有很高的相似性。

从表 2 和表 3 可以看出，CBG 数据增强后的 2 个数据集 MalImg、BIG2015 均比数据增强前有更好的性能，且本文提出方法比 DesNet、ResNet50 有更好的评估指标。

3.3 基于本文提出模型的实验评估

为了验证本文模型的有效性，将进行本文方法与其他恶意软件分类方法的对比实验，本文方法各模块的消融实验，以及与其他轻量化模型的对比实验。

3.3.1 和其他模型的对比实验

为了验证本文模型的有效性，将本文方法与最先进的恶意软件分类方法在数据集 MalImg、BIG2015 和 BODMAS 上进行对比，记录了各方法的准确率、精确率、召回率和 F1 分数 4 个评估指标，如表 4 所示。在所有评估指标上，本文方法均表现出色，特别是在精确率和 F1 分数上，体现了

表 4 本文方法与其他方法在数据集 MalImg、BIG2015 和 BODMAS 上的对比结果

模型	MalImg				BIG2015				BODMAS			
	准确率	精确率	召回率	F1 分数	准确率	精确率	召回率	F1 分数	准确率	精确率	召回率	F1 分数
IMCFN	98.82%	98.85%	98.81%	98.75%	97.46%	97.58%	97.84%	97.21%	—	—	—	—
MalSort	98.28%	98.19%	98.18%	98.28%	97.85%	97.85%	97.63%	97.85%	—	—	—	—
UDA	95.63%	95.30%	95.34%	94.98%	95.04%	95.10%	94.24%	94.65%	—	—	—	—
MCFT-CNN	99.19%	97.72%	97.76%	97.68%	98.64%	98.55%	96.79%	97.22%	—	—	—	—
Malconv	98.36%	97.67%	97.12%	97.63%	96.20%	95.93%	96.07%	96.04%	93.08%	92.07%	91.92%	92.44%
Malconv+GCG	98.37%	98.06%	98.01%	98.10%	96.77%	96.12%	95.96%	96.05%	92.10%	92.14%	91.56%	92.13%
DANN+SGD	96.58%	95.49%	96.07%	95.89%	91.49%	88.87%	87.81%	88.32%	90.65%	87.81%	88.87%	87.95%
本文方法	**99.89**%	**99.73**%	**99.72**%	**99.73**%	**99.77**%	**99.71**%	**99.76**%	**99.76**%	**93.12**%	**93.89**%	**92.96**%	**93.29**%

较强的稳定性和较高的准确性。这是由于CBG算法使恶意软件的特征更加显著，有助于模型从复杂的恶意软件图像中提取出更多有效特征，同时轻量级通道注意力机制提高了对关键信息的关注能力。

3.3.2　消融实验

为了验证轻量通道注意力模块和深度可分离卷积模块分别对本文模型的影响，本节进行了消融实验。在进行每项消融研究时，实施了5折交叉验证，从而得到了5个独立的模型，对5个模型分别进行评估，选取准确率、精确率、召回率和F1分数4个指标参数的5次实验平均值、可训练参数量及模型体积大小，本文方法在数据集MalImg、BIG2015、BODMAS上的消融实验对比结果分别如表5~表7所示。

表5~表7的数据显示，本文的轻量通道注意力模块和深度可分离卷积模块显著提升了模型性能。虽然注意力模块增加了模型的参数量，但不影响性能的提升。在数据集MalImg中，加入轻量通道注意力模块后，仅增加3 209个参数量，准确率提升2.67%，精确率提升3.88%，召回率提升3.86%，F1分数提升4.15%。在数据集BIG2015中，增加1 161个参数量，准确率提升2.76%，精确率提升3.45%，召回率提升4.38%，F1分数提升4.17%。在数据集BODMAS中，增加1 801个参数量，准确率提升1.40%，精确率提升2.35%，召回率提升1.78%，F1分数提升2.18%。

将深度可分离卷积替换为传统卷积的实验可以发现，深度可分离卷积在不降低性能的情况下极大减少了模型参数量。在数据集MalImg上，参数量降低至原来的45.989%；在数据集BIG2015上，参数量降低至原来的43.884%；在数据集BODMAS上，参数量降低至原来的44.552%。

表5　本文方法在数据集MalImg上的消融实验对比结果

模型	准确率	精确率	召回率	F1分数	参数量/个	模型大小/KB
本文方法无轻量通道注意力模块	97.22%	95.85%	95.86%	95.58%	104 473	428
本文方法无深度可分离卷积模块	98.55%	96.98%	97.09%	96.96%	234 146	933
本文方法	**99.89%**	**99.73%**	**99.72%**	**99.73%**	**107 682**	**442**

表6　本文方法在数据集BIG2015上的消融实验对比结果

模型	准确率	精确率	召回率	F1分数	参数量/个	模型大小/KB
本文方法无轻量通道注意力模块	97.01%	96.26%	95.38%	95.59%	99 337	408
本文方法无深度可分离卷积模块	98.40%	96.88%	95.89%	96.27%	229 010	913
本文方法	**99.77%**	**99.71%**	**99.76%**	**99.76%**	**100 498**	**414**

表7　本文方法在数据集BODMAS上的消融实验对比结果

模型	准确率	精确率	召回率	F1分数	参数量/个	模型大小/KB
本文方法无轻量通道注意力模块	91.72%	91.54%	91.18%	91.11%	100 942	415
本文方法无深度可分离卷积模块	92.81%	92.11%	92.23%	92.53%	230 615	920
本文方法	**93.12%**	**93.89%**	**92.96%**	**93.29%**	**102 743**	**423**

表8　本文方法与其他轻量化方法在数据集MalImg上的对比结果

模型	准确率	精确率	召回率	F1分数	参数量/个	模型大小/MB	推理时间/ms
ShuffleNetV2	98.29%	97.47%	97.82%	97.51%	1 279 229	5.10	16.00
MobileNetV3-Small	94.92%	93.22%	93.87%	93.30%	1 543 481	6.10	15.30
GhostNet	98.89%	98.36%	98.33%	98.31%	3 933 533	15.30	15.17
RepViT	99.67%	99.06%	99.12%	99.08%	2 174 317	8.60	15.20
本文方法	**99.89%**	**99.73%**	**99.72%**	**99.73%**	**107 682**	**0.43**	**14.12**

表 9 本文方法与其他轻量化方法在数据集 BIG2015 上的对比结果

模型	准确率	精确率	召回率	F1 分数	参数量/个	模型大小/MB	推理时间/ms
ShuffleNetV2	97.84%	97.21%	97.14%	97.30%	1 262 829	5.00	14.00
MobileNetV3-Small	93.31%	92.78%	91.56%	92.10%	1 527 081	6.00	12.10
GhostNet	98.78%	98.28%	98.17%	98.16%	3 913 037	15.20	11.89
RepViT	98.94%	98.73%	96.30%	97.31%	2 169 181	8.50	11.60
本文方法	**99.77%**	**99.71%**	**99.76%**	**99.76%**	**100 498**	**0.40**	**11.09**

表 10 本文方法与其他轻量化方法在数据集 BODMAS 上的对比结果

模型	准确率	精确率	召回率	F1 分数	参数量/个	模型大小/MB	推理时间/ms
ShuffleNetV2	90.43%	90.87%	90.18%	90.33%	1 267 954	5.00	4.90
MobileNetV3-Small	91.15%	90.23%	89.64%	90.17%	1 532 206	6.00	4.40
GhostNet	90.18%	90.05%	90.13%	90.01%	3 919 442	15.20	4.30
RepViT	91.27%	91.98%	90.70%	90.63%	2 170 786	8.60	4.50
本文方法	**93.12%**	**93.89%**	**92.96%**	**93.29%**	**102 743**	**0.41**	**4.11**

3.3.3 和其他轻量级模型的对比

由于邮件附件在线扫描的背景，要求本文方法与常用轻量化模型对比具有更高的准确率、更小的模型体积和更快的推理速度。将本文方法与 4 个主流的轻量化模型 ShuffleNetV2[31]、MobileNetV3-Small[32]、GhostNet[33]和 RepViT[34]在 3 个数据集上进行了比较，结果如表 8~表 10 所示。3 个数据集中图像的尺寸均为 160 像素×160 像素，所有实验参数与之前设置相同，记录了 4 个指标参数、模型大小、参数量和在 GPU（Geforce RTX 3090Ti）上每张恶意软件图像的推理时间。实验结果表明，本文方法训练时间更短，具有更少的参数量，在 GPU 上的预测时间更短，同时具有更高的准确率。

4 结束语

本文提出了一种 CBG 数据增强算法，实现了图像对比度的增强、分辨率的标准化和额外噪声的减少等功能，提出了基于轻量通道注意力模型的恶意软件分类方法，以高效分类邮件附件恶意软件。将提出模型在 3 个大型恶意软件数据集 MalImg、BIG2015 和 BODMAS 上进行实验，结果表明，本文方法在准确率、模型大小和推理速度上取得了最先进的性能。因此，部署在邮件附件在线扫描系统上可以有效防止恶意软件攻击，保护传输数据和文件的安全性和完整性。在处理恶意软件样本时，某些具有类似功能或使用相同 API 的恶意软件家族字节序列或可视化特征高度相似，这种相似性可能导致模型出现误判。在未来的研究中，可以结合更多的动态特征分析，如行为监测数据等，来提高模型对不同恶意软件家族的区分能力。在数据集中的某些样本较少的恶意软件家族中，模型容易由于训练数据不足而表现出较低的分类性能。虽然本文使用了数据增强和交叉验证等手段来缓解该问题，但在面对极少数样本时，模型的表现仍然不稳定。在未来的研究中，可以考虑引入更为先进的少样本学习方法[35]，帮助模型在小样本情况下提高性能。

参考文献：

[1] SETHURAMAN S C, DEVI PRIYA V S, REDDI T, et al. A comprehensive examination of email spoofing: Issues and prospects for email security[J]. Computers & Security, 2024, 137: 103600.

[2] 符思政, 曹春杰, 刘志远, 等. 用于攻击深度哈希图像检索模型的双分支自编码器网络[J]. 电信科学, 2023, 39(11): 96-106.
FU S Z, CAO C J, LIU Z Y, et al. Dual-branch autoencoder network for attacking deep hashing image retrieval models[J]. Telecommunications Science, 2023, 39(11): 96-106.

[3] FU S Z, CAO C J, TAO F J, et al. LGWAE: label-guided weighted auto-encoder network for flexible targeted attacks of deep hashing[C]//Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN). Piscataway: IEEE Press, 2023: 1-9.

[4] LIU Z Y, CAO C J, TAO F J, et al. Revisiting graph contrastive learning for anomaly detection[J]. arXiv Preprint, arXiv: 2305.02496, 2023.

[5] QIAO T H, CAO C J, ZOU B H, et al. A weighted discrete wavelet transform-based capsule network for malware classification[C]//Interna-

tional Conference on Pattern Recognition. Berlin: Springer, 2024: 259-274.

[6] COHEN Y, HENDLER D, RUBIN A. Detection of malicious webmail attachments based on propagation patterns[J]. Knowledge-Based Systems, 2018, 141: 67-79.

[7] QBEITAH M A, ALDWAIRI M. Dynamic malware analysis of phishing emails[C]//Proceedings of the 2018 9th International Conference on Information and Communication Systems (ICICS). Piscataway: IEEE Press, 2018: 18-24.

[8] KANG M, PARK J, PARK S, et al. Android malware family classification using images from dex files[C]//Proceedings of the 9th International Conference on Smart Media and Applications. New York: ACM Press, 2021: 181-186.

[9] HUANG H D , KAO H Y .R2-D2: color-inspired convolutional neural network (CNN)-based Android malware detections[C]//Proceedings of the 2018 IEEE International Conference on Big Data (Big Data). Piscataway: IEEE Press, 2018: 2633-2642.

[10] YUAN B G, WANG J F, WU P, et al. IoT malware classification based on lightweight convolutional neural networks[J]. IEEE Internet of Things Journal, 2022, 9(5): 3770-3783.

[11] GAO C X, DU Y, MA F, et al. A new adversarial malware detection method based on enhanced lightweight neural network[J]. Computers & Security, 2024, 147: 104078.

[12] MA R Z, YIN S N, FENG X, et al. A lightweight deep learning-based Android malware detection framework[J]. Expert Systems with Applications, 2024, 255: 124633.

[13] GU J T, ZHU H L, HAN Z W, et al. GSEDroid: GNN-based Android malware detection framework using lightweight semantic embedding[J]. Computers & Security, 2024, 140: 103807.

[14] ZOU B H, CAO C J, WANG L J, et al. FACILE: a capsule network with fewer capsules and richer hierarchical information for malware image classification[J]. Computers & Security, 2024, 137: 103606.

[15] REZA A M. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement[J]. Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology, 2004, 38(1): 35-44.

[16] WANG Y D, ZHANG J, KAN M N, et al. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation[C]// Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2020: 12272-12281.

[17] CHOLLET F. Xception: deep learning with depthwise separable convolutions[C]//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2017: 1800-1807.

[18] NATARAJ L, KARTHIKEYAN S, JACOB G, et al. Malware images: visualization and automatic classification[C]//Proceedings of the 8th International Symposium on Visualization for Cyber Security. New York: ACM Press,2011: 1-7.

[19] RONEN R, RADU M, FEUERSTEIN C, et al. Microsoft malware classification challenge[J]. arXiv Preprint, arXiv: 1802.10135, 2018.

[20] YANG L M, CIPTADI A, LAZIUK I, et al. BODMAS: an open dataset for learning based temporal analysis of PE malware[C]//Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW). Piscataway: IEEE Press, 2021: 78-84.

[21] ZOU B H, CAO C J, TAO F J, et al. IMCLNet: a lightweight deep neural network for image-based malware classification[J]. Journal of Information Security and Applications, 2022, 70: 103313.

[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] WANG F W, SHI X P, YANG F, et al. MalSort: lightweight and efficient image-based malware classification using masked self-supervised framework with Swin Transformer[J]. Journal of Information Security and Applications, 2024, 83: 103784.

[24] WANG F W, CHAI G F, LI Q R, et al. An efficient deep unsupervised domain adaptation for unknown malware detection[J]. Symmetry, 2022, 14(2): 296.

[25] SUDHAKAR, KUMAR S. MCFT-CNN: malware classification with fine-tune convolution neural networks using traditional and transfer learning in Internet of things[J]. Future Generation Computer Systems, 2021, 125: 334-351.

[26] RAFF E, BARKER J, SYLVESTER J, et al. Malware detection by eating a whole EXE[J]. arXiv Preprint, arXiv.1710.09435, 2017.

[27] RAFF E, FLESHMAN W, ZAK R, et al. Classifying sequences of extreme length with constant memory applied to malware detection[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(11): 9386-9394.

[28] MUNEER A, TAIB S M, FATI S M, et al. A hybrid deep learning-based unsupervised anomaly detection in high dimensional data[J]. Computers, Materials & Continua, 2022, 70(3): 5363-5381.

[29] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks[C]//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2017: 2261-2269.

[30] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2016: 770-778.

[31] MA N N, ZHANG X Y, ZHENG H T, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[C]// Computer Vision – ECCV 2018. Berlin: Springer, 2018: 122-138.

[32] HOWARD A, SANDLER M, CHEN B, et al. Searching for MobileNetV3[C]//Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2019: 1314-1324.

[33] HAN K, WANG Y H, TIAN Q, et al. GhostNet: more features from cheap operations[C]//Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2020: 1580-1589.

[34] WANG A, CHEN H, LIN Z J, et al. Rep ViT: revisiting mobile CNN from ViT perspective[C]//Proceedings of the 2024 IEEE/CVF Confer-

ence on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2024: 15909-15920.

[35] YE Z, WANG J, SUN T, et al. Cross-domain few-shot learning based on graph convolution contrast for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024, 62: 5504614.

[作者简介]

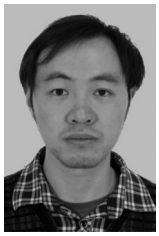
孙敬张（1993– ），男，海南海口人，博士，海南大学副研究员、博士生导师，主要研究方向为无线安全、深度学习、图像处理等。

程轶男（2000– ），女，浙江衢州人，海南大学硕士生，主要研究方向为人工智能安全、恶意软件分类等。

邹炳慧（1996– ），男，江西抚州人，海南大学博士生，主要研究方向为恶意代码分析、漏洞挖掘等。

乔彤华（1998– ），男，甘肃庆阳人，海南大学硕士生，主要研究方向为人工智能安全、恶意软件检测等。

符思政（1999– ），男，海南临高人，海南大学硕士生，主要研究方向为人工智能安全、对抗样本攻防、图像检索等。



张琪（1993– ），男，安徽淮北人，博士，澳门城市大学助理教授、硕士生导师，主要研究方向为机器学习、模式识别、图像处理等。

曹春杰（1977– ），男，河北衡水人，博士，海南大学教授、博士生导师，主要研究方向为无线网络安全、区块链、人工智能安全等。