计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(8):140−150 [doi: 10.15888/j.cnki.csa.009207]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于风格的数据集水印算法①

盛钡娜, 潘旭东, 张　谧

(复旦大学 计算机科学技术学院, 上海 200438)
通信作者: 盛钡娜, E-mail: bnsheng20@fudan.edu.cn

摘　要: 开源数据集加速了深度学习的发展, 但存在许多不合理使用数据集的现象. 为保护数据集的知识产权, 近期工作提出数据集水印算法, 在数据集发布前预先植入水印, 当模型在此数据集上训练时该水印会被附着在模型中, 之后通过验证可疑模型是否存在水印来追溯数据集的非法使用. 但已有数据集水印算法无法在小扰动下提供有效并且隐蔽的黑盒水印验证. 为解决这一问题, 本文首次提出利用独立于图像内容与标签的风格属性来植入水印, 并限制对原数据集的扰动不涉及标签的修改. 通过不引入图像内容与标签的不一致性和额外的代理模型保证水印隐蔽性和有效性. 在水印验证阶段仅使用可疑模型的预测结果通过假设检验给出判断. 本文在 CIFAR-10 数据集上与现有 5 种方法相比较, 实验结果验证了本文提出的基于风格的数据集水印算法的有效性与功能不变性. 此外, 本文开展的消融实验验证了本文所提的风格优化模块的必要性, 算法在不同超参设定以及不同数据集下的有效性.

关键词: 数据集水印; 数据集知识产权保护; 图像风格; 风格迁移; 假设检验

引用格式: 盛钡娜,潘旭东,张谧.基于风格的数据集水印算法.计算机系统应用,2023,32(8):140–150. http://www.c-s-a.org.cn/1003-3254/9207.html

Style-based Dataset Watermarking Algorithm

SHENG Bei-Na, PAN Xu-Dong, ZHANG Mi

(School of Computer Science, Fudan University, Shanghai 200438, China)

Abstract: Open-sourced datasets accelerate the development of deep learning, while unauthorized data usage frequently happens. To protect the dataset copyright, this study proposes the dataset watermarking algorithm. The watermark is embedded into the dataset before it is released. When the model is trained on this dataset, the watermark is attached to the model, which allows illegal dataset usage to be traced by verifying whether the watermark exists in a suspect model. However, existing dataset watermarking algorithms cannot provide effective and covert black-box verification under small perturbations. Given this problem, the method of embedding the watermark by a style attribute independent of the image content and label is proposed for the first time in this study, and the perturbation on the original dataset is constrained to avoid the modification of labels. The covertness and validity of the watermark are ensured without introducing the inconsistency between the image content and label or extra surrogate model. In the watermark verification stage, only the prediction results of the suspected model are applied to give the judgment via a hypothesis test. The proposed method is compared with the existing five methods on the CIFAR-10 dataset. The experimental results validate the effectiveness and fidelity of the proposed algorithm. Besides, the ablation experiments conducted in this study verify the necessity of the proposed style refinement module and the effectiveness of the proposed algorithm under various hyper-parameter settings and datasets.

Key words: dataset watermarking; dataset copyright protection; image style; style transfer; hypothesis test

① 收稿时间: 2023-01-07; 修改时间: 2023-03-01, 2023-03-14; 采用时间: 2023-03-23; csa 在线出版时间: 2023-05-22
CNKI 网络首发时间: 2023-05-24

高质量的数据集是深度学习蓬勃发展与广泛应用的基本前提[1–3], 例如, ImageNet 数据集[2] 及其 ILSVRC 挑战赛孕育了许多经典且至今仍被广泛使用的模型结构, 如 ResNet[3], VGG[4] 和 SENet[5] 等模型结构. 受开源精神影响, 很多公司与研究机构会公开发布所收集整理的数据集来加速学术研究进程. 但这类开源数据集通常不希望被用于未授权的商业用途中[1,2,6–8], 因此, 需要一种针对数据集的知识产权保护方案, 以验证商业模型是否存在在无授权数据集上训练的行为, 进而保护数据集所有者的合法权益.

数据集水印 (dataset watermarking) 这一机制应运而生[9–11]. 通常数据集水印方案包含两部分: 数据集水印植入模块与数据集水印验证模块. 前者是指在数据集发布之前将持有方指定的水印添加到数据集中. 不同于模型水印 (model watermarking)[12,13], 数据集持有方即水印嵌入者无法控制数据集窃取者会使用何种模型结构, 训练超参等信息, 因此要求所植入的水印能附着于任意在此数据集上训练所得到的模型中. 后者则是指给定可疑模型白盒或黑盒访问权限, 通过验证模型是否含有水印来判断该模型是否存在数据集的无授权使用行为.

在现有文献中, 仅有少数工作关注到数据集水印任务并提出相应的解决方案, 但这些工作都存在各式问题. Li 等人[9] 利用传统后门攻击[14] 对数据集植入后门, 之后通过检测模型是否存在后门来进行验证. 但水印植入过程涉及对原数据集标签的修改, 使得样本语义内容与其标签不一致, 进而缺乏隐蔽性. Sablayrolles 等人[10] 则是关注白盒验证场景, 在模型参数中植入水印. 这要求验证方拥有对可疑模型的白盒访问权限, 但目前大多商业模型提供的预测 API 接口仅返回模型预测标签或者各类置信度, 因此白盒验证假设是不切实际的. 近期, Li 等人[11] 针对上述两个问题提出了满足干净标签限制且能支持黑盒验证的 UBW-C 算法, 其中干净标签限制指水印嵌入中不扰动原数据标签, 保证数据语义内容与标签的一致性. 但这一算法在植入水印时需要引入代理模型, 使得其在不同模型结构下的水印效果欠佳, 尤其是限制植入水印能对训练集的可扰动比率较小时.

本文工作探索图像风格特征, 一个独立于图像语义内容与标签的特征, 来设计在干净标签和扰动比例较小限制下仍有效的且可以支持黑盒验证的数据集水印算法. 具体地, 本文利用风格迁移模型[15,16] 对目标类的少部分样本嵌入特定风格, 之后通过验证模型是否能将含有该指定风格的其他类图像分类为目标类别来判断模型是否在含水印数据集上训练获得. 此外, 为了使植入的水印更加隐蔽, 在水印植入阶段设计了风格优化模块, 对原本随机选取的风格在特定风格迁移模型和目标类别上优化, 使得在目标类中额外嵌入的水印更加隐蔽. 本文首先在 CIFAR-10 数据集上对提出方法进行实验评估, 与现有 3 个数据集水印工作和复用的两个干净标签下的数据投毒攻击工作相比, 本文提出的数据集水印算法有效性最佳. 其次, 本文开展了消融实验验证了算法的风格优化模块的必要性, 分析不同风格强度和扰动比例对算法性能的影响, 另外在 CIFAR-100 和 ImageNet 子集上的实验结果表明算法可应用于更复杂的数据集上.

总体而言, 与之前的数据集水印算法相比本文主要有以下两方面优势.

(1) 本文在干净标签与黑盒验证限制下, 提出了基于风格的数据集水印算法. 在同样的较小扰动比例下, 实验表明本文所提的算法在更严格的限制下有更强的有效性与相当的功能不变性.

(2) 本文提出的基于风格的数据集水印算法在水印植入与验证阶段都不依赖于具体模型, 使得算法在验证不同结构的可疑模型时更可靠.

本文组织如下: 第 1 节介绍数据集水印任务的相关工作; 第 2 节给出数据集水印的形式化定义与要求; 第 3 节介绍基于风格的数据集水印算法; 第 4 节介绍实验设置以及实验结果; 第 5 节进行总结与展望.

1 相关工作

1.1 模型水印

数字水印 (digital watermarking) 研究如何在图像, 视频, 语音等数字信号中添加水印信息以期声明其所有权[17–19]. 受数字水印启发, 研究者提出模型水印概念来保护深度神经网络模型的知识产权. 近期工作设计了不同的水印信息, 例如随机的 0-1 比特串[13,20–23], 随机样本集合[12,24,25] 等, 并将这类水印信息嵌入到模型的特定层参数[20–22], 特定层激活值的概率密度函数[13,23] 又或者模型对于预先选定输入样本集合的预测结果[12,24,25] 中. 然而模型水印任务的目标是保护一个特定的深度神经网络模型, 而非本文所关注的整个数

据集, 因此仍然需要数据集水印工作单独地对数据集的知识产权进行保护.

1.2 数据集水印

不同于模型水印, 仅有少数工作关注深度学习数据集的知识产权保护. 文献 [26–28] 利用模型对于参与训练样本和非训练集样本的预测行为之间的差异做数据集推断 (dataset inference), 判断可疑模型是否在私有数据集上训练. 但这类工作不预先对数据集做任何改动, 而本文主要关注在数据集发布前预先植入水印的数据集水印任务. Li 等人[9] 首次提出数据集水印的概念, 并设计了 BEDW 算法. BEDW 使用后门攻击[14] 对数据集投毒植入后门, 后续在验证阶段依据可疑模型是否存在后门来判断模型是否在水印数据集上训练获得. 但 BEDW 在水印植入阶段不仅需要扰动原数据集的图像内容还需要修改对应标签, 引入了样本语义内容与标签的不一致性, 导致数据集水印缺乏隐蔽性, 容易被检测过滤[29]. Sablayrolles 等人[10] 提出放射性数据集 (radioactive data), 通过对训练集样本添加指定方向扰动向模型分类层参数植入水印, 但在水印验证阶段要求验证方拥有可疑模型的白盒访问权限, 即能直接获取可疑模型具体参数等信息, 而当前商业化模型所提供的预测 API 接口大多只返回模型预测类别或者各类别预测置信度. 因此, 验证方拥有可疑模型的白盒访问权限假设过强, 不符实际情况. Li 等人[11] 提出可支持黑盒验证 UBW 算法, 其中为提高水印隐蔽性, 进一步设计了 UBW-C 干净标签版本, 在水印植入过程中不扰动原样本的标签信息. 但 UBW-C 算法需要引入代理模型生成扰动, 导致 UBW-C 添加的水印在使用了不同模型结构的可疑模型中效果相对较差. 本文在此基础上提出在干净标签和扰动比例较小限制下仍有效的且可以支持黑盒验证的数据集水印算法.

1.3 数据投毒攻击

数据投毒攻击 (data poisoning attacks), 包括后门攻击 (backdoor attacks), 是期望通过扰动训练数据集以使在此数据集上训练得到的模型在推理应用阶段有特殊预测行为, 例如模型将指定样本分类出错[30] 或分类为特定类别[14]. 近期, 研究者提出复用数据投毒攻击于防御用途[9,12]. 同时, 为提高数据投毒攻击的隐蔽性, 研究者提出在干净标签限制下或使用特征碰撞[30–34] 设计攻击算法或将问题形式化为双层优化问题, 通过部分展开[35], 梯度匹配[36,37], 一阶梯度近似[38] 等方式求解. 本文受此启发在数据集水印任务上提出满足干净标签限制的, 更为隐秘的水印算法. 但正如 Schwarzschild 等人[39] 所述, 当前的干净标签后门攻击工作在脱离对受害者模型 (victim model) 的假设后并不能达到理想的攻击效果, 因此本文不考虑直接复用当前已有的干净标签下的数据投毒攻击工作, 而是创新性地提出基于风格特征的数据集水印算法.

2 数据集水印

同已有数据集水印工作[9–11] 一样, 本文也先关注图像分类数据集的知识产权保护, 对其他领域的数据集水印扩展留待之后的工作探索. 考虑从联合分布$P_{\mathcal{X},\mathcal{Y}}$中采样得到的分类数据集$\mathcal{D}_{\text{ori}}=\{(x_i,y_i)\}_{i=1}^{N}$, 其中$x_i\in\mathcal{X}\subseteq\mathbb{R}^d$, $y_i\in\mathcal{Y}=\{0,\cdots,L-1\}$, N为样本数量. 数据集水印算法是预先在原数据集$\mathcal{D}_{\text{ori}}$中植入水印得到嵌入水印后的数据集$\mathcal{D}_{\text{wm}}$. 之后, 当有可疑模型出现时, 算法需提供可以判断该模型是否在嵌入水印后的数据集上训练的能力. 形式化地, 数据集水印算法主要包含以下两个模块:

(1) 水印嵌入: $\mathcal{D}_{\text{wm}},m_v\leftarrow\textbf{Embed}(\mathcal{D}_{\text{ori}},m)$嵌入算法, 即给定原数据集$\mathcal{D}_{\text{ori}}$和数据集持有方独有的隐秘信息$m$, 输出嵌入水印后的数据集$\mathcal{D}_{\text{wm}}$和之后用于验证的信息$m_v$.

(2) 水印验证: $b_v\leftarrow\textbf{Verify}(\mathcal{F},m_v)$验证算法, 即给定一个可疑模型$\mathcal{F}$和验证信息$m_v$, 输出$b_v\in\{0,1\}$表示模型是否含有水印.

方便起见, 记$\mathcal{F}\leftarrow\textbf{Train}(\mathcal{D},\mathcal{A})$表示模型$\mathcal{F}$的训练算法$\mathcal{A}$涉及数据集$\mathcal{D}$. 则给定含水印数据集$\mathcal{D}_{\text{wm}}$, 本文称$\mathcal{F}_+$为正例模型当且仅当$\mathcal{F}_+\leftarrow\textbf{Train}(\mathcal{D}_{\text{wm}},\mathcal{A})$, 即其训练过程使用了水印数据, 否则该模型被称作负例模型, 记为$\mathcal{F}_-$.

数据集水印方案要求满足以下两点.

(1) 有效性 (effectiveness): 能准确检测出正例与负例模型, 形式化地, 对任意$\mathcal{A}$和植入的水印与验证信息$\mathcal{D}_{\text{wm}}$, $m_v\leftarrow\textbf{Embed}(\mathcal{D}_{\text{ori}},m)$ 有:

$$\Pr_{\mathcal{F}_+\leftarrow\textbf{Train}(\mathcal{D}_{\text{wm}},\mathcal{A})}[\textbf{Verify}(\mathcal{F}_+,m_v)=1]=1 \tag{1}$$

$$\Pr_{\mathcal{F}_-\leftarrow\textbf{Train}(\mathcal{D}_{\text{ori}},\mathcal{A})}[\textbf{Verify}(\mathcal{F}_-,m_v)=0]=1 \tag{2}$$

(2) 功能不变性 (fidelity): 使用相同结构和训练方

式的正例与负例模型性能应当相似, 形式化地, 对任意 $\mathcal{F}_+ \leftarrow \textbf{Train}(\mathcal{D}_{\text{wm}}, \mathcal{A})$, $\mathcal{F}_- \leftarrow \textbf{Train}(\mathcal{D}_{\text{ori}}, \mathcal{A})$, 有:

$$\Pr_{(x,y)\sim\mathcal{D}_{\text{test}}}[\mathcal{F}_+(x)=y] \approx \Pr_{(x,y)\sim\mathcal{D}_{\text{test}}}[\mathcal{F}_-(x)=y] \tag{3}$$

其中, $\mathcal{D}_{\text{test}}$为从$P_{\mathcal{X},\mathcal{Y}}$中采样的干净测试样本集合.

3 基于风格的数据集水印算法

正如第 2 节中给出的关于数据集水印基本定义, 本文提出的基于风格的数据集水印算法主要包括数据集水印嵌入和数据集水印验证两个模块.

3.1 数据集水印嵌入

如图 1 左半部分所示, 本文将**Embed**($\mathcal{D}_{\text{ori}}, m$)中的水印信息$m$设计为一个数据集持有方指定的风格图像$x_s \notin \mathcal{X}$, 特定的风格迁移模型$G$以及目标类别$y_t \in \mathcal{Y}$. 值得注意的是, 上述 3 类信息的选取是不公开的, 由数据集持有方自定义不对外公布.

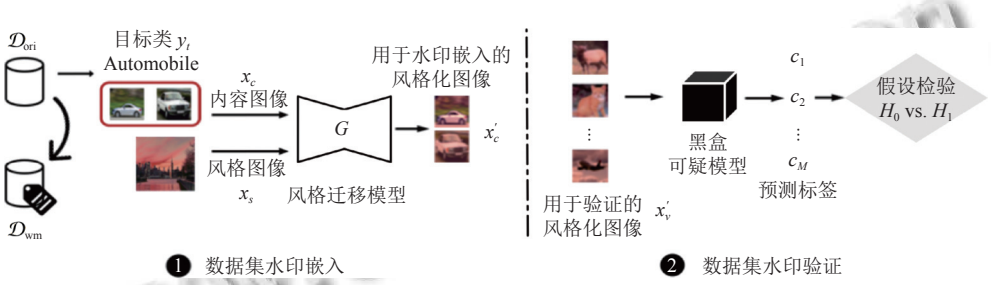


图 1 基于风格的数据集水印算法示意图

为了能在满足干净标签限制下嵌入水印, 本文提出在独立于图像语义内容与标签的特征, 即图像风格维度中嵌入水印. 具体地, 先随机从目标类图像中选取$n = \lfloor r_p \cdot N_{y_t} \rfloor$张图像作为内容图像, 记做$X_c = \{x_{c_i}\}_{i=1}^n$, 其中$r_p$为扰动比例, N_{y_t}为原数据集目标类样本数量. 给定内容图像x_{c_i}与风格图像x_s, 本文利用风格迁移模型G生成迁移后的具有x_{c_i}的内容与x_s的风格的图像x'_{c_i}. 之后, 将原数据集中的x_{c_i}用风格迁移后图像x'_{c_i}代替, 同时不对标签做任何修改, 保持其原有的真实标签不变. 如此, 替换原数据集$\mathcal{D}_{\text{ori}}$中选择的n张图像X_c后即可得到嵌入水印后的数据集$\mathcal{D}_{\text{wm}}$. 另一方面, 在**Embed**($\mathcal{D}_{\text{ori}}, m$)嵌入水印算法中还需生成后续用于验证的信息$m_v$. 具体地, 先随机选取$M$张干净测试集中其他类样本$\{x_{v_i}\}_{i=1}^M$作为内容图像, 并以同样的方式利用风格迁移模型获取风格化图像$\{x'_{v_i}\}_{i=1}^M$. 最终, 验证信息m_v为$\{(x'_{v_i}, y_t)\}_{i=1}^M$.

风格优化模块: 考虑到风格图像和风格迁移模型的选取是随机的, 这可能导致在部分风格图像和模型下嵌入水印相关的图像是相对容易被察觉. 为使风格化后的图像变得更不易察觉, 一种简单的方法是为每个数据集精心挑选合适的风格图像或对风格迁移模型精心调整超参数. 但这样的方式即费时又费力, 因此本文提出直接对随机选取的风格图像进行优化. 考虑到在图像风格迁移任务中通常使用结构相似性指标 (structure similarity index, *SSIM*)[40] 作为图像语义内容保留程度的衡量指标, 因此本文使用结构相似性指标作为优化目标. 具体地, 通过优化风格图像来最大化原始图像x_{c_i}和风格化后图像x'_{c_i}之间的 *SSIM*, 即:

$$x_s^* = \arg\min_{x_s} \mathbb{E}_{x_{c_i}\sim X_c} - SSIM\left(x_{c_i}, x'_{c_i}\right) \tag{4}$$

在本文中, 风格迁移模型选用任意风格迁移算法 (arbitrary style transfer algorithms)[15,16], 即模型以风格图像和内容图像为输入, 可以提取任意风格图像的风格信息并将其添加到内容图像上, 输出风格化后的内容图像. 这一过程可表示为$x = G(c, s)$, 其中, c和s分别表示内容和风格图像, x为风格化后图像. 依据不同的风格迁移机制, 任意风格迁移模型大致可分为两类:

(1) 基于优化的风格迁移模型 (optimization-based style transfer model): $G(c, s)$为计算得到预先定义优化目标的一个迭代式的优化过程, 即$x^* = G(c, s) = \arg\min_x \mathcal{L}_{\text{st}}(x, c, s)$. 通常, 优化目标$\mathcal{L}_{\text{st}}$是内容损失和风格损失的组合, 自 Gatys 等人[15] 首先提出基于优化的风格迁移方式后, 后续工作沿着如何设计更合适的内容损失或者风格损失发展[41].

(2) 前馈式风格迁移模型 (feed-forward style transfer model): $G(c, s)$为精心设计的映射. 例如 Yoo 等人[16] 使用统计特征直接混合风格与内容特征, Svoboda 等人[42] 则是利用生成对抗网络 (generative adversarial networks, GAN) 生成风格化图像.

对于前馈式风格迁移模型, 可以直接使用基于梯度的优化器优化式 (4), 但对于基于优化的风格迁移模型, 由于风格化图像$x'_{c_i}=G(x_{c_i},x_s)$本身牵涉到优化, 因此最大化原始图像x_{c_i}和风格化后图像x'_{c_i}之间的 *SSIM* 问题转变为如下的双层优化问题:

$$x_s^*=\arg\min_{x_s}\mathbb{E}_{x_{c_i}\sim X_c}-SSIM(x_{c_i},x'_{c_i}) \tag{5}$$

$$\text{s.t. } x'_{c_i}=\arg\min_{x''_{c_i}}\mathcal{L}_{\text{st}}(x''_{c_i},x_{c_i},x_s),\ i=1,\cdots,n \tag{6}$$

对于式 (6) 的内层循环, 通常需要几百轮的迭代优化, 因此直接求解整个双层优化问题是不切实际的. 本文使用部分展开技术[43], 对每轮式 (5) 外层优化的迭代, 仅优化K步来近似估计内层优化目标, 在具体实现中设置$K=2$.

3.2 数据集水印验证

给定一个仅有黑盒访问权限的可疑模型$\mathcal{F}$和验证信息m_v, **Verify**$(\mathcal{F},m_v)$算法需要确定模型是否在含有水印的数据集上训练. 为提供具有一定置信度水平的所有权声明, 本文用假设检验实现**Verify**$(\mathcal{F},m_v)$算法.

如图 1 右半部分所示, 首先使用验证信息m_v中的风格化图像$\{x'_{v_i}\}_{i=1}^M$查询可疑模型获取模型预测标签$\{c_i\}_{i=1}^M$, 之后与验证信息中的目标标签y_t相比获得二元值$\{b_i\}_{i=1}^M$, 其中, $b_i=1(c_i=y_t)$, $1(\cdot)$为指示函数. 由于所有的风格化图像都已嵌入风格信息且是独立同分布的, 因此可将收集到的观测值$\{b_i\}_{i=1}^M$也视作独立同分布变量, 且有$k\triangleq\sum_{i=1}^M b_i\sim B(M,p)$, 其中, $B(\cdot,\cdot)$为二项分布, p为可疑模型错分风格化图像为目标类的概率. 基于此, 本文定义零假设$\mathcal{H}_0$为可疑模型在无水印数据集上训练, 即模型将风格化图像预测为目标类的近似概率为$p=(1-ACC)/(L-1)$, 其中ACC为模型对正常测试集的准确率, L为类别数目. 当且仅当零假设被拒绝, 算法才声称可疑模型$\mathcal{F}$是在水印数据集上训练得到的. 使用零假设中的近似概率和二项分布假设, 可通过式 (7)–式 (9) 计算 p 值 (p-value):

$$p_1=\Pr(X\geqslant k)=\sum_{i=k}^{M}\binom{n}{i}p^i(1-p)^{M-i} \tag{7}$$

$$p_2=\Pr(X\leqslant k)=\sum_{i=0}^{k}\binom{n}{i}p^i(1-p)^{M-i} \tag{8}$$

$$p\text{-value}=2\times\min(p_1,p_2) \tag{9}$$

p 值给出了当零假设为真时, 比所得到的样本观察值更极端的结果出现的概率. 在算法的具体实现中, **Verify**$(\mathcal{F},m_v)$输出最终结果$b_v=1$ (p-value $<\tau$), 其中, τ为置信度阈值, 在具体实现中设置$\tau=0.05$. $b_v=1$意味着零假设$\mathcal{H}_0$被拒绝, 即模型$\mathcal{F}$是在水印数据集上训练的, 而$b_v=0$则意味着零假设$\mathcal{H}_0$被接受, 模型$\mathcal{F}$并未在水印数据集上训练.

4 实验分析

4.1 实验设置

(1) 数据集与模型. 本文主要在 CIFAR-10 数据集[44]上进行实验, 该数据集包含了 60 000 张来自 10 个不同类别的图像, 每张图像大小为$3\times32\times32$, 其中训练集中每类有 5 000 个样本, 测试集中每类也有 1 000 个样本. 关于正负例可疑模型所使用结构, 本文选用 3 个主流 DNN 模型结构: ResNet18[3], VGG16[4] 和 EfficientNet[45], 并依照 CIFAR-10 的数据大小对模型的卷积核大小做适当调整. 本文使用这 3 个主流 DNN 模型分别在原数据集和添加水印后的数据集上训练得到负例和正例可疑模型用于后续测试评估. 此外, 在第 4.3 节的消融实验评估中使用了 CIFAR-100[44] 和 ImageNet[1] 的子集来验证本文提出的基于风格数据集水印算法在数据集类别和分辨率大小不同时的性能.

(2) 基线模型. 本文选取如下 3 种现有数据集水印算法作为基线模型.

BEDW[9]: 复用传统有毒标签后门攻击 BadNet[14], 从数据集其他类别随机挑选一定比例图像添加触发器, 即右下角3×3的白色块, 并将样本标签修改为目标类别. 验证时查询模型对于干净样本与对应添加触发器的样本在目标类上的置信度, 之后通过配对样本 T 测试计算 p 值.

Radioactive data (RD)[10]: 对目标类别中部分样本的特征添加数据集所有者指定的方向扰动, 通过逆向优化得到输入空间图像上具体的扰动形式, 并将该扰动添加到原始数据集中以植入水印. 验证时, 查询含特定方向扰动的样本特征, 并计算该特征与指定方向的余弦相似度, 将其作为观测量通过 Fisher 法计算结合 p 值.

UBW-C[11]: 在干净标签限制下, 将数据集水印任务形式化为双层优化问题生成对部分样本的扰动, 其中外层的优化目标为特定类别样本在添加了特殊图案

后正例可疑模型将会预测错误. 验证时, 查询模型对于干净样本与对应地添加了特殊图案的样本上在真实标签类上的置信度, 之后同样通过配对样本 T 测试计算 p 值.

此外, 考虑复用已有的干净标签下的后门攻击工作到数据集水印任务中, 本文选取经典的具有代表性的工作 Label Consistent[33] 和当前干净标签下后门攻击任务的前沿工作 Sleeper Agent[37] 作为本文方法的基线模型.

Label Consistent[33]: 先对原始数据集中目标类的部分图像使用 PGD 算法生成对抗样本, 再在对抗样本上添加后门触发器后替换原始数据集中的干净样本来植入后门, 即本文复用后的数据集水印信息. 验证时, 使用本文提出的验证算法计算 p 值.

Sleeper Agent[37]: 将干净标签下的后门攻击形式化为双层优化问题, 内层优化是在投毒训练集上针对模型的正常训练过程, 外层优化是针对添加的投毒样本, 其优化目标即为后门攻击目标. 之后使用梯度匹配的方式进行求解得到应添加的投毒样本. 验证时, 使用本文提出的验证算法计算 p 值.

(3) 评估指标. 本文分别使用验证阶段数据集水印算法对正负例可疑模型计算的 p 值和在添加水印后的数据集上训练得到正例可疑模型的准确率 (ACC) 来评估数据集水印算法的有效性和功能不变性. 具体地, 验证时对正例可疑模型的 p 值越低, 负例可疑模型的 p 值越高, 意味着该数据集水印算法越有效. 若含水印的正例模型 ACC 接近相同结构的模型在无水印数据集上训练的 ACC, 则意味着该水印算法具有功能不变性. 另外, 为突显 p 值的变化, 第 4.3 节消融实验中使用 p 值的负对数 ($-\log_{10}p$) 评估算法的有效性.

(4) 实现细节. 在本文提出的基于风格的数据集水印算法的具体实现中, 设置 CIFAR-10 数据集的目标类 y_t为“汽车 (automobile)”; 风格迁移模型G分别选用经典的基于优化的和前馈式风格迁移模型: Gatys 和 WCT2, 并且分别将使用了这两个风格迁移模型的数据集水印算法记做 Gatys w/ refine 和 WCT2 w/ refine, 其中 w/ refine 表示使用了风格优化模块对风格图像进行优化, 另外记 w/o refine 表示未使用风格优化模块; 风格图像x_s则是先随机选自域外图像, 再经过特定迁移模型和目标类优化后获得.

对于基线模型的实现, 本文均采用原文献提供的官方开源代码. 特殊地, RD, UBW-C, Label Consistent 和 Sleeper Agent 算法在数据集水印嵌入阶段都需要引入特定代理模型生成水印相关样本, 在本文的具体实现中, 设置的代理模型结构为 ResNet18 模型, 即这 4 种算法利用 ResNet18 模型对数据集添加水印, 后使用该水印数据集在 3 种模型结构上进行测试评估.

给定已添加水印的数据集或原始数据集, 本文使用同样的训练方式得到正负例模型. 具体的训练方式如下: 设置训练轮数为 200 轮, 批大小为 32, 梯度下降的优化器为 SGD 优化器, 其中设置学习率为 0.1, 动量为 0.9, 权重衰减为 5E–4, 调度器为 CosineAnnealingLR 调度器, 其中最大迭代次数为 200.

另外, 为保证评估的公平性, 所有基线模型的目标类别y_t, 水印嵌入中数据扰动比例r_p以及模型训练方式等都与本文提出方法的实现保持一致.

4.2 数据集水印算法性能

本节首先从有效性与功能不变性两个角度评估本文提出算法的性能. 表 1 展示了不同数据集水印算法验证时对正负例模型所得的 p 值, 以及模型在已添加水印的数据集上训练得到模型的 ACC.

先关注各水印算法在正例模型上的 p 值, 由于本文设定的在水印嵌入阶段可对数据集添加的扰动比例较小, 对于 CIFAR-10 数据集仅可修改 500 个样本, 导致 RD, UBW-C 和 Label Consistent 方法的表现都相对较差. 当验证置信度阈值$\tau = 0.05$时, RD 对 3 种模型结构的正例模型都判断错误, 而 UBW-C 和 Label Consistent 方法则是分别对 VGG16 和 EfficientNet 结构下的正例模型判断出错. 除本文提出方法和 BEDW 之外, 其他基线模型在水印嵌入阶段都需引入代理模型, 在具体实现中是以 ResNet18 模型作为代理模型生成的水印数据集. 这导致了这些方法在使用了相同模型结构的正例模型中可得到相对较小的 p 值, 而在其他结构的正例模型所得的 p 值则较大, 进而可能出现错误的判断. 另一方面, 在嵌入阶段未引入额外代理模型的 BEDW 和本文提出的方法则对不同模型结构的正例模型都可取得相近的且较低的 p 值. 这也反映出了数据集水印任务中若嵌入阶段涉及具体代理模型, 则在其他模型结构上检测的准确率不高, 即本身数据集水印算法的可迁移性较差.

再观察负例模型上各个水印算法计算的 p 值, BEDW 和 Sleeper Agent 算法对部分负例模型的 p 值相对较

低, 且有可能将负例模型误判为是在水印数据集上训练得到的正例模型. 而本文提出的算法所计算的 p 值不论是使用基于优化的风格迁移模型 Gatys 还是前馈式模型 WCT2 都相对较高, 可以认为当前负例模型与水印数据集无关. 结合正负例模型的 p 值, 本文提出的基于风格特征的数据集水印算法相比其他算法可以更好地区分不同模型结构上的正负例模型, 并且在较小的扰动比例下取得不错的有效性.

表 1 不同数据集水印算法在 CIFAR-10 数据集上的实验结果

算法	ResNet18			VGG16			EfficientNet		
	p值		ACC (%)	p值		ACC (%)	p值		ACC (%)
	$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$		$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$		$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$	
无水印	—	—	95.30	—	—	94.01	—	—	91.02
BEDW[9]	10^{-48}	0.3354	94.49	10^{-51}	0.0563	93.30	10^{-54}	0.3079	90.13
RD[10]	0.4674	0.7525	94.53	0.4689	0.4462	93.04	0.8704	0.5315	91.69
UBW-C[11]	0.0027	0.3921	94.60	0.0706	0.1620	93.12	0.0301	0.1014	90.28
Label Consistent[33]	10^{-109}	0.0966	95.19	0.0380	0.1435	93.38	0.1053	0.6331	89.80
Sleeper Agent[37]	10^{-75}	0.0002	94.76	10^{-11}	10^{-7}	93.42	10^{-5}	0.0789	89.04
Gatys w/ refine	10^{-206}	0.4076	95.25	10^{-190}	0.5128	93.65	10^{-173}	0.6331	90.31
Gatys w/o refine	10^{-227}	0.0966	95.13	10^{-213}	0.1435	93.81	10^{-195}	0.6331	90.64
WCT2 w/ refine	10^{-95}	0.5924	95.28	10^{-170}	0.5128	93.41	10^{-79}	0.6331	90.56
WCT2 w/o refine	10^{-166}	0.4076	95.04	10^{-185}	0.5128	93.32	10^{-174}	0.3668	90.40

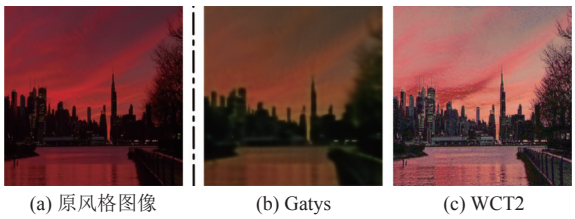
对于数据集水印算法的功能不变性, 表 1 结果显示, 本文提出的数据集水印算法, 现有的 3 个数据集水印工作以及复用的后门攻击工作基本都未破坏原始数据集本身的性能. 具体地, 5 个基线模型在 ResNet18, VGG16 和 EfficientNet 这 3 个模型上 ACC 下降平均幅度分别为 0.59%, 0.76% 和 0.83%, 而本文提出方法 (Gatys w/ refine 和 WCT2 w/ refine) 的 ACC 下降平均幅度分别为 0.04%, 0.48% 和 0.59%. 由此可见本文提出的基于风格的数据集水印算法在 3 种模型结构上都具有更好的功能不变性.

4.3 消融实验

4.3.1 风格优化模块影响

本节从定性和定量两个角度分析第 3.1 节中提出的风格优化模块在数据集水印算法中的影响.

定性地, 图 2 给出了优化前和针对不同风格迁移模型优化后所使用的风格图像. 如图所示, 最初随机选择的风格图像中大面积的红色经过优化后被削弱了, 这使得以优化后的图像作为风格图像得到的风格化模型在目标类中更为隐蔽. 此外, 对于不同的风格迁移模型, 风格图像的优化方向各不相同, 例如, 对于 Gatys 风格迁移模型的图像优化部分并不关注原风格图像中的黑色建筑部分, 而 WCT2 方法却是将这部分亮度调高. 定量地, 本文计算了以优化前和优化后图像作为风格图像生成的风格化图像与原图之间的 *SSIM* 分数, 表 2 结果显示风格优化模块能有效提高了风格化图像与原图之间的 *SSIM* 分数. 另外, 除分析水印本身在使用风格优化模块前后的区别之外, 本文也评估了使用原图作为风格图像时的水印性能. 表 1 中的倒数第 3 行和最后一行实验结果表明了风格优化模块在提升水印的隐蔽性时, 并不会对水印算法的有效性和功能不变性有太大的负面影响. 即使在最糟糕的场景, 即 EfficientNet 模型下使用 WCT2 方法进行风格迁移, 前置使用风格优化模块的数据集水印算法对负例模型计算得到的 p 值仍然有10^{-79}, 意味着可以在极高的置信度下声称这一负例模型是在水印数据集上训练得到的.



(a) 原风格图像 (b) Gatys (c) WCT2

图 2 基于风格的数据集水印算法所使用的风格图像

4.3.2 风格强度影响

直观地, 本文提出的基于风格的数据集水印算法会随嵌入时的风格强度增大而更具有效性, 但对功能不变性的影响未知, 本节开展经验性实验探索风格强

度对本文提出算法的影响. 在图像的风格迁移任务中, 通常使用格拉姆损失 (gram loss) 来评估模型生成的风格化图像的风格强度[16]. 具体地, 格拉姆损失是生成的风格化图像和风格图像格拉姆矩阵 (gram matrix) 之间的差异, 其中格拉姆矩阵是图像各通道特征之间的偏心协方差矩阵, 计算了各个特征之间的相关性, 被视为可反映出整个图像的风格特征. 本文沿用这一指标并计算 4 种水印所添加的风格强度μ. 依照风格强度递增排序, 4 种水印顺序为: WCT2 w/ refine < Gatys w/ refine < WCT2 w/o refine < Gatys w/o refine, 分别记作$\mu = 1, 2, 3, 4$.

表 2 不同数据集水印算法的实验结果

算法	CIFAR-10
Gatys w/o refine	0.431 4
Gatys w/ refine	0.540 6 (+25%)
WCT2 w/o refine	0.513 8
WCT2 w/ refine	0.889 6 (+73%)

图 3(a) 展示了使用不同风格强度水印时本文提出算法对正例模型的 p 值负对数, 该值越高意味着算法可以以更高的置信度声称正例模型是在水印数据集上训练获得. 图 3(b) 相应地给出了在嵌入了不同风格强度的水印数据集上训练得到的正例模型在测试集上的准确率. 图 3 的实验结果经验性地论证了当水印使用的风格强度越大时, 基于风格特征的数据集水印算法能更置信地声明版权, 且对于数据集本身性能的影响并不会随着风格强度的增大而有明显变化.

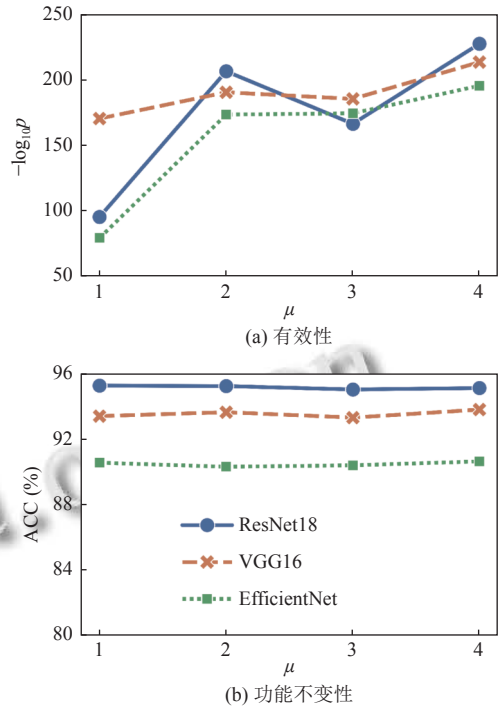


图 3 不同风格强度对数据集水印算法有效性和功能不变性的影响

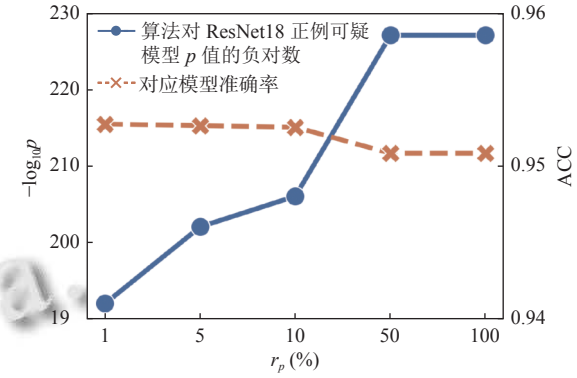


图 4 不同扰动比例对基于风格的数据集水印算法的影响

4.3.3 扰动比例影响

第 4.2 节的实验结果展示了在扰动比例r_p设置为 10% 情况下各算法性能, 本节评估基于风格的数据集水印算法在不同扰动比例下的性能. 图 4 给出了本文提出的算法在扰动比例r_p分别设置为 1%, 5%, 10%, 50% 和 100% 时, 正例模型的 p 值与准确率. 结果显示当扰动比例逐渐增大, 算法对正例模型判断为含水印模型的置信度越高, 而另一方面对数据集本身的功能有轻微影响, 模型准确率有所降低, 但其影响仍然较低, 在 1% 以内.

4.3.4 不同数据集上的表现

为进一步验证本文提出的基于风格的数据集水印算法可以应用于更复杂的数据集中, 本文在含有更多类别的 CIFAR-100 数据集和图像分辨率更高的 ImageNet 的 10 分类子集 (记做 ImageNet-10) 上与现有 3 种数据集水印算法进行对比实验评估.

表 3 的结果显示, 与现有的数据集水印算法相比, 本文提出的算法不论是在类别数更多的 CIFAR-100 数据集上还是图像分辨率更高的 ImageNet-10 数据集上都有更好的表现. 具体地, 对于水印算法的有效性, 本文提出算法 (Gatys w/ refine) 在正例模型上的 p 值都较小, 而负例模型上的 p 值则都大于设定的置信度阈值$\tau = 0.05$, 因此算法可正确判断正负例可疑模型. 同时, 注意到算法在 ImageNet-10 数据集上对于负例模型的 p 值相对偏低, 这主要是 ImageNet-10 数据集上训练得到的模型准确率较高, 导致当负例模型将个别用于

验证的风格化图像误分类时就会得到较低的 p 值. 此时, 对于模型分类准确率较高的场景, 本文提出算法可选择例如10^{-10}等更小的值作为置信度阈值τ. 反观已有数据集水印算法, BEDW 算法在含有更多类别的 CIFAR-100 数据集上表现变差, RD 和 UBW-C 算法由于嵌入时引入代理模型且限制了可扰动图像较少在 CIFAR-100 和 ImageNet-10 数据集上性能同样较差. 对于水印算法的功能不变性, 实验结果表明各种算法添加的水印基本不影响数据集的正常性能, 在更复杂的数据集上仍然具有功能不变性. 其中本文提出算法的功能不变性最佳, 模型 ACC 的下降幅度都在 0.5% 以内, 且在部分场景添加水印后模型准确率有所提升.

表 3 不同数据集水印算法在 CIFAR-100 和 ImageNet-10 数据集上的实验结果

数据集	水印算法	ResNet18			VGG16			EfficientNet		
		p值		ACC (%)	p值		ACC (%)	p值		ACC (%)
		$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$		$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$		$\mathcal{F}_+\downarrow$	$\mathcal{F}_-\uparrow$	
CIFAR-100	无水印	—	—	78.16	—	—	74.53	—	—	69.15
	BEDW	0.1188	0.2211	77.35	0.1809	0.0701	73.22	0.0843	0.2429	67.43
	RD	0.1986	0.4392	77.86	0.4645	0.6391	73.07	0.6122	0.7232	69.68
	UBW-C	10^{-46}	10^{-12}	76.99	10^{-41}	10^{-10}	73.74	10^{-41}	10^{-11}	67.91
	Gatys w/ refine	10^{-209}	0.8018	78.31	10^{-181}	0.7729	74.89	10^{-137}	0.2681	70.07
ImageNet-10	无水印	—	—	96.06	—	—	96.32	—	—	95.08
	BEDW	10^{-33}	10^{-8}	95.40	10^{-34}	10^{-6}	95.80	10^{-22}	0.0004	94.66
	RD	0.2759	0.7938	95.60	0.7427	0.8289	95.66	0.6480	0.8290	94.06
	UBW-C	0.0298	0.1657	95.46	0.4203	0.3078	95.82	0.3069	0.4176	94.62
	Gatys w/ refine	10^{-164}	0.0716	95.88	10^{-114}	0.0636	96.06	10^{-145}	0.1043	95.28

5 结论与展望

针对现有数据集水印工作的不足, 即无法在满足干净标签和扰动比例较小限制下提供有效的且可以支持黑盒验证的水印算法, 本文探究独立于图像语义内容和标签的风格特征, 提出了基于风格的数据集水印算法. 通过与现有的 3 种数据集水印方案和复用的两种干净标签下的数据投毒攻击工作进行实验对比, 实验结果显示本文提出算法在数据集水印有效性上的表现最佳, 不论是在何种模型结构上, 都可以以较高的置信度正确判断正负例可疑模型. 在数据集水印功能不变性上的表现与其他基准模型相近, 都对原始数据集本身功能的影响较小.

本文主要针对图像领域上的有监督数据集提出了基于风格的数据集水印算法, 为相关数据集提供知识产权保护. 在未来工作中, 一方面期望将基于风格的数据集水印算法扩展到其他领域中, 例如文本, 图等领域, 提出更通用的数据集水印算法. 另一方面期望设计针对无监督数据集的水印算法, 通过在无标签数据上探索植入的风格对模型产生的影响, 进而重新设计相关验证方式.

参考文献

1 Russakovsky O, Deng J, Su H, *et al*. ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 2015, 115(3): 211–252. [doi: 10.1007/s11263-015-0816-y]

2 Deng J, Dong W, Socher R, *et al*. ImageNet: A large-scale hierarchical image database. Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition. Miami: IEEE, 2009. 248–255.

3 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778.

4 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. Proceedings of the 3rd International Conference on Learning Representations. San Diego: ICLR, 2015.

5 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7132–7141.

6 Jensen MB, Philipsen MP, Møgelmose A, *et al*. Vision for looking at traffic lights: Issues, survey, and perspectives. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(7): 1800–1815. [doi: 10.1109/TITS.2015.2509509]

7 Krause J, Stark M, Deng J, *et al*. 3D object representations for fine-grained categorization. Proceedings of the 2013 IEEE International Conference on Computer Vision

Workshops. Sydney: IEEE, 2013. 554–561.
8 Philipsen MP, Jensen MB, Møgelmose A, *et al*. Traffic light detection: A learning algorithm and evaluations on challenging dataset. Proceedings of the 18th IEEE International Conference on Intelligent Transportation Systems. Gran Canaria: IEEE, 2015. 2341–2345.
9 Li YM, Zhang ZQ, Bai JW, *et al*. Open-sourced dataset protection via backdoor watermarking. arXiv:2010.05821, 2020.
10 Sablayrolles A, Douze M, Schmid C, *et al*. Radioactive data: Tracing through training. International Conference on Machine Learning. Proceedings of the 37th International Conference on Machine Learning. ICML, 2020. 8326–8335.
11 Li YM, Bai Y, Jiang Y, *et al*. Untargeted backdoor watermark: Towards harmless and stealthy dataset copyright protection. Proceedings of the 36th Conference on Neural Information Processing Systems. New Orleans: OpenReview.net, 2022. 1–13.
12 Adi Y, Baum C, Cisse M, *et al*. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. Proceedings of the 27th USENIX Conference on Security Symposium. Baltimore: USENIX Association, 2018. 1615–1631.
13 Lim JH, Chan CS, Ng KW, *et al*. Protect, show, attend and tell: Empowering image captioning models with ownership protection. Pattern Recognition, 2022, 122: 108285. [doi: 10.1016/j.patcog.2021.108285]
14 Gu TY, Dolan-Gavitt B, Garg S. Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv:1708.06733, 2019.
15 Gatys LA, Ecker AS, Bethge M. A neural algorithm of artistic style. Journal of Vision, 2016, 16(12): 326. [doi: 10.1167/16.12.326]
16 Yoo J, Uh Y, Chun S, *et al*. Photorealistic style transfer via wavelet transforms. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 9035–9044.
17 Lee TY, Lin SD. Dual watermark for image tamper detection and recovery. Pattern Recognition, 2008, 41(11): 3497–3506. [doi: 10.1016/j.patcog.2008.05.003]
18 Abdelnabi S, Fritz M. Adversarial watermarking Transformer: Towards tracing text provenance with data hiding. Proceedings of the 2021 IEEE Symposium on Security and Privacy. San Francisco: IEEE, 2021. 121–140.
19 Evsutin O, Dzhanashia K. Watermarking schemes for digital images: Robustness overview. Signal Processing:Image Communication, 2022, 100: 116523. [doi: 10.1016/j.image.2021.116523]
20 Uchida Y, Nagai Y, Sakazawa S, *et al*. Embedding watermarks into deep neural networks. Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. Bucharest: ACM, 2017.
21 Liu HW, Weng ZY, Zhu YS. Watermarking deep neural networks with greedy residuals. Proceedings of the 38th International Conference on Machine Learning. PMLR, 2021. 6978–6988.
22 Ong DS, Chan CS, Ng KW, *et al*. Protecting intellectual property of generative adversarial networks from ambiguity attacks. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 3629–3638.
23 Rouhani BD, Chen HL, Koushanfar F. DeepSigns: An end-to-end watermarking framework for ownership protection of deep neural networks. Proceedings of the 24th International Conference on Architectural Support for Programming Languages and Operating Systems. Providence: ACM, 2019. 485–497.
24 Fan LX, Ng KW, Chan CS, *et al*. DeepIPR: Deep neural network ownership verification with passports. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(10): 6122–6139. [doi: 10.1109/TPAMI.2021.3088846]
25 Jia HR, Choquette-Choo CA, Chandrasekaran V, *et al*. Entangled watermarks as a defense against model extraction. Proceedings of the 30th USENIX Security Symposium. USENIX Association, 2021. 1937–1954.
26 Maini P, Yaghini M, Papernot N. Dataset inference: Ownership resolution in machine learning. Proceedings of the 2021 International Conference on Learning Representations. Vienna: ICLR, 2021.
27 Park S, Abuadbba A, Wang S, *et al*. Tracking dataset IP use in deep neural networks. arXiv:2211.13535, 2022.
28 Dziedzic A, Duan HN, Kaleem MA, *et al*. Dataset inference for self-supervised models. Proceedings of the 36th Conference on Neural Information Processing Systems. New Orleans: OpenReview.net, 2022. 1–13.
29 Tran B, Li J, Mądry A. Spectral signatures in backdoor attacks. Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montréal: Curran Associates Inc., 2018. 8011–8021.
30 Shafahi A, Huang WR, Najibi M, *et al*. Poison frogs! Targeted clean-label poisoning attacks on neural networks.

Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montréal: Curran Associates Inc., 2018. 6106–6116.

31 Zhu C, Huang WR, Li HD, *et al*. Transferable clean-label poisoning attacks on deep neural nets. Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019. 7614–7623.

32 Aghakhani H, Meng DY, Wang YX, *et al*. Bullseye polytope: A scalable clean-label poisoning attack with improved transferability. Proceedings of the 2021 IEEE European Symposium on Security and Privacy, EuroS&P 2021. Vienna: IEEE, 2021. 159–178.

33 Turner A, Tsipras D, Madry A. Label-consistent backdoor attacks. arXiv:1912.02771, 2019.

34 Saha A, Subramanya A, Pirsiavash H. Hidden trigger backdoor attacks. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI Press, 2020. 11957–11965.

35 Huang WR, Geiping J, Fowl L, *et al*. Metapoison: Practical general-purpose clean-label data poisoning. Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 1013.

36 Geiping J, Fowl LH, Huang WR, *et al*. Witches ' brew: Industrial scale data poisoning via gradient matching. International Conference on Learning Representations. Vienna: ICLR, 2021.

37 Souri H, Fowl L, Chellappa R, *et al*. Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch. arXiv:2106.08970, 2022.

38 Zheng TH, Li BC. First-order efficient general-purpose clean-label data poisoning. Proceedings of the 2021 IEEE Conference on Computer Communications. Vancouver: IEEE, 2021. 1–10.

39 Schwarzschild A, Goldblum M, Gupta A, *et al*. Just how toxic is data poisoning? A unified benchmark for backdoor and data poisoning attacks. Proceedings of the 38th International Conference on Machine Learning. PMLR, 2021. 9389–9398.

40 Wang Z, Bovik AC, Sheikh HR, *et al*. Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 2004, 13(4): 600–612. [doi: 10.1109/TIP.2003.819861]

41 Kolkin N, Salavon J, Shakhnarovich G. Style transfer by relaxed optimal transport and self-similarity. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE 2019. 10043–10052.

42 Svoboda J, Anoosheh A, Osendorfer C, *et al*. Two-stage peer-regularized feature recombination for arbitrary image style transfer. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 13813–13822.

43 Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks. Proceedings of the 34th International Conference on Machine Learning. Sydney: PMLR, 2017. 1126–1135.

44 Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images [Master's Thesis]. Toronto: University of Toronto, 2009.

45 Tan MX, Le QV. Efficientnet: Rethinking model scaling for convolutional neural networks. Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019. 6105–6114.

(校对责编: 牛欣悦)