基于通用逆扰动的对抗攻击防御方法

陈晋音[1, 2] 吴长安[2] 郑海斌[2] 王巍[3] 温浩[4]

摘 要 现有研究表明深度学习模型容易受到精心设计的对抗样本攻击, 从而导致模型给出错误的推理结果, 引发潜在的安全威胁. 已有较多有效的防御方法, 其中大多数针对特定攻击方法具有较好防御效果, 但由于实际应用中无法预知攻击者可能采用的攻击策略, 因此提出不依赖攻击方法的通用防御方法是一个挑战. 为此, 提出一种基于通用逆扰动 (Universal inverse perturbation, UIP) 的对抗样本防御方法, 通过学习原始数据集中的类相关主要特征, 生成通用逆扰动, 且 UIP 对数据样本和攻击方法都具有通用性, 即一个 UIP 可以实现对不同攻击方法作用于整个数据集得到的所有对抗样本进行防御. 此外, UIP 通过强化良性样本的类相关重要特征实现对良性样本精度的无影响, 且生成 UIP 无需对抗样本的先验知识. 通过大量实验验证, 表明 UIP 在不同数据集、不同模型中对各类攻击方法都具备显著的防御效果, 且提升了模型对正常样本的分类性能.

关键词 深度学习, 通用逆扰动, 对抗样本, 通用防御

引用格式 陈晋音, 吴长安, 郑海斌, 王巍, 温浩. 基于通用逆扰动的对抗攻击防御方法. 自动化学报, 2023, **49**(10): 2172−2187

DOI 10.16383/j.aas.c201077

Universal Inverse Perturbation Defense Against Adversarial Attacks

CHEN Jin-Yin[1, 2] WU Chang-An[2] ZHENG Hai-Bin[2] WANG Wei[3] WEN Hao[4]

Abstract Existing studies have shown that deep learning models are vulnerable to carefully crafted adversarial sample, leading to wrong decision by the model, which will cause potential security threats. Many effective defense methods have been proposed, most of which have good defense effects against specific attack methods. However, since the possible strategies of attackers cannot be predicted in practical applications, it is a challenge to propose a general defense that does not rely on attack methods. This paper proposes a defense method based on universal inverse perturbation (UIP), which generates universal inverse perturbation by learning important features of classes in the original data. UIP is universal to data and attack methods, that is, one UIP can realize defense against all samples obtained by different attack methods acting on the entire data set. In addition, UIP can guarantee the accuracy of benign samples by enhancing the important characteristics of the benign samples, and the generation of UIP does not require prior knowledge of adversarial samples. Extensive experiments are carried out to testify that UIP has a significant defense effect against various attack methods in different data sets and different models, and the model′s classification performance for normal samples is improved as well.

Key words Deep learning, universal inverse perturbation (UIP), adversarial example, general defense

Citation Chen Jin-Yin, Wu Chang-An, Zheng Hai-Bin, Wang Wei, Wen Hao. Universal inverse perturbation defense against adversarial attacks. *Acta Automatica Sinica*, 2023, **49**(10): 2172−2187

收稿日期 2020-12-28 录用日期 2021-04-16
Manuscript received December 28, 2020; accepted April 16, 2021
国家自然科学基金 (62072406), 浙江省自然科学基金 (LY19F020025), 教育部产学合作协同育人项目资助
Supported by National Natural Science Foundation of China (62072406), Natural Science Foundation of Zhejiang Province (LY19F020025), and Ministry of Education Industry-University Cooperation Collaborative Education Project
本文责任编委 赫然
Recommended by Associate Editor HE Ran

1. 浙江工业大学网络空间安全研究院 杭州 310023 2. 浙江工业大学信息工程学院 杭州 310023 3. 中国电子科技集团公司第三十六研究所 嘉兴 314001 4. 重庆中科云从科技有限公司 重庆 401120
1. Institute of Cyberspace Security, Zhejiang University of Technology, Hangzhou 310023 2. College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023 3. The 36th Research Institute of China Electronics Technology Group Corporation, Jiaxing 314001 4. Chongqing Zhongke Yuncong Technology Co., Ltd., Chongqing 401120

随着计算机硬件计算力的发展, 深度学习技术[1]凭借其良好的性能和较强的拟合能力广泛应用于计算机视觉[2]、自然语言处理[3]、语音识别[4]、工业控制[5]等领域. 然而, 近期研究发现, 深度学习模型容易受到精心制作的微小扰动的影响[6]. 对抗攻击可以定义为: 在模型测试阶段, 攻击者通过在原始数据上添加精心设计的微小扰动得到对抗样本, 从而使得深度学习模型完全失效并以较高置信度误判的恶意攻击. 在应用深度模型的各个领域, 对抗样本均可实现较高概率的攻击, 如何设计高效的防御方法提高深度学习模型的鲁棒性是其进一步推广应用的安全保障[7].

已有大量面向深度学习的对抗攻击研究工作, 根据其对抗样本生成原理不同, 可分为基于梯度的

攻击方法、基于优化的攻击方法和其他攻击方法[7]. 其中, 基于梯度的攻击方法利用模型的参数信息, 通过目标损失函数对输入的求导得到梯度信息, 获取对抗扰动, 例如: 快速梯度符号法 (Fast gradient sign method, FGSM[8])、动量迭代的快速梯度符号法 (Momentum iterative fast gradient sign method, MI-FGSM[9])、基于雅克比的显著图攻击 (Jacobian-based saliency map attack, JSMA)[10] 等. 基于优化的攻击方法通过多次查询样本的输出置信度或类标, 优化对抗扰动, 或者通过等价的梯度信息进行攻击, 例如: 基于零阶优化的攻击 (Zeroth order optimization, ZOO)[11] 和基于边界的攻击 (Boundary)[12]. 相比于基于梯度的攻击, 基于优化的攻击方法由于需要多次查询计算, 因此算法复杂度和运行成本都较高. 除此之外, 还有基于生成式对抗网络 (Generative adversarial network, GAN) 的攻击[13]、基于迁移的攻击[14] 等.

随着对抗攻击研究的深入, 相应的对抗攻击防御方法的研究也相继展开, 根据防御方式的差异, 可分为基于数据修改的防御、基于模型修改的防御和基于附加网络的防御[7]. 其中, 基于数据修改的防御对模型的输入进行修改, 包括数据重编码、数据变换、对抗训练等; 基于模型修改的防御包括修改模型的目标损失、在模型中加入随机层、“蒸馏”得到新的网络等; 基于附加网络的防御包括添加扰动整流网络、自编码器网络、生成式对抗网络等. 已有的防御方法研究大多关注防御成功率, 在实际应用中仍面临以下一些挑战:

1) 对抗样本依赖, 即防御的效果依赖于预先已知的对抗样本的数量和质量, 如对抗训练, 当遇到新的攻击方法时防御效果不明显;

2) 影响良性样本的识别精度, 即防御的效果以牺牲良性样本的识别精度为代价, 如随机缩放图像操作虽然能够破坏对抗扰动, 但也干扰了良性样本识别;

3) 参数敏感性与防御实时性, 即需要根据数据集和攻击方法调整参数, 如数据变换中的图像缩放和图像旋转需要多次测试得到合适的参数, 附加网络防御方法增加了计算步骤, 降低了模型的处理速度.

通用对抗扰动攻击方法[15] 是不断对对抗样本的扰动进行叠加和优化, 得到通用扰动, 随后叠加到任意良性样本上都能够实现攻击. 受到通用对抗扰动攻击[15] 的启发, 本文提出一种基于通用逆扰动 (Universal inverse perturbation, UIP) 的对抗样本防御方法 (UIP defense, UIPD), 通过设计具有通用逆扰动的矩阵, 叠加到对抗样本, 实现对抗样本的重识别防御. 此外, 对抗样本鲁棒特征的提出[16], 认为样本包含鲁棒特征和非鲁棒特征, 且都会影响预测结果. 良性样本中两者一致因此得到正确识别结果; 而对抗样本中鲁棒特征不受影响, 非鲁棒特征变化较大, 影响了识别结果. 因此, 可以通过设计强化样本中的非鲁棒特征, 即类相关特征, 实现对对抗样本的防御, 抵消对抗扰动对非鲁棒特征的影响; 而且根据非鲁棒特征在数据分布中的相似性和通用性, 设计生成通用逆扰动进行抵消.

本文的主要贡献如下:

1) 设计一种基于通用逆扰动的对抗样本防御方法 UIPD, 仅依据良性样本即可快速生成通用逆扰动矩阵, 有效防御多种未知的攻击方法;

2) UIPD 不影响良性样本的识别, 在生成 UIP 的过程中, 通过对良性样本的类相关特征进行强化, 实现良性样本识别精度提升的效果;

3) UIPD 的参数敏感性低且防御速度快, 在多个数据集和多个模型上的实验结果表明了 UIPD 对各类攻击都具有良好的防御效果.

本文其余部分结构如下: 第 1 节介绍了对抗攻防的相关工作; 第 2 节详细说明了 UIPD 方法; 第 3 节实验从多个角度验证 UIPD 的性能; 最后对全文进行总结和展望, 更多的通用逆扰动可视化图示例参见附录 A.

1 相关工作

本节主要介绍实验中涉及到的对抗攻击方法与已有的防御方法.

1.1 对抗攻击方法

已有的对抗攻击方法众多, 根据对抗样本的生成机理, 可以分为以下两类:

1) 基于梯度的攻击: 指在基于梯度的迭代过程中, 寻找图像中关键的像素点进行扰动. Szegedy 等[6] 首次证明了可以通过对图像添加无法察觉的扰动误导网络做出错误分类. 但由于问题的复杂度太高, 于是转而求解简化后的问题, 将其称为约束型拟牛顿法 (Box-constrained limited memory Broyden-Fletcher-Goldfarb-Shanno, L-BFGS). Goodfellow 等[8] 在此基础上, 提出快速梯度符号法 (FGSM), 通过计算单步梯度快速生成对抗样本. Madry 等[17] 提出投影梯度下降法 (Project gradient descent, PGD), 可以将其看作是 FGSM 的改进版 —— K-FGSM (K 表示迭代的次数), 每次迭代都会将扰动限制到规定范围, 提高攻击的有效性. Kurakin 等[18] 提出基本迭代法 (Basic iterative methods, BIM), 将一大步运算扩展为通过多个小步增大损失函数,

从而提高对抗样本的攻击成功率并且减小对抗扰动. Carlini 等[19] 提出一种对抗攻击方法 C&W, 通过梯度迭代优化的低扰动对抗样本生成算法, 限制 L_∞、L_2 和 L_0 范数使得扰动无法被察觉, 但是攻击速度较慢. Moosavi-Dezfooli 等[20] 提出了深度欺骗攻击 (DeepFool), 通过迭代计算的方法生成最小规范对抗扰动, 将位于分类边界内的图像逐步推到边界外, 直到出现错误分类. 此方法添加的对抗性扰动比 FGSM 更小, 同时能够达到相似的攻击效果. 一般攻击方法均采用限制 L_2 或 L_∞ 范数的值控制扰动, 而 Papernot 等[10] 提出基于雅克比的显著图攻击 (JSMA), 采取限制 L_0 范数的方法, 即仅改变良性样本几个像素生成对抗样本, 使得添加的扰动更小. 一般的攻击方法只能针对单个样本生成对抗扰动, Moosavi-Dezfooli 等[15] 研究并设计了一种通用对抗扰动 (Universal adversarial perturbation, UAP) 攻击, 与 DeepFool 攻击相似, 使用对抗扰动将图像推出分类边界, 但是同一个扰动针对的是所有的图像, 结果显示即使是当时最优的深度网络模型也难以抵抗通用扰动的攻击. 此外, 通用的对抗扰动具有很强的迁移性, 即跨数据集、跨模型有效.

2) 基于优化的攻击: 通过将对抗样本的生成问题转化为多目标的优化问题, 使分类模型损失最大化, 对抗扰动最小化, 导致模型分类错误. Brendel 等[12] 提出边界攻击, 通过对样本引入最小扰动来改变模型对样本的决策. 受 C&W 攻击的启发, Chen 等[11] 提出基于零阶优化的攻击 (ZOO), 使用对称差商来估计梯度, 进行对抗扰动的优化更新. 通过在样本中添加噪声并进行对抗扰动优化是一种常见的对抗攻击方法, Rauber等[21] 提出在样本中添加高斯噪声 (Additive Gaussian noise attack, AGNA) 使分类器出错, 添加的扰动是通过多次迭代优化直到使分类器出错的最小扰动. 除此以外, Rauber 等[21] 通过改变添加的噪声类型, 提高攻击的效率, 如添加均匀噪声 (Additive uniform noise attack, AUNA) 和添加椒盐噪声 (Salt and pepper noise attack, SPNA).

本文提出的 UIPD 在上述的对抗攻击中均取得了良好的防御效果, 除了上述的对抗攻击方法以外, 还有很多其他优秀的对抗攻击方法: Su 等[22] 提出单像素攻击 (One pixel attack), 使用差分进化算法, 对每个像素进行迭代的修改生成子图像, 并与原图像对比, 根据选择标准保留攻击效果最好的子图像, 仅改变图样本中的一个像素值就可以实现对抗攻击. Baluja 等[23] 训练了多个对抗性转移网络 (Adversarial transformation networks, ATNs) 来生成对抗样本, 可用于攻击一个或多个网络模型. Cisse 等[24] 通过生成特定于任务损失函数的对抗样本实现对抗攻击, 即利用网络的可微损失函数的梯度信息生成对抗扰动. Sarkar 等[25] 提出了两种对抗攻击算法: 精确目标的通用扰动 (Universal perturbations for steering to exact targets, UPSET) 攻击和生成恶意图像的对抗网络 (Antagonistic network for generating rogue images, ANGRI) 攻击. UPSET 攻击为针对原始样本生成具有通用扰动的对抗样本, 且可以使模型误分类为指定的目标类别, 而 ANGRI 攻击为针对原始样本生成具有特定扰动的对抗样本, 且可以使模型误分类为指定的目标类别.

以上攻击方法都是基于肉眼不可见扰动的对抗攻击, 除了基于对抗扰动的攻击外, 还有一类基于对抗补丁的攻击. Brown 等[26] 提出一种在物理空间的对抗图像补丁的方法. Karmon 等[27] 利用修改后的损失函数, 使用基于优化的方法提升对抗补丁的鲁棒性. 为了提高视觉保真度, Liu 等[28] 提出了 PS-GAN 框架来生成类似涂鸦的对抗补丁, 以愚弄自动驾驶系统. 为了解决对抗补丁泛化能力差的问题, Liu 等[29] 利用模型的感知和语义上的偏见, 提出了一个基于偏见的框架生成具有强泛化能力的通用对抗补丁方法. 综上, 基于补丁的对抗攻击也是一种有效的攻击方法.

1.2 对抗防御方法介绍

根据防御效果, 防御方法可分为仅检测防御和重识别防御, 仅检测防御是对检测出的攻击样本进行甄别, 而不做进一步处理; 重识别防御则是将对抗样本进行还原处理, 重新识别其正确类标, UIPD 属于重识别防御方法, 因此在实验中采用的对比算法同样都属于重识别防御. 而根据防御作用对象的不同, 可以进一步分为以下三类:

1) 基于数据预处理的防御: 指在模型训练前, 或模型测试的过程中, 对数据进行预处理, 从而提高模型对于对抗样本的防御性. Xie 等[30] 研究发现, 对图像进行尺寸变换或者空间变换能有效降低对抗样本的攻击性能, 这是一种非常简单有效的数据预处理防御方法, 但无法从根本上提升模型的防御能力. Song 等[31] 提出了对抗训练方法, 通过生成的大量对抗样本, 然后将对抗样本作为模型的训练集执行对抗训练, 从而不断提升模型的鲁棒性, 该方法需要使用大量高强度的对抗样本, 并且网络架构要有充足的表达能力, 高度依赖于对抗样本的数量和质量, 面对多种攻击组合时防御的泛化能力较弱. 为此, Miyato 等[32] 和 Zheng 等[33] 分别提出了虚拟对抗训练和稳定性训练方法提升防御效果. Dziugaite

等[34] 提出基于数据压缩的方法, 使用 JPG 图像压缩的方法, 减少对抗扰动对于模型的干扰, 但同时也会降低对良性样本的分类准确率. 此外, Das 等[35] 通过研究数据中的高频成分, 提出了集成防御技术. Luo 等[36] 提出基于“Foveation”机制的防御方法提高显著鲁棒性. 对抗训练能够提高深度模型的鲁棒性, 但是需要生成大量的对抗样本, 存在防御代价大、无法防御没有出现过的攻击等问题.

2) 基于网络修正的防御: 指通过添加或者改变多层/子网络、改变损失/激活函数等方式, 改变模型的架构和参数, 从而滤除扰动, 提高模型的防御性. 受到将去噪自编码器 (Denoising auto encoders, DAE) 堆叠到原来的网络上会使其变得更加脆弱这一特性的启发, Gu 等[37] 引入深度压缩网络 (Deep compression network, DCN), 减少对抗样本的扰动. Rifai 等[38] 通过添加平滑操作训练 DCN 滤除扰动. Ross 等[39]提出使用输入梯度正则化以提高对抗攻击鲁棒性, 该方法和对抗训练结合有很好的效果, 但防御代价以及防御的复杂度都会提高一倍以上. Hinton 等[40] 提出可以使用“蒸馏”的方法将复杂网络的知识迁移到简单网络上后, Papernot 等[41] 基于“蒸馏”的概念设计对抗防御方法, 通过解决数值不稳定问题扩展了防御性蒸馏方法. Nayebi 等[42] 受生物启发, 使用类似于生物大脑中非线性树突计算的高度非线性激活函数以防御对抗攻击. Cisse 等[43] 提出了在一层网络中利用全局 Lipschitz 常数加以控制, 利用保持每一层的 Lipschitz 常数来减少对抗样本的干扰的防御方法. Gao 等[44] 提出 DeepCloak 方法, 在分类层的前一层加上特意为对抗样本训练的额外层以掩盖对抗扰动. 此外, Jin 等[45] 通过引入前馈神经网络添加额外噪声减轻攻击的影响. Sun 等[46] 基于统计滤波设计了超网络提高网络鲁棒性. Madry 等[17] 从鲁棒优化角度研究了对抗防御性. 通过网络修正的方式改变模型内部结构和参数的优化能够有效提高模型的鲁棒性, 采取梯度隐蔽、蒸馏结构、激活函数重设计等措施提高模型防御性能.

3) 基于附加网络的防御: 指在保持原始深度学习模型结构不变的前提下, 添加外部模型作为附加网络来提高原始模型防御性能. 针对对抗攻击的防御, Akhtar 等[47] 通过添加扰动整流网络, 利用一个单独训练的网络附加到目标网络上, 以抵御通用扰动产生的对抗性攻击, 达到不需要调整原本的网络参数也能对对抗样本产生良好的防御效果的目的. Hlihor 等[48] 在训练过程中将对抗样本提供给自动编码器, 从而滤除对抗性扰动, 并减少输出样本与干净样本之间的距离. 孔锐等[49] 研究了基于 GAN 框架训练目标模型的鲁棒性. Samangouei 等[50] 使用 GAN 生成与对抗样本相似但不含扰动的样本, 实现防御. Lin 等[51] 在 Samangouei 等[50] 的工作基础上, 在 GAN 结构中引入自编码器, 提高防御效率. Jin 等[52] 提出对抗扰动滤除的生成式对抗网络 (Adversarial perturbation elimination with GAN, APE-GAN), 利用对抗样本训练基于 GAN 的防御模型, 达到正确识别对抗样本, 同时不影响干净样本的识别的目的. Xu 等[53] 提出特征压缩法, 用两个近似模型检测图像中的对抗扰动. Ju 等[54] 研究了多个模型的集成决策防御, 提出了一种集成对抗防御方法.

本文提出的通用逆扰动对抗防御方法与贝叶斯案例模型 (Bayesian case model, BCM)[55] 通过选择数据中具有代表性的典型样本, 然后提取典型样本中的重要特征, 达到对基于案例推理算法和原型分类算法的解释, 在思想上相似, 但是主要任务、技术方法与应用场景均不同.

2 基于通用逆扰动的防御方法

2.1 基本定义

通常, 神经网络的前向传播过程表示为 $f:\mathbf{R}^M\rightarrow\mathbf{R}^N$, 其中 M 表示输入的维度, N 表示输出的维度. 进一步, 可以将整个模型表示为: $f(x,\theta):X\rightarrow Y$, 其中 $x\in X$ 表示模型输入, Y 表示模型的输出, θ 表示模型的内部参数. 进一步将 θ 表示为深度模型的各层非线性权重与偏置组合: $\theta=w\times\phi(x)+b$, 其中 w 表示权重矩阵, 在训练的过程中更新, $x\in X$ 表示输入矩阵, 即原始数据集中的良性样本, b 表示偏置, $\phi(x)$ 表示输入样本特征. $y\in Y$ 表示良性样本的真实类标经过 one-hot 编码后的数组, $l=\arg\max(y)$, $\arg\max(\cdot)$ 表示数组元素值最大的位置的坐标作为真实类标, $l\in\{0,1,2,\cdots,N-1\}$. $\hat{y}=f(x,\theta)$表示良性样本的预测置信度数组, $\hat{l}=\arg\max(\hat{y})$ 表示预测类标, $\hat{l}=\{0,1,2,\cdots,N-1\}$. 当 $\hat{l}=l$ 时, 则预测正确, 反之则预测错误. 以交叉熵为例, 定义模型训练的损失函数为

$$Loss_{\mathrm{CE}}=-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\hat{y}^{(i)}+(1-y^{(i)})\log(1-\hat{y}^{(i)})\right] \tag{1}$$

其中, m 表示训练样本数, $y^{(i)}$ 和 $\hat{y}^{(i)}$ 分别表示数组 y 和 $\hat{y}$ 在位置 i 处的值, $\log(\cdot)$ 表示对数函数. 训练的优化目标是最小化损失, 即 $\arg\min Loss_{\mathrm{CE}}$, 一般采用梯度下降法, 梯度计算式为

$$g_w = \frac{\partial Loss_{\rm CE}}{\partial w} \tag{2}$$

进一步得到权重的更新式为

$$w_{i+1} = w_i - lr \times g_w \tag{3}$$

其中, lr 表示学习率.

当模型受到攻击后, 攻击者会在良性样本上添加精心设计的扰动得到对抗样本, 表示为 $x^* = x + \Delta x$, 其中 Δx 表示对抗扰动. 将对抗样本输入模型后, 得到 $\hat{y}^* = f(x^*)$ 是对抗样本的预测置信度数组, $\hat{l}^* = \arg\max(\hat{y}^*)$ 表示预测类标, $\hat{l}^* = \{0, 1, 2, \cdots, N-1\}$. 当 $\hat{l}^* \neq l$ 时, 则无目标攻击成功; 当 $\hat{l}^* = l_t$ 时, 其中 l_t 是攻击者预设的攻击目标, 则目标攻击成功; 当 $\hat{l}^* = l$ 时, 则攻击失败. 攻击的目的是实现损失的增大, 即 $\arg\max Loss_{\rm CE}$, 同样采用梯度下降计算

$$g_x = \frac{\partial Loss_{\rm CE}}{\partial x} \tag{4}$$

进一步得到对抗样本的更新式为

$$x^*_{i+1} = x^*_i + \varepsilon \times g_x \tag{5}$$

其中, ε 表示迭代步长, "+"运算表示样本与对抗扰动叠加.

最后, 使用防御方法加固模型后, 重新实现损失的最小化. 根据前面的定义, 可以采用对权重更新, 也可以采用对样本更新, UIPD 方法是对样本进行更新实现防御, 恰好是式 (5) 的逆过程, 可以粗略表示为

$$x'_{i+1} = x'_i - \varepsilon \times g_x \tag{6}$$

其中, 为避免混淆, 使用 x'_i 表示良性样本的更新过程, "–"运算表示良性样本的防御强化, 减少样本中的对抗扰动.

2.2 通用逆扰动生成方法

通用逆扰动的通用性体现在: 测试阶段, 只需单个逆扰动, 就可以对不同攻击方法生成的任意对抗样本实现防御; 训练阶段, 不涉及到攻击方法和对抗样本. 生成过程如图 1 所示, 其中 UIP 与训练集样本的尺寸和维度一致, 首先初始化为 0; 然后分别和训练集中的每一张样本叠加后输入到深度模型中, 计算损失函数; 最后根据损失的趋势得到逆扰动在特征空间中的位置, 反馈训练更新通用逆扰动.

图 1 的方法框图中包括 UIP、良性样本和深度神经网络 (Deep neural network, DNN) 模型三部分, UIP 通过对图像空间的特征进行不断迭代强化, 提取良性样本的特征, 并通过反馈训练对 UIP 不断进行加强. 在迭代过程中, 图 1 形象地展示了通用逆扰动与良性样本、特征空间的关系. 在前文中提到, 通用逆扰动强化了良性样本的类相关特征, 因此能够保持良性样本的识别准确率, 甚至在一定范围内提升识别准确率. 但是通用逆扰动不是直接采样自样本空间, 而是通过损失反馈训练学习其在高维特征空间中的分布, 这解释了通用逆扰动对数据样本和攻击方法具有较好的通用性, 但是对同一个数据集的训练模型的通用性则较差.

根据式 (6) 和图 1 的说明可以得到通用逆扰动的生成式. 首先令 $x'_i = x_i + \rho_i^{\rm uip}$, 则深度模型变为

$$f(x) = f(x_i + \rho_i^{\rm uip}) \tag{7}$$

其中, x_i 表示原样本, $\rho_i^{\rm uip}$ 表示通用逆扰动矩阵, x'_i 表示原样本叠加上通用逆扰动矩阵后的样本.

此时的梯度是损失函数对叠加后的输入进行求导, 得到

$$g_{x_i} = \frac{\partial Loss_{\rm CE}}{\partial (x_i + \rho_i^{\rm uip})} \tag{8}$$

其中, g_{x_i} 表示此时的梯度, $Loss_{\rm CE}$ 表示交叉熵损失函数.

进一步得到修改后的 UIP 迭代式

$$(x_i + \rho_i^{\rm uip})_{\rm new} = (x_i + \rho_i^{\rm uip})_{\rm old} - \varepsilon^{\rm uip} \times g_{x_i} \tag{9}$$

因为其中良性样本在迭代前后不变, 所以两边减去一个 x_i, 得到最终 UIP 迭代式

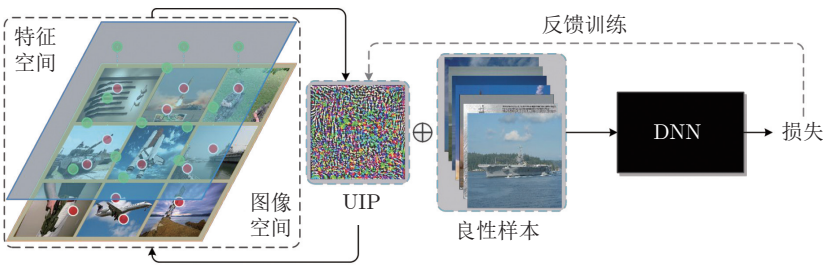


图 1 通用逆扰动防御方法框图

Fig. 1 The framework of UIPD method

$$\rho_{i+1}^{\text{uip}} = \rho_i^{\text{uip}} - \varepsilon^{\text{uip}} \times g_{x_i} \tag{10}$$

其中, ε^{uip} 表示通用逆扰动矩阵的迭代步长.

需要说明的是, 图 1 中的 UIP 即是在 ImageNet 数据集、VGG19 模型上优化得到的通用逆扰动, 为了更好的可视化, 将其归一化到 [0, 1] 的范围内进行可视化, 原始的 UIP 的均值为: −0.0137, 方差为: 0.0615, 是十分微小的. UIPD 方法的详细伪代码如算法 1 所示.

算法 1. UIPD 方法

输入. 良性样本集 X, 分类器 $f(x)$, 逆扰动步长 ε^{uip}, 最大 epoch 数 N

输出. 通用逆扰动 ρ^{uip}

初始化: $\rho_0^{\text{uip}} = 0$

For $k = 1 : N$ **do**

  For $x_i \in X$ **do**

    基于式 (1) 和式 (7) 计算 $Loss_{\text{CE}}$

    基于式 (8) 计算 g_{x_i}

    基于式 (10) 计算 ρ_{i+1}^{uip}

  End For

End For

2.3　算法复杂度分析

对算法的时间复杂度进行分析, UIPD 的时间复杂度包括训练时间复杂度和测试时间复杂度, 根据算法 1 可知, 其训练的时间复杂度和测试的时间复杂度都是 O(n), 都是与样本数呈一阶增长关系. 尽管在算法 1 中存在两个“For”循环语句, 但是最大 epoch 数是一个常数, 因此训练时间复杂度是 O(n_{train}); 而测试时, 只需要将良性样本与 UIP 做“+”运算操作 (“+”运算操作是指将训练完成的 UIP 与良性样本进行像素上的叠加, 即将 UIP 以一种“扰动”的形式添加到良性样本图像上去, 在完成“+”操作的过程中, 需要先将 UIP 与良性样本转化为数组像素值, 完成“+”操作后再以图像形式输出), 因此也是 O(n_{test}), 其中 n_{train} 和 n_{test} 表示训练样本数和测试样本数. 相比于数据修改防御中的数据变换操作, 如 resize、rotate 等, UIPD 方法多了训练的时间复杂度, 但是由于 UIP 能够进行离线训练和在线防御, 训练样本是有限的, 即 $n_{\text{test}} \gg n_{\text{train}}$, 因此其训练时间复杂度是可以忽略的; 相比于对抗训练, UIPD 方法不需要使用大量的对抗样本进行训练, 节省了大量的对抗样本生成时间.

分析空间复杂度, 无论是在训练过程, 还是在测试过程, UIPD 方法都是只需要占据一个 UIP 存储的空间, 因此空间复杂度是 O(1).

2.4　通用逆扰动有效性分析

本文从高维特征的决策边界和样本的鲁棒安全边界两个角度说明通用逆扰动的有效性. 基于样本在高维特征空间中的分布和决策边界, 分析 UIP 具有防御效果的原因. 如图 2 所示, UIPD 方法不改变模型的决策边界, 因此决策边界是固定的, 但样本在决策空间的位置与决策边界是相对而言的, UIP 导致样本在决策空间中的位置发生了变化, 导致样本与决策边界的相对位置发生了变化, 使得原本在错误决策空间的样本重新回到正确决策空间. 当训练好一个模型, 良性样本被正确分为 C1 类和 C2 类, 其中还存在 C2 类的一个样本被误分类为 C1 (图中的灰色方块). 当良性样本叠加了 UIP 后, 能够促使样本在特征空间中的分布向类中心移动, 从而改善良性样本识别结果 (即将原本分类错误的样本进行正确识别). 当模型受到攻击, 原本处在决策边界附近的样本越过边界进入错误类的特征空间 (即图中的红色圆点). 此时, 当对抗样本叠加了 UIP 后, 能够重新回到正确的特征空间并向类中心移动.

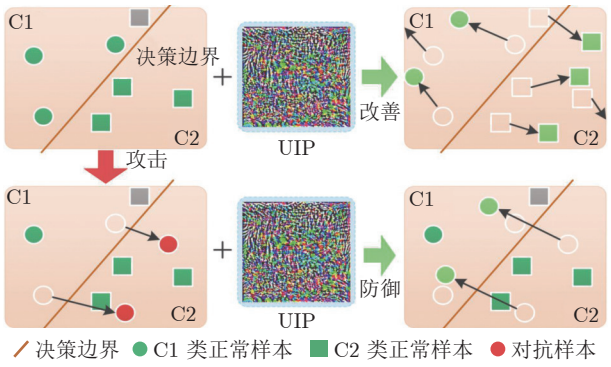


图 2　基于特征分布和决策边界的 UIPD 分析示意图

Fig. 2　The UIPD analysis based on feature distribution and decision boundary

基于样本的鲁棒安全边界说明 UIP 具有防御效果的原因, 具体如图 3 所示. 最优化观点认为, 模型的鲁棒性可以等价为一个最大最小模型. 最大化攻击者的目标函数, 其物理意义是寻找合适的扰动使损失函数在 $(x + \Delta x, y)$ 这个样本点上的值越大越好; 最小化防御者的目标函数, 其目的是为了让模型在遇到对抗样本的情况下, 整个数据分布上的损失的期望还是最小. 基于最优化观点建模的计算式为

$$\min_{w} \rho(w)$$
$$\rho(w) = \mathrm{E}_{(x,y)\sim D}\left[\max_{\Delta x \in S_x} L(w, x + \Delta x, y)\right] \tag{11}$$

其中, $\rho(\cdot)$ 是需要最小化的防御目标, w 表示权重矩

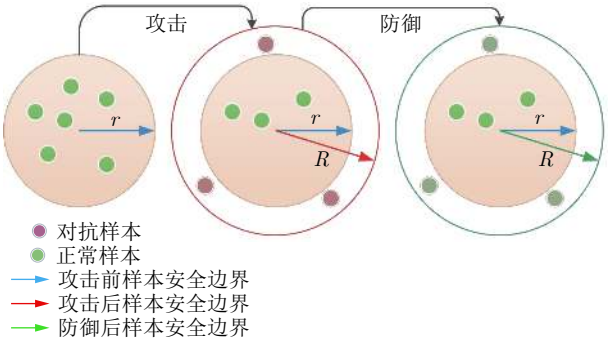


图 3 基于鲁棒安全边界的 UIPD 分析示意图

Fig. 3 The UIPD analysis based on robust security boundaries

阵, x 表示输入矩阵, y 表示样本标签, $\mathrm{E}_{(x,y)\sim D}[\cdot]$ 表示平均损失, $D(x, y)$ 表示输入和标签所在的联合概率分布, Δx 表示对抗扰动, $L(\cdot, \cdot, \cdot)$ 表示损失函数. 式中 $\Delta x \in S_x$, 即此时对抗样本的扰动落在 S_x 范围内都是安全的, 因此将 S_x 称为输入扰动的安全边界.

图 3 中, 良性样本的安全边界原本是 r, 即 $S_x \leq r$ 时为安全; 受到攻击后, 发生了样本点落在半径 r 以外的事件, 但是若此时能够将安全边界由 r 拓展到 R, 则可以实现新的鲁棒边界; UIPD 方法的防御过程就是通过学习数据样本在高维特征空间中的类相关重要特征, 然后反映在图像空间中, 最后等效于将 $S_x \leq r$ 的安全边界拓展为 $S_x \leq R$.

3 实验与结果

本节首先介绍实验基本设置, 包括软硬件环境、数据集、深度模型、攻击方法、防御方法、评价指标等. 然后, 从 UIP 在攻击方法上的通用性、数据样本上的通用性, 与不同防御方法的防御效果比较, 在良性样本识别中的性能改善、参数敏感性和时间复杂度等方面进行实验和分析.

3.1 基本实验设置

1) 实验硬件及软件平台: i7-7700K 4.20 GHz×8 (CPU), TITAN Xp 12GiB×2 (GPU), 16 GB×4 memory (DDR4), Ubuntu16.04 (OS), Python3.7, Tensorflow-gpu 1.1.14, Tflearn 0.3.2.2.

2) 数据集: 实验采用 MNIST、Fashion-MNIST (FMNIST)、CIFAR-10 和 ImageNet 四个公共数据集. 其中, MNIST 数据集包括 10 类共 60 000 张训练样本及 10 类共 10 000 张测试样本, 样本大小是 28×28 的灰度图像; CIFAR-10 数据集由 10 类共 50 000 张训练样本及 10 类共 10 000 张测试样本组成, 样本是大小为 $32 \times 32 \times 3$ 的彩色图片; FMNIST 数据集包括 10 类共 60 000 张训练样本及 10 类共 10 000 张测试样本, 样本大小是 28×28 的灰度图像; ImageNet 数据集由 1 000 多类共计 200 多万张样本组成, 本文随机挑选训练集中的 10 类图片进行实验, 每类 1 300 张样本, 其中 70% 作为训练样本, 30% 作为测试样本. 实验中的所有图像像素值都归一化到 [0, 1].

3) 深度模型: 针对 MNIST 数据集, 分别使用 AlexNet、LeNet 和自己搭建的网络结构 (M_CNN); 针对 FMNIST 数据集, 分别使用 AlexNet 和自己搭建的网络 (F_CNN); 针对 CIFAR-10 和 ImageNet 数据集, 都使用 VGG19 网络. 由于 MNIST 和 FMNIST 数据集十分相似, 实验中 M_CNN 和 F_CNN 使用相同的结构, 如表 1 所示. 深度模型的训练参数采用 Tflearn 提供的默认参数.

表 1 自行搭建的网络模型结构

Table 1 The network structure built by ourselves

网络层	M_CNN/F_CNN
Conv + ReLU	$5 \times 5 \times 5$
Max pooling	2×2
Conv + ReLU	$5 \times 5 \times 64$
Max pooling	2×2
Dense (Fully connected)	1024
Dropout	0.5
Dense (Fully connected)	10
Softmax	10

4) 攻击方法: 为了证明生成的 UIP 对于不同攻击方法的通用性, 采用了 FGSM[8]、BIM[18]、MI-FGSM[9]、PGD[17]、C&W[19]、L-BFGS[6]、JSMA[10]、DeepFool[20]、UAP[15]、Boundary[12]、ZOO[11]、AGAN[21]、AUNA[21]、SPNA[21] 共 14 种攻击方法, 攻击调用 foolbox[21] 的函数, 参数默认.

5) 防御方法: 实验选择了 8 种防御方法作为对比算法, 分别是 resize[30]、rotate[30]、Distillation Defense (Distil-D)[41]、Ensemble Defense (Ens-D)[54]、Defense GAN (D-GAN)[50]、添加 Gaussian 噪声 (GN)、DAE[37] 和 APE-GAN[52]. 为了使对比实验更全面, 选取的对比算法包含了重识别防御的 3 类防御方法, 其中 resize、rotate 和 GN 是基于数据预处理的防御; Distil-D 是基于网络修正的防御; Ens-D、D-GAN、DAE 和 APE-GAN 是基于附加网络的防御. 以下对参数进行具体说明, 其中选定的缩放尺寸和旋转角度参数都是经过多次修改测试, 挑选出最优的参数.

a) resize1: 对于 MNIST 和 FMNIST, 首先将样本缩小为 6×6, 再放大回 28×28. 对于 CIFAR-10,

首先将样本缩小为 16 × 16, 再放大回 32 × 32; 对于 ImageNet, 首先将样本缩小为 128 × 128, 再放大回 224 × 224.

b) resize2: 对于 MNIST 和 FMNIST, 首先将样本放大为 32 × 32, 再缩小回 28 × 28; 对于 CIFAR-10, 首先将样本放大为 56 × 56, 再缩小回 32 × 32; 对于 ImageNet, 首先将样本放大为 512 × 512, 再缩小回 224 × 224.

c) rotate: 对于 MNIST、FMNIST、CIFAR-10 和 ImageNet 数据集, 首先将样本顺时针旋转 45°, 再逆时针旋转 45°.

d) Distil-D: 对于 MNIST、FMNIST 和 CIFAR-10 数据集, 蒸馏训练 epoch 设置为 20, 批尺寸为 64, 学习率为 0.001, 优化器为 Adam; 对于 ImageNet 数据集, 蒸馏训练 epoch 设置为 50, 批尺寸为 16, 学习率为 0.0001, 优化器为 Adam.

e) Ens-D: 对于 MNIST、FMNIST, 集成 3 种模型: AlexNet、LeNet 和 M_CNN; 对于 CIFAR-10 和 ImageNet, 集成 3 种模型: AlexNet、VGG16 和 VGG19.

f) D-GAN: 对于 MNIST、FMNIST, 训练生成式对抗网络的参数: epoch 设置为 10, 批尺寸为 32, 学习率为 0.001, 优化器为 Adam; 对于 CIFAR-10, 生成式对抗网络的参数: epoch 设置为 30, 批尺寸为 32, 学习率为 0.001, 优化器为 Adam; 对于 ImageNet, 训练生成式对抗网络的参数: epoch 设置为 50, 批尺寸为 16, 学习率为 0.001, 优化器为 Adam.

g) GN: 在样本上添加均值为 0、方差为 1 的随机高斯噪声, 作为 UIP 的对照, 说明 UIP 具有一定的规律.

h) DAE: 对于 MNIST、FMNIST, 训练编码器和解码器的参数: epoch 设置为 10, 批尺寸为 64, 学习率为 0.001, 优化器为 Adam; 对于 CIFAR-10, 训练编码器和解码器的参数: epoch 设置为 20, 批尺寸为 64, 学习率为 0.001, 优化器为 Adam; 对于 ImageNet, 训练编码器和解码器的参数: epoch 设置为 50, 批尺寸为 32, 学习率为 0.001, 优化器为 Adam.

i) APE-GAN: 对于 MNIST、FMNIST, 训练生成式对抗网络的参数: epoch 设置为 20, 批尺寸为 32, 学习率为 0.001, 优化器为 Adam; 对于 CIFAR-10, 训练生成式对抗网络的参数: epoch 设置为 40, 批尺寸为 32, 学习率为 0.001, 优化器为 Adam; 对于 ImageNet, 训练生成式对抗网络的参数: epoch 设置为 50, 批尺寸为 16, 学习率为 0.001, 优化器为 Adam.

6) 评价指标: 本文采用分类准确率 (Accuracy, ACC)、攻击成功率 (Attack success rate, ASR)、防御成功率 (Defense success rate, DSR) 和相对置信度变化 (Rconf) 来评价 UIPD. 具体为

$$\begin{cases} ACC = \dfrac{n^{\text{right}}}{N} \\ ASR = \dfrac{n_{\text{adv}}}{n^{\text{right}}} \\ DSR = \dfrac{n_{\text{adv}}^{\text{right}}}{n_{\text{adv}}} \end{cases} \tag{12}$$

其中, N 表示待分类的良性样本数, n^{right} 表示分类正确的良性样本数, n_{adv} 表示攻击成功的对抗样本数, 即成功被深度模型错误识别的样本数量, $n_{\text{adv}}^{\text{right}}$ 表示防御后重新分类正确的对抗样本数量.

$$\begin{aligned} Rconf = &(confD(l_{\text{true}}) - confA(l_{\text{true}})) + \\ &(confA(l_{\text{adv}}) - confD(l_{\text{adv}})) \end{aligned} \tag{13}$$

其中, $confD(l_{\text{true}})$ 表示防御后真实类标的预测置信度, $confA(l_{\text{true}})$ 表示攻击后真实类标的预测置信度, $confA(l_{\text{adv}})$ 表示攻击后对抗类标的预测置信度, $confD(l_{\text{adv}})$ 表示防御后对抗类标的预测置信度.

7) 实验步骤: 首先, 如图 1 所示, 通过良性样本的特征空间与深度学习模型的损失进行迭代训练, 生成通用逆扰动, 具体算法如算法 1 所示: 输入包括良性样本集 X, 分类器 $f(x)$, 逆扰动步长 ε^{uip} 和最大 epoch 数 N, 接着初始化通用逆扰动 ρ^{uip}, 利用良性样本集的样本特征和标签对通用逆扰动进行迭代训练, 训练完成后得到通用逆扰动. 随后, 在不同的攻击算法下针对深度模型分类器 $f(x)$ 生成各类型的对抗样本. 最后, 训练得到的通用逆扰动添加到对抗样本中, 完成识别防御.

3.2 UIPD 的攻击方法通用性

本文主要验证了同一个数据集和模型的 UIP 在不同攻击方法下的通用性. 具体实验结果如表 2 所示, 实验中采用 DSR 来衡量 UIPD 方法对不同攻击的防御有效性.

由表 2 可知, 在 MNIST、FMNIST 和 CIFAR-10 这三个小数据集上, 每个模型训练得到的 UIP 在不同攻击方法下都能达到 50% 以上的防御成功率, 大部分情况下能达到 70% 以上. 对于 ImageNet 大数据集, 通用逆扰动防御在不同攻击方法下的防御成功率也能达到 30% 以上. UIP 对不同攻击方法的防御能力在小数据集上普遍优于大数据集, 这是因为小数据集的图像尺寸小, 所包含的特征信息也远小于 ImageNet 大数据集中的图像, 所以训练 UIP 时更容易收敛, 而且包含的非鲁棒性特征更加全面, 导致 UIP 的防御效果更优.

表 2 UIPD 针对不同攻击方法的防御成功率 (%)

Table 2 The defense success rate of UIPD against different attack methods (%)

DSR	MNIST			FMNIST		CIFAR-10	ImageNet
	AlexNet	LeNet	M_CNN	AlexNet	F_CNN	VGG19	VGG19
良性样本识别准确率	92.34	95.71	90.45	89.01	87.42	79.55	89.00
FGSM[8]	73.31	85.21	77.35	79.15	80.05	78.13	43.61
BIM[18]	**99.30**	93.73	99.11	**95.28**	**97.61**	**85.32**	**72.90**
MI-FGSM[9]	69.65	90.32	98.99	88.35	85.75	56.93	44.76
PGD[17]	99.31	95.93	**99.19**	97.80	95.83	81.05	73.13
C&W[19]	99.34	96.04	92.10	96.44	94.44	80.67	46.67
L-BFGS[6]	98.58	70.12	67.79	66.35	71.75	68.69	31.36
JSMA[10]	64.33	55.59	76.61	72.31	69.51	60.04	37.54
DeepFool[20]	98.98	**97.98**	94.52	93.54	91.63	83.13	62.54
UAP[15]	97.46	97.09	99.39	97.85	96.55	83.07	72.66
Boundary[12]	93.63	94.38	95.72	92.67	91.88	76.21	68.45
ZOO[11]	77.38	75.43	76.39	68.36	65.42	61.58	54.18
AGNA[21]	75.69	76.40	81.60	64.80	72.14	62.10	55.70
AUNA[21]	74.20	73.65	78.53	65.75	62.20	62.70	52.40
SPNA[21]	92.10	88.35	89.17	77.58	74.26	72.90	60.30

除此之外, 还可以观察到, 同一个 UIP 虽然对不同的攻击方法都有效果, 但是防御效果在不同攻击方法上也是有差异的. 同一个 UIP 在 DeepFool 和 PGD 上的防御效果明显优于 JSMA, 这是因为不同攻击方法生成的对抗扰动的大小和约束条件不同. DeepFool 和 PGD 要求扰动的 L_2 范数尽可能小, 这导致了虽然这些攻击方法生成的对抗样本更加隐蔽, 但对抗样本中包含的非鲁棒性特征更容易被 UIP 抵消, 所以防御效果更好. 但是 JSMA 的攻击中限制扰动的个数而不限制单个像素点的扰动大小, 攻击时一旦发现非鲁棒性特征的像素点, 就会改变很大的像素值去激活非鲁棒性特征, 所以 UIP 很难完全抵消被激活的非鲁棒性特征, 这就导致了防御效果更差一点. 基于优化的攻击通过不断优化对抗扰动, 生成扰动较小但攻击性强的对抗样本, 因此, UIPD 在针对基于优化的攻击上的防御效果普遍低于基于梯度的攻击.

在式 (11) 的基础上, 使用最优化观点看待 UIP 的防御过程, 具体为

$$\rho(\Delta x)=\min\left\{\mathrm{E}_{(x,\,y)\sim D}\left[\min_{\Delta x^{\mathrm{uip}}\in S_x} L\left(x+\Delta x^{\mathrm{uip}},\,y\right)\right]\right\} \tag{14}$$

其中, $\rho(\cdot)$ 是需要最小化的优化目标, x 表示输入, y 表示样本标签, $\mathrm{E}_{(x,\,y)\sim D}[\cdot]$ 表示平均损失, $D(x,\,y)$ 表示输入和标签所在的联合概率分布, Δx^{uip} 表示通用逆扰动, $L(\cdot,\,\cdot,\,\cdot)$ 表示损失函数. 上述建模中 $\Delta x^{\mathrm{uip}}\in S_x$, 即此时扰动落在 S_x 范围内都是安全的, 因此将 S_x 称为安全边界. UIP 使用梯度下降的优化算法进行迭代训练, 在已训练好的模型基础上进一步朝着损失函数下降的方向进行 UIP 的扰动优化, 这一过程中能够提取更多的样本特征, 强化良性样本中的类相关特征, 使得样本向着类中心移动, UIP 的训练使用的是全局样本, 即训练集所有样本, 因此同一个全局 UIP 能够对不同类都能使用.

综合而言, UIP 在不同攻击方法上都有较好的防御效果.

3.3 UIPD 对数据样本的通用性

本节主要介绍 UIPD 方法在同一个模型和数据集上对所有样本数据的通用性. 表 3 展示了 UIPD 在 M_CNN 模型上、MNIST 数据集中不同样本的通用性 (更多模型上的数据集通用性展示见附录 A). 图 4 展示了 MNIST 数据集中不同模型的 UIP 可视化图 (更多数据集中不同模型的 UIP 可视化图见附录 A).

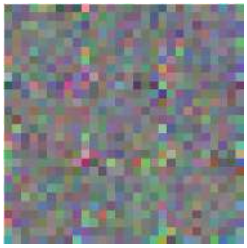
由表 3 的前两组数据可知, MNIST 数据集中 0 到 9 个良性样本在加上同一个 UIP 后, 类标和置信度都没有改变, 体现了 UIPD 在不损失良性样本分类准确率上的通用性. 表 3 的第 3 组表示分类错误的 0 ~ 9 个对抗样本. 由第 4 组可知, 在加上同一个 UIP 后, 9 张对抗样本都以较高的置信度重新正确分类, 这体现了 UIPD 在防御同一数据集中的对抗样本的通用性.

图 4 中的 UIP 可视化图由 python 中的 mat-

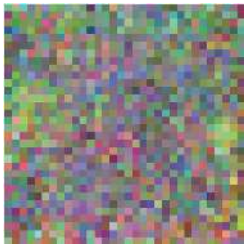
表 3　UIPD 针对不同数据样本的通用性 (MNIST, M_CNN)

Table 3　The universality of UIPD for different examples (MNIST, M_CNN)

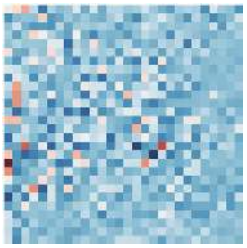
第 1 组		第 2 组		第 3 组		第 4 组	
良性样本类标	置信度	(良性样本 + UIP) 类标	置信度	对抗样本类标	置信度	(对抗样本 + UIP) 类标	置信度
0	1.000	0	1.000	5	0.5390	0	0.9804
1	1.000	1	1.000	8	0.4906	1	0.9848
2	1.000	2	1.000	1	0.5015	2	0.9841
3	1.000	3	1.000	7	0.5029	3	0.9549
4	1.000	4	1.000	9	0.5146	4	0.9761
5	1.000	5	1.000	3	0.5020	5	0.9442
6	1.000	6	1.000	4	0.5212	6	0.9760
7	1.000	7	1.000	3	0.5225	7	0.8960
8	1.000	8	1.000	6	0.5228	8	0.9420
9	1.000	9	1.000	7	0.5076	9	0.9796



(a) MNIST, AlexNet
均值: 0.0003, 方差: 0.0002
(a) MNIST, AlexNet
mean: 0.0003, variance: 0.0002



(b) MNIST, LeNet
均值: 0.0005, 方差: 0.0009
(b) MNIST, LeNet
mean: 0.0005, variance: 0.0009



(c) MNIST, M_CNN
均值: 0.0034, 方差: 0.0005
(c) MNIST, M_CNN
mean: 0.0034, variance: 0.0005

图 4　MNIST 数据集中不同模型的 UIP 可视化图

Fig. 4　The UIP visualization of MNIST dataset in different models

plotlib 库里面的 pyplot 以 rainbow 的涂色形式绘制, 像素值归一化到 [0, 1]. 由图 4 可知, 同一数据集下的不同模型的 UIP 都不相同, 但是 UIP 的均值和方差都很小, 所以图像加上 UIP 后的效果不影响人的视觉感受. 由式 (5) 和式 (6) 可知, UIP 通过对样本进行更新生成的过程是对抗样本生成的逆过程, 对抗样本与 UIP 的生成过程都是通过样本反馈到损失函数, 进而完成对模型预测输出的影响, 不同之处在于, 对抗样本生成扰动的方向是损失函数增大的方向, 而 UIP 生成扰动的方向是损失函数减小的方向, 因此, UIP 不仅不会对模型的预测准确率产生不良影响, 反而能够对模型分类精度有一定提升作用. 由算法 1 可知, 在 UIP 的迭代过程中, 输入深度模型分类器 $f(x)$ 是已经训练完成的收敛模型, 因此 UIP 在较小的逆扰动步长 ε^{uip} 下, 最终生成的 UIP 的扰动大小在较小范围内就能够使模型达到收敛.

3.4　不同防御方法的防御效果比较

在本节中, 本文主要比较了 UIPD 与其他防御方法针对不同模型、不同数据集, 采用不同攻击方法生成的对抗样本的防御效果. 具体实验结果如表 4 和表 5 所示, 其中表 4 是不同防御方法针对基于梯度的各种攻击方法的防御效果, 表 5 是不同防御方法针对基于优化的各种攻击方法的防御效果. 本文用 DSR 和 Rconf 来衡量不同防御方法之间的防御有效性. 表 4 和表 5 中的 DSR 均是两类攻击方法中不同攻击的平均防御成功率.

首先, 本文比较表 4 和表 5 中不同防御方法在不同模型和不同数据集下的 DSR. 在任意模型和数据集中, UIPD 的 DSR 均高于图像缩放、图像旋转、基于 GAN 的防御、基于自编码器的防御、高斯噪声、蒸馏防御和集成防御, 本文提出的 UIPD 不需要依赖大量的对抗样本, 也不改变模型的结构和训练量, 与这些同样不依赖对抗样本的对比算法相比, 本文提出的 UIPD 防御效果是最好的. 图像缩放和图像旋转这些简单的预处理操作也能对攻击起到较好的防御效果, 这间接说明了造成对抗攻击的非鲁棒性特征的脆弱性, 激活效果能够被 UIP 所抵消, 说

表 4 不同防御方法针对基于梯度的攻击的防御效果比较

Table 4 The performance comparison of different defense methods against gradient-based attacks

		MNIST			FMNIST		CIFAR-10	ImageNet
		AlexNet	LeNet	M_CNN	AlexNet	F_CNN	VGG19	VGG19
平均 ASR (%)		95.46	99.69	97.88	98.77	97.59	87.63	81.79
DSR (%)	resize1	78.24	74.32	81.82	79.84	77.24	69.38	47.83
	resize2	78.54	64.94	78.64	79.34	69.65	64.26	43.26
	rotate	76.66	80.54	84.74	77.63	61.46	72.49	42.49
	Distil-D	83.51	82.08	80.49	85.24	82.55	75.17	57.13
	Ens-D	87.19	88.03	85.24	87.71	83.21	77.46	58.34
	D-GAN	72.40	68.26	70.31	79.54	75.04	73.05	51.04
	GN	22.60	30.26	27.56	27.96	22.60	23.35	13.85
	DAE	84.54	85.25	85.68	86.94	80.21	75.85	59.31
	APE-GAN	83.40	80.71	82.36	84.10	79.45	72.15	57.88
	UIPD	**88.92**	**86.89**	**87.45**	**87.77**	**83.91**	**78.23**	**59.91**
Rconf	resize1	0.9231	0.9631	0.9424	0.8933	0.9384	0.6742	0.4442
	resize2	0.8931	0.9184	0.9642	**0.9731**	0.9473	0.7371	0.4341
	rotate	0.9042	0.8914	0.9274	0.9535	0.8144	0.6814	0.4152
	Distil-D	0.9221	0.9053	0.9162	0.9340	0.9278	0.6741	0.4528
	Ens-D	0.9623	0.9173	0.9686	0.9210	0.9331	0.7994	0.5029
	D-GAN	0.8739	0.8419	0.8829	0.9012	0.8981	0.7839	0.4290
	GN	0.1445	0.1742	0.2452	0.1631	0.1835	0.1255	0.0759
	DAE	0.9470	0.9346	0.9633	0.9420	0.9324	0.7782	0.5090
	APE-GAN	0.8964	0.9270	0.9425	0.8897	0.9015	0.6301	0.4749
	UIPD	**0.9788**	**0.9463**	**0.9842**	0.9642	**0.9531**	**0.8141**	**0.5141**

明了 UIPD 方法的防御可行性. 添加高斯随机噪声起到的防御效果微乎其微, 这体现了用训练的方法获得 UIP 的必要性. 此外, 小数据集的 ASR 和 DSR 均高于大规模的数据集, 这是由于大规模数据集图像所包含的特征信息远多于小数据集中的特征信息.

其次, 本文比较表 4 和表 5 中不同防御方法在不同模型和不同数据集中的 Rconf 指标. 在任意模型数据集下, UIPD 的 Rconf 均高于图像缩放、图像旋转、蒸馏防御、基于 GAN 的防御、基于自编码器的防御、高斯噪声和集成防御. 置信度变化越大, 表示防御后的对抗样本越鲁棒, 体现了防御的可靠性. 不同防御方法在不同模型数据集下的置信度变化与防御成功率保持高度的一致, 这显示了 UIPD 在防御成功率和防御可靠性上都有很好的表现. 由表 4 和表 5 可知集成防御的防御效果也优于其他防御, 但是集成防御需要训练多个模型, 训练代价更大, 所以相较之下, UIPD 方法是一个更好的防御选择.

3.5 不同防御方法对良性样本识别的影响

本节主要分析 UIPD 与其他防御方法对良性样本识别的准确率的影响. 具体实验结果如表 6 所示, 本文统计了不同数据集中的良性样本在不同防御方法下的分类准确率 (ACC).

由表 6 可知, 不同数据集的良性样本在 UIPD 防御和集成防御后分类准确率有了略微的上升, 但在其他防御方法防御后都有了一定程度的下降. 为了抵抗对抗攻击, 各种高性能的防御方法相继提出, 但是防御方法在提供防御有效性的同时, 会牺牲一定程度的良性样本分类精度. 然而 UIPD 防御后不仅没有损失良性样本的分类性能, 反而有略微的改善效果, 这得益于 UIPD 在训练时用良性样本作为训练数据集, 进一步的训练提升了原有的分类精度. 集成防御虽然同样能够提高分类准确率, 但是需要训练多个模型, 增大了训练成本.

3.6 参数敏感性和时间复杂度分析

在本节中, 主要对 UIPD 方法迭代步长的敏感性和时间复杂度进行分析.

图 5 展示了迭代步长敏感性实验结果, 横坐标表示训练 UIPD 的迭代步长, 纵坐标表示 UIPD 的防御成功率. 实验使用 MNIST 数据集, 目标模型

表 5　不同防御方法针对基于优化的攻击的防御效果比较

Table 5　The performance comparison of different defense methods against optimization-based attacks

		MNIST			FMNIST		CIFAR-10	ImageNet
		AlexNet	LeNet	M_CNN	AlexNet	F_CNN	VGG19	VGG19
平均 ASR (%)		93.28	96.32	94.65	95.20	93.58	88.10	83.39
DSR (%)	resize1	78.65	70.62	79.09	74.37	66.54	65.31	38.28
	resize2	63.14	67.94	77.14	66.98	63.09	62.63	41.60
	rotate	76.62	72.19	71.84	66.75	64.42	65.60	42.67
	Distil-D	82.37	82.22	80.49	82.47	83.28	71.14	45.39
	Ens-D	86.97	83.03	85.24	83.41	82.50	74.29	47.85
	D-GAN	82.43	80.34	86.13	79.35	80.47	70.08	43.10
	GN	20.16	21.80	25.30	19.67	18.63	21.40	13.56
	DAE	83.66	84.17	86.88	82.40	83.66	74.30	51.61
	APE-GAN	82.46	85.01	85.14	81.80	82.50	73.80	49.28
	UIPD	**87.92**	**85.22**	**87.54**	**83.70**	**83.91**	**75.38**	**52.91**
Rconf	resize1	0.8513	0.8614	0.8460	0.7963	0.8324	0.6010	0.3742
	resize2	0.7814	0.8810	0.8655	0.8290	0.8475	0.6320	0.3800
	rotate	0.8519	0.8374	0.8319	0.8100	0.8040	0.6462	0.4058
	Distil-D	0.9141	0.8913	0.9033	0.9135	0.9200	0.7821	0.4528
	Ens-D	0.9515	0.9280	0.8720	0.8940	0.9011	0.8155	0.4788
	D-GAN	0.8539	0.8789	0.8829	0.8733	0.8820	0.7450	0.4390
	GN	0.1630	0.1920	0.2152	0.1761	0.1971	0.1450	0.0619
	DAE	0.9120	0.9290	0.9510	0.9420	0.9324	0.7782	0.5090
	APE-GAN	0.8964	0.9270	0.9425	0.8897	0.9015	0.6301	0.4749
	UIPD	**0.9210**	**0.9340**	**0.9520**	**0.9512**	**0.9781**	**0.8051**	**0.5290**

表 6　不同防御方法处理后良性样本的识别准确率 (%)

Table 6　The accuracy of benign examples after processing by different defense methods (%)

	MNIST			FMNIST		CIFAR-10	ImageNet
	AlexNet	LeNet	M_CNN	AlexNet	F_CNN	VGG19	VGG19
良性样本	92.34	95.71	90.45	89.01	87.42	79.55	89.00
resize1	92.27 (−0.07)	95.66 (−0.05)	90.47 (+0.02)	88.97 (−0.04)	87.38 (−0.04)	79.49 (−0.06)	88.98 (−0.02)
resize2	92.26 (−0.80)	95.68 (−0.30)	90.29 (−0.16)	88.71 (−0.30)	87.38 (−0.04)	79.48 (−0.07)	87.61 (−1.39)
rotate	92.31 (−0.03)	95.68 (−0.03)	90.39 (−0.06)	88.95 (−0.06)	87.40 (0.02)	79.53 (−0.02)	88.82 (−0.18)
Distil-D	90.00 (−2.34)	95.70 (−0.01)	90.02 (−0.43)	88.89 (−0.12)	86.72 (−0.70)	76.97 (−2.58)	87.85 (−1.15)
Ens-D	**94.35 (+2.01)**	**96.15 (+0.44)**	**92.38 (+1.93)**	89.13 (+0.12)	87.45 (+0.03)	**80.13 (+0.58)**	89.05 (+0.05)
D-GAN	92.08 (−0.26)	95.18 (−0.53)	90.04 (−0.41)	88.60 (−0.41)	87.13 (−0.29)	78.80 (−0.75)	87.83 (−1.17)
GN	22.54 (−69.80)	25.31 (−70.40)	33.58 (−56.87)	35.71 (−53.30)	28.92 (−58.59)	23.65 (−55.90)	17.13 (−71.87)
DAE	91.57 (−0.77)	95.28 (−0.43)	89.91 (−0.54)	88.13 (−0.88)	86.80 (−0.62)	79.46 (−0.09)	87.10 (−1.90)
APE-GAN	92.30 (−0.04)	95.68 (−0.03)	90.42 (−0.03)	89.00 (−0.01)	87.28 (−0.14)	79.49 (−0.06)	88.88 (−0.12)
UIPD	92.37 (+0.03)	95.96 (+0.25)	90.51 (+0.06)	**89.15 (+0.14)**	**87.48 (+0.06)**	79.61 (+0.06)	**89.15 (+0.15)**

是 AlexNet. 本文选择 BIM、PGD、C&W、L-BFGS 和 DeepFool 五种方法进行敏感性实验. 从实验的结果可以看出, 当生成 UIP 的迭代步长变化时, UIPD 对于各攻击方法的防御成功率变化幅度都在 0.3% 以内. 实验结果表明, UIPD 是一种稳定的迭代训练方法, 当训练 UIPD 的迭代步长产生变化, 并不会影响最后 UIPD 的防御效果. 所以, UIPD 是一种稳健的防御方法, 具有一定的鲁棒性.

图 6 展示了不同防御方法实施 1 000 次防御的测试阶段时间消耗对比, 数据集是 MNIST, 采用的模型结构是 LeNet. 由图 6 可知, UIPD 所消耗的时间少于或十分接近其他的防御方法, 可知 UIPD 属

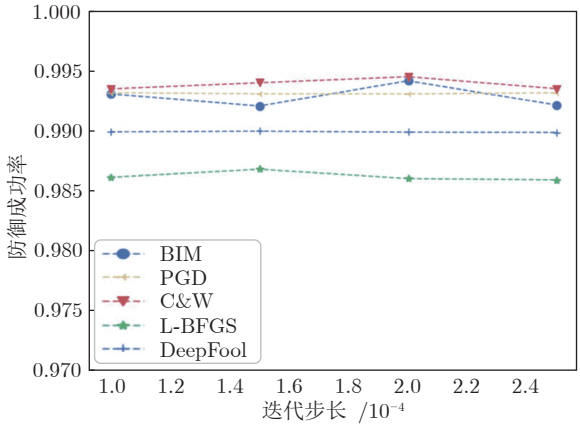


图 5 参数敏感性实验结果图

Fig. 5 The results of Parameter sensitivity experiment

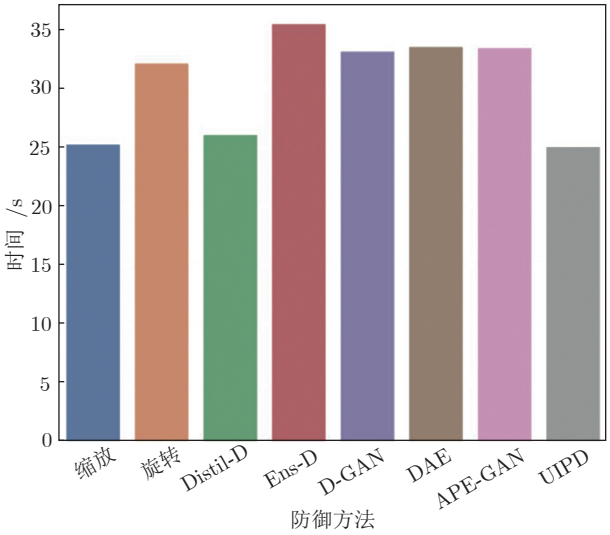


图 6 不同防御方法实施 1000 次防御的时间消耗

Fig. 6 The time cost in 1000 defenses of different defense methods

于时间复杂度低、防御速度快的一种对抗防御方法.

3.7 对抗补丁攻击的防御分析

在本节中, 主要对 UIPD 方法在基于对抗补丁的攻击下的防御进行分析.

图 7 是针对基于补丁的攻击的防御结果, 攻击方法是 Adversarial-Patch (AP)[26] 攻击, 在 AP 攻击后, 样本识别准确率大幅度下降, 可见基于补丁的对抗攻击是一种强大的攻击方法. UIPD 方法对于基于补丁的攻击有着一定的防御效果, 但是相比于基于扰动的防御效果而言, 性能略差. 这是由于基于扰动的对抗攻击生成的扰动是肉眼不可见的, 而基于补丁的攻击添加的扰动是肉眼可见的局部大范围补丁, 两者在扰动的量级上是存在明显差异的.

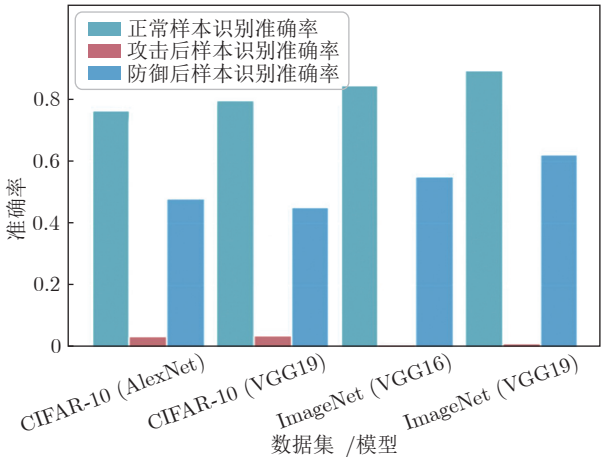


图 7 UIPD 对 AP 攻击的防御实验结果

Fig. 7 The results of UIPD against AP attacks

4 总结与展望

本文提出了一种基于通用逆扰动的对抗样本防御方法, 对数据样本、攻击方法都具有通用性. 在训练生成 UIP 的过程中, 只需要使用良性样本, 不需要任何关于对抗样本的先验知识, 即不依赖于对抗样本; UIP 会强化样本的类相关特征, 因此不会影响良性样本的识别, 甚至能够在一定范围内提升良性样本识别精度; UIP 的生成涉及到迭代步长的设置, 实验发现在一定范围内, 不同的迭代步长对 UIP 的防御效果几乎没有影响, 说明参数敏感性低; 在测试过程中, 只需要单个 UIP 叠加在任意待测试的样本上, 就能实现防御, 只需增加一个矩阵的“+”运算操作, 大大加快了防御速度. 因此, UIPD 方法防御对抗攻击是可行且高效的.

此外, 研究中也发现 UIPD 方法存在针对基于对抗补丁的攻击防御效果较差的问题, 这是由于基于对抗补丁的攻击生成的是局部大范围的扰动, UIP 无法完全抵消由对抗补丁带来的扰动干扰, 如何提升 UIP 对基于补丁的对抗攻击的防御效果, 是需要在后续工作中继续研究的. 同时, 研究中还发现 UIPD 方法虽然在数据样本上有较好的通用性, 但在模型间通用性不佳, 这是算法采用迭代优化造成的, 使得对模型具有较好的鲁棒性, 但是模型间泛化能力较差. 因此, 在未来的研究中, 将继续研究基于生成式对抗网络的通用逆扰动生成方法, 改善在模型间的通用性与泛化能力.

附录 A UIPD 不同数据样本的通用性举例和可视化

UIPD 方法在不同数据集上针对不同数据样本的通用性比较参见表 A1 ~ A3. 在不同数据集上, 不同模型的 UIP 可视化图见图 A1.

表 A1 UIPD 针对不同数据样本的通用性 (FMNIST, F_CNN)

Table A1 The universality of UIPD for different examples (FMNIST, F_CNN)

第 1 组		第 2 组		第 3 组		第 4 组	
良性样本类标	置信度	(良性样本 + UIP) 类标	置信度	对抗样本类标	置信度	(对抗样本 + UIP) 类标	置信度
0	1.000	0	1.000	6	0.4531	0	0.9415
1	1.000	1	1.000	3	0.4714	1	0.8945
2	1.000	2	1.000	6	0.5641	2	0.9131
3	1.000	3	1.000	1	0.5103	3	0.9425
4	1.000	4	1.000	2	0.4831	4	0.8773
5	1.000	5	1.000	7	0.5422	5	0.9026
6	1.000	6	1.000	5	0.4864	6	0.8787
7	1.000	7	1.000	5	0.5144	7	0.8309
8	1.000	8	1.000	4	0.4781	8	8.9424
9	1.000	9	1.000	7	0.4961	9	0.8872

表 A2 UIPD 针对不同数据样本的通用性 (CIFAR-10, VGG19)

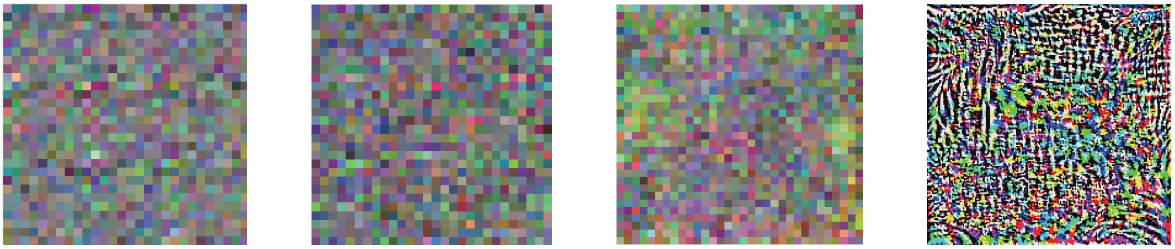
Table A2 The universality of UIPD for different examples (CIFAR-10, VGG19)

第 1 组		第 2 组		第 3 组		第 4 组	
良性样本类标	置信度	(良性样本 + UIP) 类标	置信度	对抗样本类标	置信度	(对抗样本 + UIP) 类标	置信度
飞机	1.000	飞机	1.000	船	0.4914	飞机	0.9331
汽车	1.000	汽车	1.000	卡车	0.5212	汽车	0.9131
鸟	1.000	鸟	1.000	马	0.5031	鸟	0.8913
猫	1.000	猫	1.000	狗	0.5041	猫	0.9043
鹿	1.000	鹿	1.000	鸟	0.5010	鹿	0.8831
狗	1.000	狗	1.000	马	0.5347	狗	0.9141
青蛙	1.000	青蛙	1.000	猫	0.5314	青蛙	0.8863
马	1.000	马	1.000	狗	0.4814	马	0.8947
船	1.000	船	1.000	飞机	0.5142	船	0.9251
卡车	1.000	卡车	1.000	飞机	0.4761	卡车	0.9529

表 A3 UIPD 针对不同数据样本的通用性 (ImageNet, VGG19)

Table A3 The universality of UIPD for different examples (ImageNet, VGG19)

第 1 组		第 2 组		第 3 组		第 4 组	
良性样本类标	置信度	(良性样本 + UIP) 类标	置信度	对抗样本类标	置信度	(对抗样本 + UIP) 类标	置信度
导弹	0.9425	导弹	0.9445	军装	0.5134	导弹	0.8942
步枪	0.9475	步枪	0.9525	航空母舰	0.4981	步枪	0.7342
军装	0.9825	军装	0.9925	防弹背心	0.5014	军装	0.8245
皮套	0.9652	皮套	0.9692	军装	0.4831	皮套	0.8074
航空母舰	0.9926	航空母舰	0.9926	灯塔	0.4788	航空母舰	0.8142
航天飞机	0.9652	航天飞机	0.9652	导弹	0.5101	航天飞机	0.7912
防弹背心	0.9256	防弹背心	0.9159	步枪	0.4698	防弹背心	0.8141
灯塔	0.9413	灯塔	0.9782	客机	0.5194	灯塔	0.7861
客机	0.9515	客机	0.9634	坦克	0.4983	客机	0.7134
坦克	0.9823	坦克	0.9782	灯塔	0.5310	坦克	0.7613



(a) FMNIST, AlexNet 均值: −0.0002, 方差: 0.0002

(b) FMNIST, F_CNN 均值: 0.0339, 方差: 0.0191

(c) CIFAR-10, VGG19 均值: −0.0005, 方差: 0.0003

(d) ImageNet, VGG19 均值: −0.0137, 方差: 0.0615

(a) FMNIST, AlexNet mean: −0.0002, variance: 0.0002

(b) FMNIST, F_CNN mean: 0.0339, variance: 0.0191

(c) CIFAR-10, VGG19 mean: −0.0005, variance: 0.0003

(d) ImageNet, VGG19 mean: −0.0137, variance: 0.0615

图 A1 不同数据集和模型的 UIP 可视化图

Fig. A1 The UIP visualization of different datasets and models

References

1 Goodfellow I, Bengio Y, Courville A. *Deep Learning*. Cambridge: The MIT Press, 2016. 24−45

2 Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: ACM, 2012. 1097−1105

3 Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks. In: Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal, Canada: ACM, 2014. 3104−3112

4 Yuan Wen-Hao, Sun Wen-Zhu, Xia Bin, Ou Shi-Feng. Improving speech enhancement in unseen noise using deep convolutional neural network. *Acta Automatica Sinica*, 2018, **44**(4): 751−759 (袁文浩, 孙文珠, 夏斌, 欧世峰. 利用深度卷积神经网络提高未知噪声下的语音增强性能. 自动化学报, 2018, **44**(4): 751−759)

5 Dai Wei, Chai Tian-You. Data-driven optimal operational control of complex grinding processes. *Acta Automatica Sinica*, 2014, **40**(9): 2005−2014
(代伟, 柴天佑. 数据驱动的复杂磨矿过程运行优化控制方法. 自动化学报, 2014, **40**(9): 2005−2014)

6 Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I J, et al. Intriguing properties of neural networks. In: Proceedings of the 2nd International Conference on Learning Representations. Banff, Canada: ICLR, 2014.

7 Akhtar N, Mian A. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 2018, **6**: 14410−14430

8 Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: Proceedings of the 3rd International Conference on Learning Representations. San Diego, USA: ICLR, 2015.

9 Dong Y P, Liao F Z, Pang T Y, Su H, Zhu J, Hu X L, et al. Boosting adversarial attacks with momentum. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 9185−9193

10 Papernot N, McDaniel P, Jha S, Fredrikson M, Celik Z B, Swami A. The limitations of deep learning in adversarial settings. In: Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P). Saarbruecken, Germany: IEEE, 2016. 372−387

11 Chen P Y, Zhang H, Sharma Y, Yi J F, Hsieh C J. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. Dallas, USA: ACM, 2017. 15−26

12 Brendel W, Rauber J, Bethge M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In: Proceedings of the 6th International Conference on Learning Representations. Vancouver, Canada: ICLR, 2018.

13 Xiao C W, Li B, Zhu J Y, He W, Liu M Y, Song D. Generating adversarial examples with adversarial networks. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm, Sweden: IJCAI, 2018. 3905−3911

14 Papernot N, McDaniel P, Goodfellow I. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples. arXiv preprint arXiv: 1605.07277, 2016.

15 Moosavi-Dezfooli S M, Fawzi A, Fawzi O, Frossard P. Universal adversarial perturbations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, USA: IEEE, 2017. 86−94

16 Ilyas A, Santurkar S, Tsipras D, Engstrom L, Tran B, Mądry A. Adversarial examples are not bugs, they are features. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver, Canada: ACM, 2019. Article No. 12

17 Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. In: Proceedings of the 6th International Conference on Learning Representations. Vancouver, Canada: ICLR, 2018.

18 Kurakin A, Goodfellow I, Bengio S. Adversarial examples in the physical world. In: Proceedings of the 5th International Conference on Learning Representations.Toulon, France: ICLR, 2017.

19 Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: Proceedings of the IEEE Symposium on Security and Privacy (SP). San Jose, USA: IEEE, 2017. 39−57

20 Moosavi-Dezfooli S M, Fawzi A, Frossard P. DeepFool: A simple and accurate method to fool deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, USA: IEEE, 2016. 2574−2582

21 Rauber J, Brendel W, Bethge M. Foolbox: A python toolbox to benchmark the robustness of machine learning models. arXiv preprint arXiv: 1707.04131, 2017.

22 Su J W, Vargas D V, Sakurai K. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 2019, **23**(5): 828−841

23 Baluja S, Fischer I. Adversarial transformation networks: Learning to generate adversarial examples. arXiv preprint arXiv: 1703.09387, 2017.

24 Cisse M, Adi Y, Neverova N, Keshet J. Houdini: Fooling deep structured prediction models. arXiv preprint arXiv: 1707.05373, 2017.

25 Sarkar S, Bansal A, Mahbub U, Chellappa R. UPSET and ANGRI: Breaking high performance image classifiers. arXiv preprint arXiv: 1707.01159, 2017.

26 Brown T B, Mané D, Roy A, Abadi M, Gilmer J. Adversarial patch. arXiv preprint arXiv: 1712.09665, 2017.

27 Karmon D, Zoran D, Goldberg Y. LaVAN: Localized and visible adversarial noise. In: Proceedings of the 35th International Conference on Machine Learning. Stockholm, Sweden: ICML, 2018. 2512−2520

28 Liu A S, Liu X L, Fan J X, Ma Y Q, Zhang A L, Xie H Y, et al. Perceptual-sensitive GAN for generating adversarial patches. In: Proceedings of the AAAI Conference on Artificial Intelligence. Honolulu, USA: AAAI, 2019. 1028−1035

29 Liu A S, Wang J K, Liu X L, Cao B W, Zhang C Z, Yu H. Bias-based universal adversarial patch attack for automatic checkout. In: Proceedings of the 16th European Conference on Computer Vision. Glasgow, UK: Springer, 2020. 395−410

30 Xie C H, Wang J Y, Zhang Z S, Zhou Y Y, Xie L X, Yuille A. Adversarial examples for semantic segmentation and object detection. In: Proceedings of the International Conference on Computer Vision (ICCV). Venice, Italy: IEEE, 2017. 1378−1387

31 Song C B, He K, Lin J D, Wang L W, Hopcroft J E. Robust local features for improving the generalization of adversarial training. In: Proceedings of the 8th International Conference on Learning Representations. Addis Ababa, Ethiopia: ICLR, 2020.

32 Miyato T, Dai A M, Goodfellow I J. Adversarial training methods for semi-supervised text classification. In: Proceedings of the 5th International Conference on Learning Representations. Toulon, France: ICLR, 2017.

33 Zheng S, Song Y, Leung T, Goodfellow I. Improving the robustness of deep neural networks via stability training. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, USA: IEEE, 2016. 4480−4488

34 Dziugaite G K, Ghahramani Z, Roy D M. A study of the effect of JPG compression on adversarial images. arXiv preprint arXiv: 1608.00853, 2016.

35 Das N, Shanbhogue M, Chen S T, Hohman F, Chen L, Kounavis M E, et al. Keeping the bad guys out: Protecting and vaccinating deep learning with JPEG compression. arXiv preprint arXiv: 1705.02900, 2017.

36 Luo Y, Boix X, Roig G, Poggio T, Zhao Q. Foveation-based mechanisms alleviate adversarial examples. arXiv preprint arXiv: 1511.06292, 2015.

37 Gu S X, Rigazio L. Towards deep neural network architectures robust to adversarial examples. In: Proceedings of the 3rd International Conference on Learning Representations. San Diego, USA: ICLR, 2015.

38 Rifai S, Vincent P, Muller X, Glorot X, Bengio Y. Contractive

auto-encoders: Explicit invariance during feature extraction. In: Proceedings of the 28th International Conference on International Conference on Machine Learning. Bellevue, USA: ACM, 2011. 833−840

39 Ross A S, Doshi-Velez F. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence. Menlo Park, CA, USA: AAAI, 2018. 1660−1669

40 Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv: 1503.02531, 2015.

41 Papernot N, McDaniel P, Wu X, Jha S, Swami A. Distillation as a defense to adversarial perturbations against deep neural networks. In: Proceedings of the IEEE Symposium on Security and Privacy (SP). San Jose, USA: IEEE, 2016. 582−597

42 Nayebi A, Ganguli S. Biologically inspired protection of deep networks from adversarial attacks. arXiv preprint arXiv: 1703.09202, 2017.

43 Cisse M, Adi Y, Neverova N, Keshet J. Houdini: Fooling deep structured visual and speech recognition models with adversarial examples. In: Proceedings of Advances in Neural Information Processing Systems. 2017.

44 Gao J, Wang B L, Lin Z M, Xu W L, Qi T J. DeepCloak: Masking deep neural network models for robustness against adversarial samples. In: Proceedings of the 5th International Conference on Learning Representations. Toulon, France: ICLR, 2017.

45 Jin J, Dundar A, Culurciello E. Robust convolutional neural networks under adversarial noise. arXiv preprint arXiv: 1511.06306, 2015.

46 Sun Z, Ozay M, Okatani T. HyperNetworks with statistical filtering for defending adversarial examples. arXiv preprint arXiv: 1711.01791, 2017.

47 Akhtar N, Liu J, Mian A. Defense against universal adversarial perturbations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 3389−3398

48 Hlihor P, Volpi R, Malagò L. Evaluating the robustness of defense mechanisms based on autoencoder reconstructions against Carlini-Wagner adversarial attacks. In: Proceedings of the 3rd Northern Lights Deep Learning Workshop. Tromsø, Norway: NLDL, 2020. 1−6

49 Kong Rui, Cai Jia-Chun, Huang Gang. Defense to adversarial attack with generative adversarial network. *Acta Automatica Sinica*, DOI: 10.16383/j.aas.c200033
(孔锐, 蔡佳纯, 黄钢. 基于生成对抗网络的对抗攻击防御模型. 自动化学报, DOI: 10.16383/j.aas.c200033)

50 Samangouei P, Kabkab M, Chellappa R. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In: Proceedings of the 6th International Conference on Learning Representations. Vancouver, Canada: ICLR, 2018.

51 Lin W A, Balaji Y, Samangouei P, Chellappa R. Invert and defend: Model-based approximate inversion of generative adversarial networks for secure inference. arXiv preprint arXiv: 1911.10291, 2019.

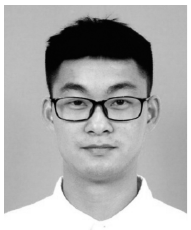
52 Jin G Q, Shen S W, Zhang D M, Dai F, Zhang Y D. APE-GAN: Adversarial perturbation elimination with GAN. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Brighton, UK: IEEE, 2019. 3842−3846

53 Xu W L, Evans D, Qi Y J. Feature squeezing: Detecting adversarial examples in deep neural networks. In: Proceedings of the 25th Annual Network and Distributed System Security Symposium. San Diego, USA: NDSS, 2018.

54 Ju C, Bibaut A, Van Der Laan M. The relative performance of ensemble methods with deep convolutional neural networks for image classification. *Journal of Applied Statistics*, 2018, **45**(15): 2800−2818

55 Kim B, Rudin C, Shah J. Latent Case Model: A Generative Approach for Case-Based Reasoning and Prototype Classification, MIT-CSAIL-TR-2014-011, MIT, Cambridge, USA, 2014.

陈晋音 浙江工业大学网络空间安全研究院和信息工程学院教授. 2009 年获得浙江工业大学博士学位. 主要研究方向为人工智能安全, 图数据挖掘和进化计算. 本文通信作者.
E-mail: chenjinyin@zjut.edu.cn
(**CHEN Jin-Yin** Professor at the Institute of Cyberspace Security and the College of Information Engineering, Zhejiang University of Technology. She received her Ph.D. degree from Zhejiang University of Technology in 2009. Her research interest covers artificial intelligence security, graph data mining, and evolutionary computing. Corresponding author of this paper.)



吴长安 浙江工业大学硕士研究生. 主要研究方向为深度学习, 计算机视觉, 对抗攻击和防御.
E-mail: wuchangan@zjut.edu.cn
(**WU Chang-An** Master student at the College of Information Engineering, Zhejiang University of Technology. His research interest covers deep learning, computer vision, adversarial attack and defense.)

郑海斌 浙江工业大学信息工程学院博士研究生. 主要研究方向为深度学习, 人工智能安全, 对抗攻击和防御, 图像识别.
E-mail: haibinzheng320@gmail.com
(**ZHENG Hai-Bin** Ph.D. candidate at the College of Information Engineering, Zhejiang University of Technology. His research interest covers deep learning, artificial intelligence security, adversarial attack and defense, and image recognition.)

王 巍 中国电子科技集团公司第三十六研究所研究员. 主要研究方向为无线通信分析, 网络安全.
E-mail: wwzwh@163.com
(**WANG Wei** Researcher at the 36th Research Institute of China Electronics Technology Group Corporation. His research interest covers wireless communication analysis and network security.)

温 浩 重庆中科云从科技有限公司高级工程师. 主要研究方向为量子通信, 计算机通信网络与大规模人工智能计算. E-mail: wenhao@cloudwalk.com
(**WEN Hao** Senior engineer at Chongqing Zhongke Yuncong Technology Co., Ltd.. His research interest covers guantum communication, computer communication networks, and large-scale artificial intelligence computing.)