

学 号： 2020211807

密 级： 公开

合肥工业大学

Hefei University of Technology

# 本科毕业设计（论文）

UNDERGRADUATE THESIS



类 型： 设计

题 目： 面向图像分类任务的  
对抗样本防御方法设计与实现

专业名称： 通信工程

入校年份： 2020 级

学生姓名： 尤量子

指导教师： 王昕 副教授

学院名称： 计算机与信息学院

完成时间： 2024 年 5 月

合 肥 工 业 大 学

本科毕业设计（论文）

面向图像分类任务的  
对抗样本防御方法设计与实现

学生姓名： 尤量子

学生学号： 2020211807

指导教师： 王昕 副教授

专业名称： 通信工程

学院名称： 计算机与信息学院

2024 年 5 月

**A Dissertation Submitted for the Degree of Bachelor**

**Design and Implementation of Adversarial Example  
Defense Methods for Image Classification**

By

You Liangzi

Hefei University of Technology


Hefei, Anhui, P.R.China

May, 2024

## 毕业设计（论文）独创性声明

本人郑重声明：所呈交的毕业设计（论文）是本人在指导教师指导下进行独立研究工作所取得的成果。据我所知，除了文中特别加以标注和致谢的内容外，设计（论文）中不包含其他人已经发表或撰写过的研究成果，也不包含为获得合肥工业大学或其他教育机构的学位或证书而使用过的材料。对本文成果做出贡献的个人和集体，本人已在设计（论文）中作了明确的说明，并表示谢意。


毕业设计（论文）中表达的观点纯属作者本人观点，与合肥工业大学无关。


毕业设计（论文）作者签名： 签名日期：2024 年 5 月 30 日

## 毕业设计（论文）版权使用授权书

本学位论文作者完全了解合肥工业大学有关保留、使用毕业设计（论文）的规定，即：除保密期内的涉密设计（论文）外，学校有权保存并向国家有关部门或机构送交设计（论文）的复印件和电子光盘，允许设计（论文）被查阅或借阅。本人授权合肥工业大学可以将本毕业设计（论文）的全部或部分内容编入有关数据库，允许采用影印、缩印或扫描等复制手段保存、汇编毕业设计（论文）。

（保密的毕业设计（论文）在解密后适用本授权书）

学位论文作者签名：

指导教师签名：

签名日期：2024 年 5 月 30 日

签名日期：2024 年 5 月 30 日

# 摘要

近年来,深度神经网络凭借其强大的数据处理能力和特征提取表达能力被广泛应用于多个领域,在深入探究的过程中,学者们发现深度神经网络同样面临着一系列不容忽视的安全隐患。他们发现,通过在图像上添加人类视觉系统难以辨识的微小扰动,就可以导致分类网络对图像的预测产生错误的结果。这种人为精心构造或者其他手段所生成的,并且导致深度学习模型以高置信度给出错误预测的样本就被称为对抗样本。对抗样本的存在对深度学习模型在多个关键安全领域的应用构成了严重威胁,成为了一个亟待解决的重要问题。

为了应对这一挑战,本文提出一种基于 GANomaly 的对抗样本防御模型对抗样本进行重建,利用 ResNet 的特征提取能力构造编码器,提取对抗样本中的显著特征,将对抗样本压缩为特征向量,并利用 Swin Transformer 的上下文建模能力,将原下采样部分改为上采样构造解码器,将特征向量重建为图片。同时利用干净样本的特征分布引导模型,学习干净样本特征分布的固有特性,让生成样本即重建后的对抗样本越来越接近于干净样本,最终使得对抗样本能被正确分类,达到防御对抗攻击的目的。

为验证防御模型的有效性,经过源模型训练、对抗样本数据集制作和防御模型训练后,设计三个实验测试本模型的防御性能。主体实验中对六种攻击方法进行防御测试,其中 MNIST 对抗样本数据集重建后的平均分类准确率达到 95.65%, CIFAR-10 对抗样本数据集重建后的平均分类准确率达到 79.83%,防御效果佳,可以有效防御多种对抗攻击。对比实验证明本防御模型的防御性能优于绝大多数现有的基于数据的防御方法。最后在泛化能力实验中,本模型的防御性能表现出了良好的泛化性。

**关键词:** 深度学习; 对抗防御; 图像重建; 生成对抗网络

# ABSTRACT

In recent years, deep neural networks have been widely applied in multiple fields due to their powerful data processing and feature extraction capabilities. However, researchers have discovered that deep neural networks also face a series of security risks that cannot be ignored. They found that by adding tiny perturbations or random noises to images, the classification network can be led to produce incorrect predictions for the images. These samples, which cause deep learning models to give wrong predictions with high confidence, are known as adversarial examples. The existence of adversarial examples poses a serious threat to the application of deep learning models in multiple critical security areas, becoming an important issue that needs to be addressed urgently.

To address this challenge, this thesis proposes an adversarial sample defense model based on GAN to reconstruct adversarial examples. It utilizes the feature extraction capability of ResNet to construct an encoder to extract salient features from adversarial examples, compresses the adversarial examples into features, and leverages the contextual modeling capabilities of Swin Transformer to replace the original downsampling part with upsampling to construct a decoder, reconstructing the features into images. Meanwhile, it utilizes the feature distribution of clean samples to guide the model, learning the inherent characteristics of the features of clean samples, so that the reconstructed adversarial examples become increasingly similar to clean samples, ultimately leading to correct classification.

To verify the effectiveness of the defense model, three experiments were designed to test the defense performance of this model. In the main experiment, six attack methods were tested for defense, achieving an average classification accuracy of 95.65% for the reconstructed MNIST adversarial example dataset and 79.83% for the reconstructed CIFAR-10 adversarial example dataset, demonstrating excellent defense effectiveness against multiple adversarial attacks. Comparative experiments proved that the defense performance of this defense model is superior to other preprocessing-based defense methods. Finally, in the generalization ability experiment, the defense performance of this model showed good generalization.

**KEYWORDS:** Deep Learning; Adversarial Defense; Image Reconstruction;  
Generative Adversarial Network

# 目录

<b>1 绪论.....</b>	<b>1</b>
1.1 研究背景与意义.....	1
1.2 国内外研究现状.....	1
1.2.1 对抗样本攻击研究现状.....	2
1.2.2 对抗样本防御研究现状.....	3
1.2.3 研究存在的问题.....	4
1.3 本文研究内容.....	4
1.4 本文结构安排.....	5
<b>2 相关理论基础.....</b>	<b>6</b>
2.1 生成对抗网络.....	6
2.1.1 GAN.....	6
2.1.2 GANomaly.....	7
2.2 对抗样本概述.....	8
2.3 对抗样本攻击.....	8
2.3.1 FGSM.....	9
2.3.2 BIM.....	9
2.3.3 PGD.....	9
2.3.4 AutoAttack.....	10
2.4 对抗样本防御.....	11
2.4.1 特征压缩.....	11
2.4.2 ComDefend.....	11
2.4.3 随机化.....	11
2.4.4 像素偏转.....	12
2.4.5 图像重建.....	12
2.5 本章小结.....	12
<b>3 基于 GAN 的防御模型设计与实现.....</b>	<b>13</b>
3.1 问题描述.....	13
3.2 防御模型设计.....	13



3.2.1 模型框架结构及防御原理.....	13
3.2.2 生成网络.....	14
3.2.3 判别网络.....	19
3.3 损失函数.....	20
3.4 本章小结.....	21
<b>4 实验内容及结果分析.....</b>	<b>22</b>
4.1 实验设置.....	22
4.1.1 实验内容.....	22
4.1.2 实验环境.....	22
4.1.3 采取数据集.....	22
4.2 源模型准备.....	23
4.3 数据集准备.....	24
4.4 防御主体实验及结果分析.....	25
4.5 防御对比实验及结果分析.....	26
4.6 泛化能力实验及结果分析.....	27
4.7 样本展示.....	28
4.8 本章小结.....	29
<b>5 总结与展望.....</b>	<b>30</b>
<b>参考文献.....</b>	<b>32</b>
<b>致谢.....</b>	<b>35</b>

## 插图清单

图 2.1 GAN 模型结构 .....	6
图 2.2 GANomaly 模型结构 .....	7
图 3.1 防御模型框架和网络结构图 .....	13
图 3.2 编码器网络结构图 .....	15
图 3.3 解码器网络结构图 .....	17
图 3.4 Pixel Shuffle 具体过程图 .....	18
图 3.5 解码器每层输出参数图 .....	19
图 3.6 判别网络结构图 .....	19
图 4.1 MNIST 数据集样本展示 .....	29
图 4.2 CIFAR-10 数据集样本展示 .....	29

# 表格清单

表 3.1 编码器具体参数.....	16
表 3.2 判别网络具体参数.....	20
表 4.1 实验环境.....	22
表 4.2 分类器训练结果 (%) .....	23
表 4.3 对抗样本分类准确率 (%) .....	24
表 4.4 在 MNIST 测试集 1 上的防御效果 (%) .....	25
表 4.5 在 CIFAR-10 测试集 1 上的防御效果 (%) .....	26
表 4.6 ResNet50 模型上各防御方法的防御效果对比 (%) .....	27
表 4.7 在 MNIST 测试集 2 上的泛化防御效果 (%) .....	28
表 4.8 在 CIFAR-10 测试集 2 上的泛化防御效果 (%) .....	28

# 1 绪论

## 1.1 研究背景与意义

近年来,数据量的激增和计算机处理能力的迅猛提升推动了深度神经网络的蓬勃发展。深度神经网络以其卓越的特征提取和表达能力而备受关注,广泛应用于各个领域。在计算机视觉领域,深度神经网络取得了巨大突破,用于图像识别<sup>[1]</sup>、目标检测和图像生成等任务。通过对大规模数据集的训练,深度神经网络能够学习到图像中的纹理、形状和结构等特征,从而准确识别和分类图像中的对象。在自然语言处理<sup>[2]</sup>领域,深度神经网络广泛应用于机器翻译、情感分析和文本生成等任务。通过学习大量的文本数据,深度神经网络能够理解和生成自然语言,实现自动翻译、情感分析和之智能对话等功能。此外,深度神经网络还在自动驾驶<sup>[3]</sup>、人脸识别<sup>[4]</sup>、医学影像分析等领域中都展现出了巨大的潜力。

在深入探究的过程中,学者们发现深度神经网络同样面临着一系列不容忽视的安全隐患。这一发现对于确保神经网络系统的稳健性和安全性至关重要。2013年,Szegedy 等人<sup>[5]</sup>深入探究了微小扰动对神经网络稳健性的影响,他们发现,通过在图像上添加人类视觉系统难以辨识的微小扰动,并将这些被篡改的图像输入到分类器模型中,竟然能导致分类网络对图像的预测产生根本性的变化。这种人为精心构造或者其他手段所生成的,并且导致深度学习模型以高置信度给出错误预测的样本,我们称之为对抗样本。对抗样本的存在对深度学习模型在多个关键安全领域的应用构成了严重威胁,成为了一个亟待解决的重要问题。

目前,国内外的学者们针对对抗样本带来的问题研究了多种防御方案,并取得了显著的防御效果。在深入探索对抗攻击与对抗防御的交互关系中,我们发现这两者之间存在着紧密的相辅相成与持续博弈。随着对抗攻击的持续演进和升级,我们能够生成更为复杂且有效的对抗样本,从而推动防御方法的进一步发展和完善。为了应对这一挑战,本文提出一种基于 GAN 的对抗样本防御模型,利用干净样本的特征分布引导模型,提取对抗样本中的显著特征,并重建出干净的图像,使得该样本分类正确。

## 1.2 国内外研究现状

自对抗样本的概念被引入后,为深入探究其存在机理,学者们提出了多种假说<sup>[6]</sup>,涵盖了模型正则化不充分、输入维度过高等方面。学者们基于这些假说提

出了多样化的对抗样本攻击算法，旨在测试不同深度学习架构的鲁棒性。这些攻击算法的涌现不仅凸显了深度学习在多方面应用过程中面临的种种安全隐患，还推动了相关防御机制的研究。为了应对对抗样本带来的威胁，研究者们还依据其特性提出了多种防御策略。接下来，本章将详细阐述国内外在对抗样本攻击与防御领域的研究进展与现状。

### 1.2.1 对抗样本攻击研究现状

目前，根据攻击者是否可以访问目标模型的具体信息，将对抗攻击分为白盒攻击和黑盒攻击两部分，并在下文进行介绍。

#### （1）白盒攻击

白盒攻击是指攻击者可以访问目标模型的具体信息，包括网络结构、神经元参数和输入输出等。

2013 年，Szegedy 等人<sup>[5]</sup>发现通过在图像上添加人类视觉系统难以辨识的微小扰动或随机噪声，就可以导致分类网络对图像的预测产生错误的结果。他们提出了 L-BFGS 攻击方案，利用盒约束优化的概念生成对抗样本，通过迭代优化求解使模型误分类的最小扰动。

IJ Goodfellow 等人<sup>[7]</sup>认为高维空间中具有线性特性的局部区域，随着扰动的线性累积仍能产生对抗样本，由于线性部分的存在导致了深度神经网络易受对抗样本的影响。他们提出了 FGSM 攻击算法，使得扰动方向与输入空间的梯度方向一致，利用梯度下降快速生成对抗样本。

根据决策边界假说<sup>[8]</sup>，Moosavi-Dezfooli 等人<sup>[9]</sup>提出了 DeepFool 攻击方法，通过求解高维空间中距离当前样本点与其他类距离最小的决策边界的扰动问题生成对抗样本，从而用更微小的扰动得到更好的攻击效果。

2016 年，Kurakin 等人<sup>[10]</sup>提出了 I-FGSM 攻击算法，该算法在 FGSM 的基础上进行多步迭代，每次迭代都在上一次生成的对抗样本的基础上，因此防御效果要优于 FGSM。2017 年，Aleksander Madry 等人<sup>[11]</sup>又在 I-FGSM 的基础上提出了 PGD 攻击算法，一定程度上提高了攻击的成功率和迁移性。

#### （2）黑盒攻击

黑盒攻击则指攻击者无法访问目标模型的具体信息。

Papernot 等人<sup>[12]</sup>首次提出了一种在黑盒环境下实施对抗样本攻击的新方法，他们通过一个代理模型来近似目标模型，并在白盒设定下基于代理模型生成对抗样本，随后利用对抗样本跨模型决策边界的迁移能力，成功迁移至目标模型实施

黑盒攻击。

Chen 等人<sup>[13]</sup>提出了基于梯度估计的 ZOO 算法,该方法通过坐标下降法来迭代优化各个坐标或坐标子集。通过有限次数的梯度估计更新坐标值,使得攻击效果接近于白盒攻击。

Brendel 等人提出了一种基于决策边界的 BA 攻击,该算法无需依赖模型输出类别的置信度,避免训练代理模型所需的大量时间成本,且无需对梯度进行估计,而是从决策边界出发,逐步减少添加的扰动,以生成对抗样本。

可以看出,对抗样本的攻击算法在持续演进与升级,对抗攻击的效果也在不断增强,从而进一步推进了防御的研究。

### 1.2.2 对抗样本防御研究现状

目前,根据防御过程中是否改变分类器模型的训练方法和结构,可以将防御技术分为基于模型的防御和基于数据的防御。

#### (1) 基于模型的防御

基于模型的防御可以分为改变分类器模型的训练方法和改进分类器模型的结构两种手段。

改变训练方法的目的是提升模型的鲁棒性,常常通过增加训练数据的数量和多样性来实现。

Goodfellow 等人<sup>[7]</sup>在提出 FGSM 攻击方法的同时提出了对抗训练的概念,即在训练时将对抗样本加入训练样本,以此防御该对抗攻击算法生成的对抗样本,提高模型的鲁棒性,并减轻了模型的过拟合程度。为防御不同类型的对抗样本,Tramer 等人<sup>[14]</sup>提出了一种集成式对抗训练,在对抗训练过程中添加不同类型的对抗样本,进一步提高了防御模型的防御性能。尽管对抗训练可以有效防御对抗攻击,但训练过程中添加的训练样本无疑大量增加了计算成本,于是 Shafahi 等人<sup>[15]</sup>对对抗训练进行了改进,在训练时循环使用模型参数的梯度信息,以此减少计算梯度的次数,降低了计算成本。

改进模型结构的目的是降低模型对微小扰动的敏感性,常常通过修改网络层结构的手段来实现。

Gu 等人<sup>[16]</sup>提出了一种基于梯度隐藏的深度收缩网络,当目标模型的梯度信息被隐藏时,大多数攻击方法无法获取梯度信息从而大大提高了生成对抗样本的难度。该方法引入的压缩自编码器降低了目标模型对输入的梯度信息的敏感度,以此达到梯度隐藏的效果。同时,防御蒸馏<sup>[17]</sup>也属于基于梯度隐藏的防御,该方

法将已训练网络的数据迁移到未训练的网络，导致未训练网络在反向传播过程中梯度信息被混淆，以此达到梯度隐藏的目的。

## （2）基于数据的防御

基于数据的防御可以分为对抗样本检测和减少对抗性扰动两种手段。

对抗样本检测的目的是在对抗样本输入到分类模型之前就将其鉴别出来。

Hendrycks 等人<sup>[18]</sup>通过主成分分析发现 FGSM 攻击所生成的对抗样本的系数方差相较于干净样本显著增大，以此鉴别出对抗样本和干净样本的差别。Lu 等人<sup>[19]</sup>通过观察 ReLU 函数<sup>[20]</sup>在处理输入数据时的输出特性来鉴别对抗样本和干净样本。Frosst 等人<sup>[21]</sup>提出的胶囊网络用来计算输入样本的重建误差，对抗样本的重建误差显著大于干净样本并因此被鉴别出来。

对抗性检测通常只是将对抗样本检测并避免其输入分类模型，不会进行进一步的处理，而对抗性消除是消除对抗样本中的微小扰动，且不会改变输入样本的数量，以此提高分类准确率。

Xie 等人<sup>[22]</sup>认为对于简单攻击的对抗样本来说，随机调整图像大小或随机填充可以改变对抗样本的内部扰动结构，减少对抗性扰动对分类模型的影响，在不改变干净样本分类精度的前提下提高对抗样本的分类准确率。Liao 等人<sup>[23]</sup>提出的 HGD 防御模型利用对抗样本和干净样本输出的高层特征不同来训练降噪器，即减少对抗样本中的对抗性噪声，提高分类精度。Jia 等人<sup>[24]</sup>提出的 ComDefend 防御模型利用一个端到端的图像压缩方法来消除对抗样本中的对抗扰动。Meng 等人<sup>[25]</sup>提出的 MagNet 模型利用自编码器重建对抗样本，使其接近于干净样本的数据分布。Samangouei 等人<sup>[26]</sup>提出的 Defense-GAN 模型利用生成对抗网络重建对抗样本。

### 1.2.3 研究存在的问题

对抗攻击和防御两者相辅相成、持续博弈，目前对抗攻击的研究相对成熟，而对抗防御的研究则相对缓慢，因此随着对抗样本技术的不断革新和进步，现有的防御策略可能无法防御升级后的对抗样本的攻击。并且多数对抗防御的方法往往聚焦于特定的攻击类型，因此它们在面对多样化的攻击时普遍缺乏泛化性。

### 1.3 本文研究内容

针对目前对抗防御存在的问题，本文提出一种基于 GAN 的对抗样本防御模型，该模型利用干净样本的特征分布进行引导，提取对抗样本中的显著特征，并

重建出干净的图像，使得该样本分类正确。本模型无需改变分类器的训练方法和结构，仅在对抗样本进入分类器前消除或减少对抗性扰动，并且可以有效防御多种对抗攻击。本文主要研究内容如下：

（1）分析目前主流对抗攻击和防御算法的特点，设计 GAN 防御模型的框架结构，分析生成网络和判别网络的网络结构，阐述该模型可以有效防御多种对抗攻击的可能性。

（2）利用 ResNet-V2 模型的特征提取能力及 Swin-Transformer 模型的建模能力组成 GAN 模型，阐述该 GAN 模型防御对抗样本的原理及过程。

（3）本文采用 MNIST 和 CIFAR-10 两种数据集，选取 6 种对抗样本攻击算法进行主体实验，测试防御模型的防御性能。

（4）同其他防御模型进行对比实验，并测试本防御模型的泛化性，进一步验证本防御模型的防御性能。

#### 1.4 本文结构安排

本文一共包含五个章节，每个章节的具体内容如下：

第一章是绪论。首先阐述本文的研究背景及意义，然后分析目前对抗样本攻防技术的国内外研究现状，最后概述本文的研究内容和结构安排。

第二章是相关理论基础。主要介绍生成对抗网络、卷积神经网络的基本概念，并详细阐述对抗样本的定义、攻击算法以及防御算法，分析各种攻防方法的优点与不足，为后续展开的防御研究提供理论支撑。

第三章是基于 GAN 的防御模型设计与实现。主要详细阐述防御模型的框架结构设计、防御原理与过程、具体网络结构设计、优势、参数设置以及每层网络输入输出等。

第四章是实验内容及结果分析。主要介绍整个实验的实验流程与内容，首先进行分类器与对抗样本数据集的准备，然后利用本模型进行防御实验并进行结果分析，最后进行对比实验和泛化能力测试进一步验证本模型的防御性能。

第五章是总结与展望。主要是总结本文的研究内容，并对未来的研究工作进展进行展望。



## 2 相关理论基础

### 2.1 生成对抗网络

#### 2.1.1 GAN

2014 年, Goodfellow 等人<sup>[28]</sup>提出了生成对抗网络(Generative adversarial nets, GAN)的概念。GAN 作为一种无监督生成式模型, 由生成器 G 和判别器 D 构成, 模型结构如图 2.1 所示。

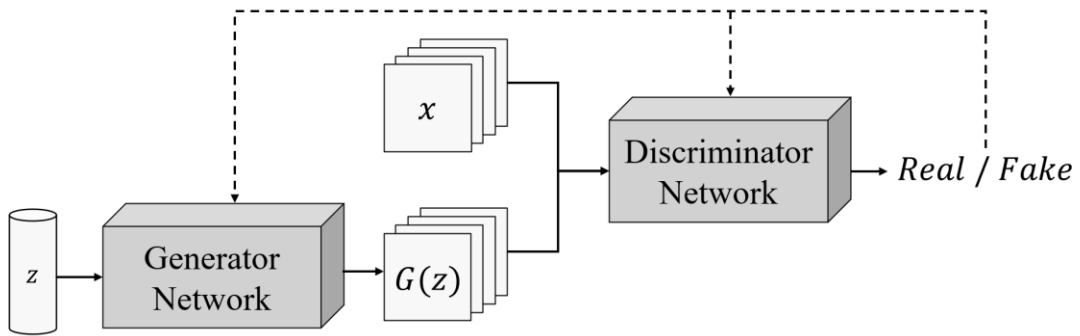


图 2.1 GAN 模型结构

生成器用于捕捉真实数据分布, 生成新的数据, 而判别器相当于一个二分类器, 判别输入数据是真实数据还是生成数据并计算概率值。在训练过程中, 整个框架类似于一个极大极小博弈游戏, 两者相互对抗学习, 最终达到纳什平衡, 即判别器无法将真实数据和生成数据区分开来。

GAN 的损失函数公式如下:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (2.1)$$

其中,  $x$  表示输入数据, 服从  $p_{data}(x)$  分布,  $z$  表示输入噪声, 服从  $p_z(z)$  分布。训练判别器时, 生成器参数不变, 并最大化目标函数  $V(D, G)$ , 使  $D(x)$  尽量接近于 1,  $D(G(z))$  尽量接近于 0; 训练生成器时, 判别器参数不变, 并最小化目标函数  $V(D, G)$ , 使  $D(G(z))$  尽量接近于 1, 即尽量使生成数据无法被判别器判别出来。在这种对抗博弈的过程中, 生成器不断生成无限接近于真实数据的生成数据, 判别器不断学习并尽量分辨出输入的数据是真实的还是生成器生成的。随着 GAN 在图像、语音视频等各方面的应用与发展, 其变体也在不断地被提出, 如 WGAN、CycleGAN、StyleGAN、GANomaly 等等, 接下来将介绍 GAN 在异常检测方面的应用, 即 GANomaly 模型的网络结构与检测流程。

### 2.1.2 GANomaly

GANomaly<sup>[27]</sup>是基于 GAN 的异常检测模型，由生成器网络和判别器网络组成，模型结构如图 2.2 所示。

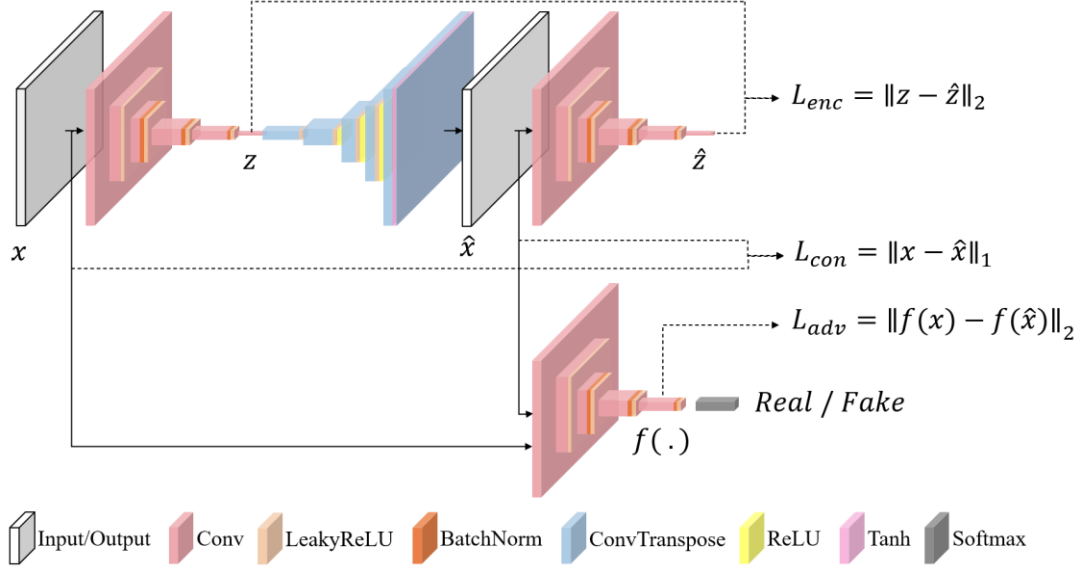


图 2.2 GANomaly 模型结构

生成器网络由编码器  $G_E(x)$ 、解码器  $G_D(z)$  和编码器  $E(x)$  三部分构成，前二者相当于一个自编码器，用于学习输入数据  $x$  的数据分布并重建图像  $\hat{x}$ ，其中  $G_E(x)$  输出的特征向量  $z$  代表了包含输入数据最好表征的最小维度。 $E(x)$  是一个与  $G_E(x)$  结构相同但参数不同的编码器，将重建图像  $\hat{x}$  压缩为特征向量  $\hat{z}$ ，用于后续损失的计算。判别器网络与 DCGAN 中的标准判别器网络相同，用于判别输入数据  $x$  和生成数据  $\hat{x}$  是真是假。

GANomaly 模型的损失函数由生成器损失函数和判别器损失函数组成。生成器损失函数又由对抗损失、上下文损失和编码器损失三个部分组成，判别器损失函数与原始 GAN 函数一样采用 BCE 损失。

其中，对抗损失使用特征对齐的损失函数，根据输入  $x$  选择判别器的中间层来计算对应层生成器的输出，使得输入数据  $x$  和生成数据  $\hat{x}$  ( $\hat{x} = G(x)$ ) 尽可能接近，即让生成器生成的图片更加逼真。对抗损失的公式如下：

$$L_{adv} = E_{x \sim p_x} \|f(x) - E_{x \sim p_x} f(G(x))\|_2 \quad (2.2)$$

上下文损失用于学习输入数据  $x$  中的上下文信息，目的仍是使输入数据  $x$  和生成数据  $\hat{x}$  ( $\hat{x} = G(x)$ ) 尽可能接近，其公式如下：

$$L_{con} = E_{x \sim p_x} \|x - G(x)\|_1 \quad (2.3)$$

编码器损失用于优化训练过程，使特征向量  $z$  ( $z = G_E(x)$ ) 与  $\hat{z}$  ( $\hat{z} = E(G(x))$ )

之间的距离更小，其公式如下：

$$L_{enc} = E_{x \sim p_x} ||G_E(x) - E(G(x))||_2 \quad (2.4)$$

而生成器的总损失由上述三种损失的加权和构成，其公式如下：

$$L = w_{adv}L_{adv} + w_{con}L_{con} + w_{enc}L_{enc} \quad (2.5)$$

在检测过程中，GANomaly 会对输入的数据进行评分，当评分结果大于一定阈值时，该输入样本就会被判定为异常样本，评分函数利用了编码器损失，用于计算输入样本经两次编码得到的特征向量之间的差距，其公式如下：

$$A(x) = ||G_E(x) - E(G(x))||_2 \quad (2.6)$$

## 2.2 对抗样本概述

2013 年，Szegedy 等人<sup>[5]</sup>首次提出了对抗样本的概念，他们发现，通过在原始样本上添加人类视觉系统难以辨识的微小扰动，就会使得基于 DNN 的分类器以高置信度给出错误的分类结果，而这种被人为精心构造或者其他手段所生成的样本就被称为对抗样本。

作者认为样本的真实决策边界和模型训练得到的真实边界无法完全重合，可能存在一定的盲区，盲区就是对抗样本可能存在的空间。作者将寻找这些盲区的过程表示为一个有界优化问题：

$$\text{Minimize } c|r| + \text{loss}_f(x + r, l) \quad \text{subject to } x + r \in [0,1]^m \quad (2.7)$$

其中， $r$ 表示扰动， $x + r$ 表示对抗样本， $x + r \in [0,1]^m$ 以盒约束来约束扰动的大小， $c$ 用于调节优化结果。

同时，扰动大小对生成对抗样本也具有一定的影响，并使用 $L_p$ 范数来度量，其定义如下：

$$||x||_p = \left( \sum_{i=1}^n |x|^p \right)^{\frac{1}{p}} \quad (2.8)$$

最为广泛使用的范数为 $L_0$ 、 $L_2$ 和 $L_\infty$ 范数， $L_0$ 范数度量对抗样本和原始样本之间的像素值变化， $L_2$ 范数度量对抗样本和原始样本之间的欧氏距离， $L_\infty$ 范数主要度量对抗样本与原始样本像素的最大差值，一般情况下基于 $L_\infty$ 范数的攻击算法攻击性最强，基于 $L_0$ 范数的攻击算法生成对抗样本扰动最小<sup>[29]</sup>。

## 2.3 对抗样本攻击

由于对抗攻击算法种类繁多且各有特点，本节主要详细介绍和课题相关的四种对抗攻击算法。

### 2.3.1 FGSM

IJ Goodfellow 等人<sup>[7]</sup>认为 DNN 由于使用了大量 ReLU 线性激活函数，使其具有线性特性的局部区域，该线性部分的存在使得 DNN 易受对抗样本的影响，并随着网络层数量呈递增趋势。因此他们提出了 FGSM 攻击算法利用梯度下降快速生成对抗样本，使扰动方向与输入空间的梯度方向一致，增大损失函数的值，导致分类结果出错。

FGSM 攻击可以进行无目标攻击和目标攻击。其中无目标攻击的公式如下：

$$x^{adv} = x + \varepsilon \text{sign}(\nabla_x J(\theta, x, y)) \quad (2.9)$$

目标攻击的公式如下：

$$x^{adv} = x - \varepsilon \text{sign}(\nabla_x J(\theta, x, y^{target})) \quad (2.10)$$

其中， $x$ 表示模型输入， $y$ 和 $y^{target}$ 表示结果标签， $\nabla_x J(\theta, x, y)$ 为输入被正确分类的损失函数的梯度， $\text{sign}(z)$ 作为符号函数限制扰动大小。

FGSM 算法的优点是攻击方法简单，单步攻击就可生成对抗样本，生成效率非常高，且生成的对抗样本具有较好的迁移攻击能力。缺点是只计算单次梯度造成攻击能力有限，并且对损失函数梯度变化有要求，在损失函数与模型输入呈线性的区域内 FGSM 攻击效果好，在非线性区间并不能保证攻击成功。

### 2.3.2 BIM

BIM 攻击也称为 I-FGSM 攻击，该算法作为 FGSM 的变体，将原来的单步攻击改为多步迭代，且每步迭代都基于上一步生成的对抗样本，相当于在同一大扰动范围下的多次小扰动 FGSM 攻击，公式如下：

$$x_0^{adv} = x, x_{n+1}^{adv} = \text{Clip}_{x, \varepsilon} \left\{ x_n^{adv} + \alpha \cdot \text{sign}(\nabla_x J(\theta, x_n^{adv}, y)) \right\} \quad (2.11)$$

其中， $n$ 表示迭代次数， $\alpha$ 表示每次迭代时的扰动大小， $x_n^{adv}$ 是第 $n$ 次生成的对抗样本。因此，在 BIM 攻击过程中，相当于将 FGSM 的单步攻击拆解为多步攻击，且每次攻击都要重新计算梯度方向，使攻击效果更好。

BIM 算法的优点是对抗样本攻击能力要强于 FGSM，但是对抗样本迁移攻击能力要弱于 FGSM。

### 2.3.3 PGD

Madry 等人<sup>[11]</sup>在 BIM 的基础上提出的 PGD 攻击算法同样是多步迭代攻击，但 PGD 攻击下第 $n$ 次生成的对抗样本需要重新映射回原始输入 $x$ 基于 $\varepsilon$ 的 $L_\infty$ 范数范围内，公式如下：

$$x_{n+1}^{adv} = Proj\{x_n^{adv} + \alpha \cdot sign(\nabla_x J(\theta, x_n^{adv}, y))\} \quad (2.12)$$

其中,  $Proj\{z\}$ 使得每次迭代生成的对抗样本都能落在约束条件中, 即设定的范围 $\epsilon$ , 并且 PGD 攻击添加了随机扰动进行初始化, 使对抗样本的起点随机化。

PGD 算法的优点是对抗样本攻击能力很强, 常用于评估模型鲁棒性, 缺点同样是迁移攻击能力较弱。

### 2.3.4 AutoAttack

2020 年, Francesco Croce 等人<sup>[30]</sup>提出了 AutoAttack 攻击算法, 该算法由 PGD 的两个变体 APGD-CE 和 APGD-DLR、FAB 攻击和 Square Attack 攻击组合而成。

当 APGD 使用 CE 损失时用 APGD-CE 表示, 当 APGD 使用 DLR 损失时用 APGD-DLR 损失表示。APGD 也叫 Auto-PGD, 该攻击方法唯一的自由参数是迭代次数, 其他参数都是自动调整的, 攻击能力强于 PGD 攻击。

假设 PGD 的攻击公式为:

$$x^{(k+1)} = P_S(x^{(k)} + \eta^{(k)} \nabla f(x^{(k)})) \quad (2.13)$$

则 APGD 的攻击公式为:

$$z^{(k+1)} = P_S(x^{(k)} + \eta^{(k)} \nabla f(x^{(k)})) \quad (2.14)$$

$$x^{(k+1)} = P_S(x^{(k)} + \alpha \cdot (z^{(k+1)} - x^{(k)}) + (1 - \alpha) \cdot (x^{(k)} - x^{(k-1)})) \quad (2.15)$$

其中,  $k$ 表示迭代次数,  $\eta^{(k)}$ 表示第 $k$ 次迭代时的步长, 默认损失为 CE 损失。

为确定步长的选择, 首先需要确定总的迭代次数 $N_{iter}$ , 并给定一些检查点  $w_0 = 0, w_1, \dots, w_n$ , 在每个检查点上按以下两个条件决定是否需要将当前步长减半:

$$\sum_{i=w_{j-1}}^{w_j-1} \mathbb{1}_{f(x^{(i+1)}) > f(x^{(i)})} < \rho \cdot (w_j - w_{j-1}) \quad (2.16)$$

$$\eta^{(w_{j-1})} \equiv \eta^{(w_j)} \text{ and } f_{\max}^{(w_{j-1})} \equiv f_{\max}^{(w_j)} \quad (2.17)$$

其中,  $f_{\max}^{(k)}$ 时前 $k$ 次迭代中找到的最高目标值, 若满足上述两个条件之一, 就对之后的迭代的学习率减半, 即 $\eta^{(k)} := \eta^{(w_j)}/2, \forall k = w_j + 1, \dots, w_{j+1}$ 。条件 1 是为了检查这一阶段的迭代是否有效, 条件 2 是为了检查这一阶段相较于之前的阶段是否提升。当学习率减半时, 需要判断何时减少步长, 即如何选择检查点 $w_j$ , 作者采用以下方案:

$$w_j = \lfloor p_j N_{iter} \rfloor \leq N_{iter} \quad (2.18)$$

$$p_{j+1} = p_j + \max\{p_j - p_{j-1} - 0.03, 0.06\}, p_j \in [0, 1] \quad (2.19)$$

其中  $p_0 = 0$ ,  $p_1 = 0.22$ 。

AutoAttack 算法能在所有测试模型中达到良好的攻击效果, 且不需要任何超参数调整, 并且具有较低的计算成本, 为对抗鲁棒性的评估提供了一种可靠的方法。

## 2.4 对抗样本防御

本节主要介绍和课题相关的五种对抗防御方法。

### 2.4.1 特征压缩

Xu 等人<sup>[31]</sup>提出的特征压缩方法常用于检测对抗样本。此方法基于一个核心观点: 往往输入样本在特征空间上存在过多的冗余信息, 而非所有特征都对分类结果产生决定性影响。攻击者在这些冗余空间上添加对抗扰动, 对该样本的分类精度造成了一定的影响。经特征压缩后的样本有效缩减了这些冗余特征空间, 也进一步减少了攻击者的攻击范围。

该方法通过颜色位深度缩减和特征平滑等技术来处理图像数据, 并比较压缩后的样本和原样本分类精度的差别来检测出对抗样本。是一种操作简单、低成本的对抗样本检测方法。

### 2.4.2 ComDefend

Jia 等人<sup>[25]</sup>提出了一种端到端的图像压缩重建方法来消除对抗样本中的对抗扰动——ComDefend。作者认为图像压缩会在保留图像显著性信息的同时去除含对抗扰动的冗余信息, 图像重建则将压缩后的信息还原成接近原始图像的形态。这种方法能够在不影响图像主体内容的情况下, 有效消除对抗样本中潜在的干扰因素, 从而提高图像分类或识别的准确性和鲁棒性。

ComDefend 防御模型需要预训练压缩模型和重建模型, 并同时更新训练参数。该方法能防御大部分的对攻击, 但会对干净样本的分类精度有所影响。

### 2.4.3 随机化

随机化如随机调整图像大小或随机填充可以破坏对抗样本的内部扰动结构, 从而减少对抗性扰动对分类模型的影响, 在不改变干净样本分类精度的前提下提高对抗样本的分类准确率。

该方法仅增加两个随机层, 计算成本较小, 且操作简单, 但无法防御较强的

攻击，因此常作为数据预处理与其他防御方法结合使用。

#### 2.4.4 像素偏转

像素偏转是指随机选取图像中的一些像素点来替换周围的像素，且像素偏转不会对干净样本的分类精度造成影响，而对于对抗样本来说，打乱像素能够使其分类正确。在像素偏转的基础上，作者使用了小波去噪法进一步消除了扰动噪声，提高了对抗样本的分类精度。

该方法也属于常作为图像预处理与其他防御方法结合使用，同样适用于防御简单攻击。

#### 2.4.5 图像重建

图像重建目的是将对抗样本重建为干净样本，在训练生成模型时会学习到干净样本的数据分布，从而将对抗样本映射到干净样本。Samangouei 等人<sup>[26]</sup>提出的 Defense-GAN 模型就是利用生成对抗网络学习了干净样本的潜在分布，从而重建对抗样本使其分类正确。

图像重建方法防御效果较好，但训练生成模型的数据量和计算量较大，且训练 GAN 网络任务较复杂，如果 GAN 模型没有经过充分的训练与精细的调整，将会影响该模型的防御性能。

### 2.5 本章小结

本章首先介绍了生成对抗网络的概念以及对抗博弈的过程，生成器和判别器进行对抗训练，生成器不断生成图片企图欺骗判别器，而判别器尽量将生成数据与真实数据区分开来，两者相互对抗相互学习，从而使生成数据无限接近于真实数据。其次介绍了 GAN 的变体 GANomaly 异常检测模型的网络结构、损失函数和检测流程，当输入样本经两次编码得到的潜在空间差距大于一定阈值时，就会被判别为异常样本。然后介绍了对抗样本的概念、可能存在的原因和扰动大小的对抗样本的影响。并详细介绍了与课题相关的四种对抗攻击算法，阐述了这些攻击方法的具体攻击方式即对抗样本的生成方法，并且比较了各种攻击算法的优缺点。最后介绍了五种基于数据的防御方法，阐述了各种方法的防御核心思路与内容以及对应的优缺点。

### 3 基于 GAN 的防御模型设计与实现

#### 3.1 问题描述

针对对抗样本带来的威胁以及目前对抗防御存在的问题，本文将实现一种可以有效应对多种对抗样本攻击的防御模型。

已知 GANomaly 作为基于 GAN 的异常检测模型，训练时只学习干净样本及其特征向量的数据分布，检测时计算原始样本和生成样本编码后的特征向量之间的差距，即潜在空间差距，往往对抗样本经两次编码后的潜在空间差距要大于干净样本，因此可以将对抗样本检测出来。本章设计的防御模型将采用同样的网络结构，但训练时输入对抗样本，并使对抗样本经两次编码后的潜在空间差距尽量小，学习干净样本的固有特性，促使对抗样本向干净样本的特征靠拢。在优化目标时输入干净样本，计算重建后的对抗样本与干净样本之间的对抗损失、上下文损失，促使重建后的对抗样本尽量接近干净样本，以此重建出干净的图像，最终使得对抗样本分类正确。

本章将详细阐述防御模型框架设计及防御原理、网络结构具体实现、模型优化目标和防御性能衡量指标。

#### 3.2 防御模型设计

##### 3.2.1 模型框架结构及防御原理

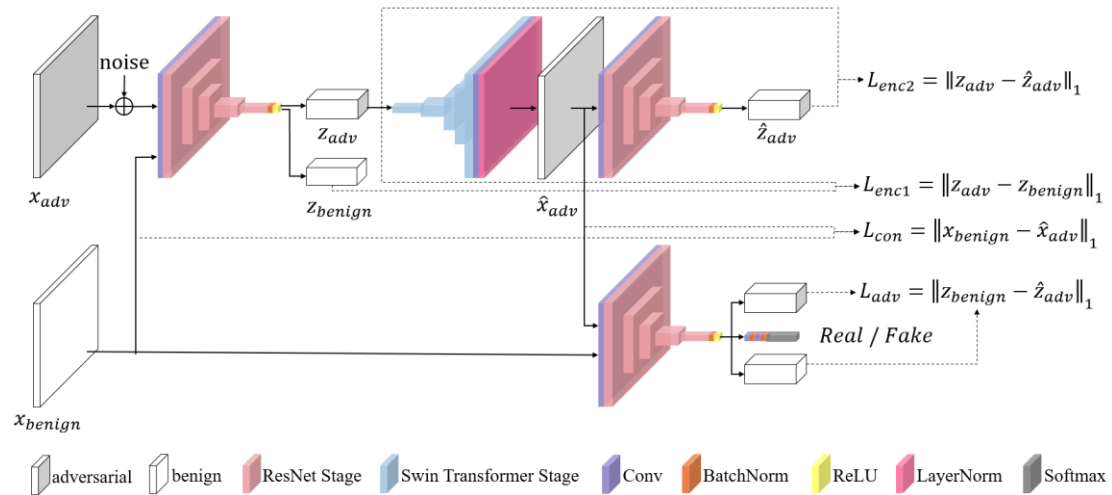


图 3.1 防御模型框架和网络结构图

本文提出的防御模型框架和网络结构如图 3.1 所示，一共包含以下两个网络：

- （1）生成网络：由编码器、解码器和编码器组成。其中每个编码器都具有



1 个卷积层 (Convolution, Conv), 4 个 ResNet Stage 和 1 个批归一化激活层 (Batch Normalization+Rectified Linear Unit, BN)。解码器含有 4 个 Swin Transformer Stage, 1 个卷积层和 1 个层归一化层 (Layer Normalization, LN)。

(2) 判别网络：由编码器和分类层组成。编码器和生成网络中的编码器结构相同，分类层具有 2 个卷积层，2 个批归一化层和 1 个 Sigmoid 激活函数，输出图片是真或假的二分类结果即判别概率。

训练生成器时固定判别器参数，首先在对抗样本  $x_{adv}$  上引入随机噪声并输入生成网络，随后编码器将其压缩为特征向量  $z_{adv}$ ，解码器将特征向量重建为生成样本  $\hat{x}_{adv}$ 。为了让判别器无法准确区分生成样本和干净样本，在计算生成网络的损失时需要引入干净样本作为参照。通过最小化干净样本与对抗样本、生成样本以及它们对应的特征向量之间的重建误差来不断优化生成器，使生成样本更加逼真，难以被判别器识别。

训练判别器时固定生成器参数，判别器的核心任务是求解一个二分类问题，即判断输入的样本是真实的干净样本还是由生成器产生的样本，并据此输出该样本属于干净样本的概率值。判别器的优化目标旨在最大程度地提高其对生成样本的识别能力，使干净样本和生成样本之间的差异更加明显，促使生成器生成的样本逐渐接近干净样本。

生成器和判别器通过这种对抗训练的方式相互竞争和协作，最终使得生成器生成的样本即重建后的对抗样本越来越接近于干净样本，从而实现对抗样本分类正确的目的。

在本防御模型中，无需改变分类器的训练方法和结构，也不会对攻击方法进行任何假设，仅在对抗样本进入分类器前将其重建为干净的图像，因此可以有效防御多种对抗攻击。

### 3.2.2 生成网络

上文提到，生成网络由编码器，解码器与编码器组成，本节将详细阐述编码器和解码器的网络结构和实现。

#### (1) 编码器

如图 3.2 所示，编码器含有 1 个 Conv 层，4 个 ResNetV2 Stage<sup>[32]</sup>和 1 个 BN 层和 1 个 ReLU 层。对于每个 ResNet Stage 都由一系列 ResNet Bottleneck 残差块组成，对输入张量进行归一化和降采样等操作，即进行特征提取和通道数转换，将输入分辨率为  $32*32$ ，通道数为 3 的张量转换为分辨率为  $4*4$ ，通道数为 256

的张量。通过引入 ResNetV2 残差块，使得信息在残差单元中更好的传递。

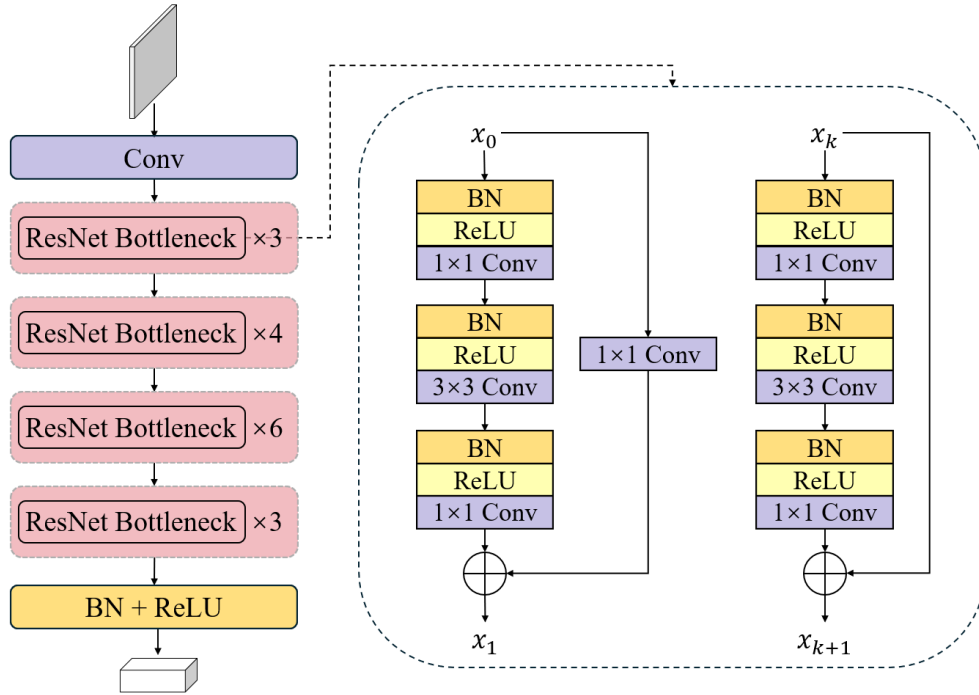


图 3.2 编码器网络结构图

在传统神经网络中，每一层的特征提取是指对输入数据进行非线性变换，随着网络层数加深，梯度在反向传播的过程中会逐层递减，容易造成梯度消失的问题，导致模型训练困难，本模型的编码器引入残差网络进行特征提取，残差连接通过将前一层的输出加入后一层的输入，使得后一层不仅包括前一层数据经非线性变换后得到的 $F(x)$ ，还包括了前一层的数据本身 $x$ ，使得梯度信息更好地进行反向传播，让模型训练变得更加容易。同时，本模型引入 BN 做批归一化处理，并使用 ReLU 作为激活函数，大大增加网络的非线性程度，可以很大程度上解决梯度消失和梯度爆炸的问题。

ResNet Bottleneck 残差块中存在两种映射，恒等映射和残差映射。在本模型中，残差映射作为残差部分对输入进行下采样，即调整通道数和特征图大小。对于每个 ResNet Stage 的首个 ResNet Bottleneck 残差块，残差映射对输入 $x_0$ 进行下采样，造成残差部分 $F(x_0)$ 和 $x_0$ 维度不同，无法直接相加，因此需要在恒等映射路径对 $x_0$ 进行 $1 \times 1$  Conv，使卷积后的 $x_0$ 和 $F(x_0)$ 维度一致，从而进行相加。

编码器每层网络的具体参数和输入输出如表 3.1 所示，可以大致分为扩大通道数、降采样、归一化激活三个部分。

表 3.1 编码器具体参数

Layer	Input size, channels	filter	output size, channels
Conv	$32 \times 32$ , 1 or 3	$3 \times 3$ , 64	$32 \times 32$ , 64
ResNet Stage	$32 \times 32$ , 64	$\begin{bmatrix} 1 \times 1, 8 \\ 3 \times 3, 8 \\ 1 \times 1, 32 \end{bmatrix} \times 3$	$32 \times 32$ , 32
ResNet Stage	$32 \times 32$ , 32	$\begin{bmatrix} 1 \times 1, 16 \\ 3 \times 3, 16 \\ 1 \times 1, 64 \end{bmatrix} \times 4$	$16 \times 16$ , 64
ResNet Stage	$16 \times 16$ , 64	$\begin{bmatrix} 1 \times 1, 32 \\ 3 \times 3, 32 \\ 1 \times 1, 128 \end{bmatrix} \times 6$	$8 \times 8$ , 128
ResNet Stage	$8 \times 8$ , 128	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$4 \times 4$ , 256
BN + ReLU	$4 \times 4$ , 256	—	$4 \times 4$ , 256

鉴于 CIFAR-10 数据集的图像分辨率为  $32 \times 32$ ，通道数为 3，MNIST 数据集的图像分辨率为  $28 \times 28$ ，通道数为 1，为确保后续降采样操作的顺畅性，我们需要对 MNIST 数据集中的图像进行预处理，将图像分辨率先插值扩大到  $32 \times 32$  后再进行编码。

在第一部分的卷积层，卷积核大小为  $3 \times 3$ ，数量为 64，步长为 1，边缘填充值为 1，主要作用是扩大图像通道数到 64，便于后续处理。在第二部分的降采样，只有第二、三、四层 ResNet Stage 中的首个卷积核大小为  $3 \times 3$  的卷积层，步长为 2，将特征图大小变为原来的一半，进行特征提取，其余部分的卷积层步长都为 1，通过卷积层数的增加，使网络学习到更复杂更高级的特征。且除第一层 ResNet Stage 将图像通道数变为原来的一半，其余层都将图像通道数扩大到原来的 2 倍。最终得到分辨率为  $4 \times 4$ ，通道数为 256 的特征图。第三部分确保输出具有恰当的范围和非线性特质，利于后续上采样。

## （2）解码器

如图 3.3 所示，解码器含有 4 个 Swin Transformer Stage<sup>[33]</sup>，1 个 Conv 层和 1 个 LN 层。前三个 Swin Transformer Stage 由多个 Swin Transformer Block 和一个 Patch Expanding 层组成，与原有的 Patch Merging 层不同，Patch Expanding 层

执行上采样操作，将低分辨率的图像转换为高分辨率的图像，实现图像重建的功能，最后一个 Swin Transformer Stage 只含有两个 Swin Transformer Block，不进行上采样操作，最终将分辨率为  $4 \times 4$ ，通道数为 256 的张量转换为分辨率为  $32 \times 32$ ，通道数为 3 或 1 的张量。通过引入 Swin Transformer 的自注意力机制和窗口化卷积操作，可以更好地处理大尺寸图像，并捕捉全局上下文信息。

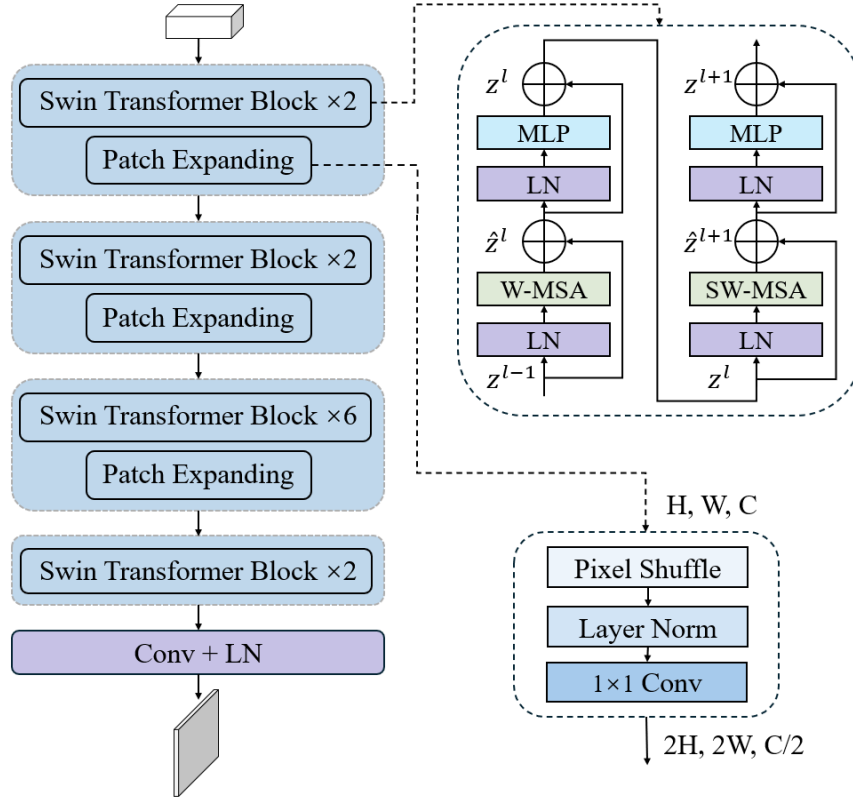


图 3.3 解码器网络结构图

Swin Transformer Block 主要是由 LayerNorm、MLP、Window Multi-head Self-Attention（W-MSA）和 Shifted Window Multi-head Self-Attention（SW-MSA）组成。其中 W-MSA 和 SW-MSA 是成对使用的，先使用 W-MSA 结构再使用 SW-MSA 结构，因此 Swin Transformer Block 的使用次数都为偶数。W-MSA 和 SW-MSA 分别表示使用规则和移位窗口分割配置的基于窗口的多头自注意力机制。由于普通的 MSA 对于特征图中的每个像素的 Self-Attention，需要和所有的像素去计算，而 Swin Transformer Block 引入的 W-MSA 是基于窗口的自注意力机制，会将特征图划分为一个个窗口，并在每个窗口内部进行自注意力计算。相比 MSA，W-MSA 大大减少了计算量，但具有不同窗口之间无法信息交互的缺点，因此需要配套使用 SW-MSA。SW-MSA 将窗口从左上角分别向右侧和下方进行偏移，使得某些在 W-MSA 中被划分为不同窗口的像素在 SW-MSA 中被划分到同一个

窗口内，从而让信息在不同的窗口之间进行传递。Swin Transformer Block 通过堆叠多个自注意力机制模块获得更大的感受野和更好的局部感知能力，可以广泛地捕获上下文信息，为后续对特征图进行上采样提供基础。

Patch Expanding 层可以将低分辨率的特征图上采样到高分辨率，并且减少特征维度，以逐渐恢复图像的细节和结构。本模型中，Patch Expanding 层可以分为像素重排（Pixel Shuffle）、层归一化（LayerNorm）和 $1 \times 1$  Conv 三个步骤。像素重排作为一种常用的上采样方法，将输入的特征图重塑为 2 倍上采样的大特征图，并相应地将特征维度减少到原始维度的 $1/4$ ，然后经 $1 \times 1$  Conv 将特征维度扩大 2 倍，最终将形状为 $[H, W, C]$ 的特征图重新排列成形状为 $[2H, 2W, C/2]$ 的特征图。

其中 Pixel Shuffle 的具体过程如图 3.4 所示， $r$  表示上采样的倍数，首先通过卷积得到 $r^2$ 个通道的特征图，特征图大小与输入的低分辨率图像一致，然后通过周期筛选，将每个特征图相同位置的像素提取出来作为最后输出的大特征图的一个 patch，遍历整个特征图，将得到 $H \times W$ 个 patch，将所有 patch 拼合在一起成为最后的大特征图，比如上图的 4 个特征图，将每个特征图的第 1 行第 1 列个像素拼合成一个含绿色、蓝色、黄色、红色的小 patch，作为大特征图的第 1 行第 1 列的 patch，如此遍历所有像素即可完成  $r$  倍上采样，即将形状为 $[H, W, C]$ 的特征图重新排列成形状为 $[H * r, W * r, C/r^2]$ 的特征图。

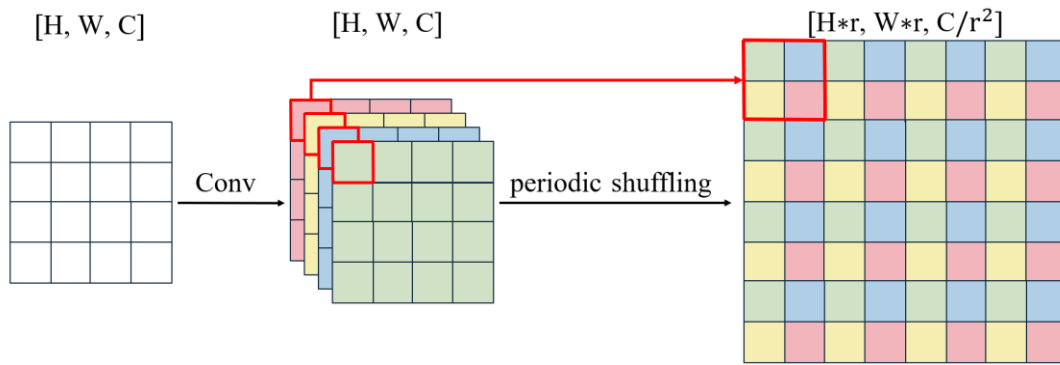


图 3.4 Pixel Shuffle 具体过程图

解码器每层网络的输入输出参数如图 3.5 所示。前三个 Swin Transformer Stage 对特征图进行上采样，其中 Pixel Shuffle 像素重排进行 2 倍上采样，将形状为 $[H, W, C]$ 的特征图重新排列成形状为 $[2H, 2W, C/4]$ 的特征图，并通过一个 $1 \times 1$  Conv 扩大特征图的数量，提供更多的信息通道。第四个 Stage 不进行上采样，主要用于加深网络层。最后将输出的特征图经卷积后输出，得到重建样本。

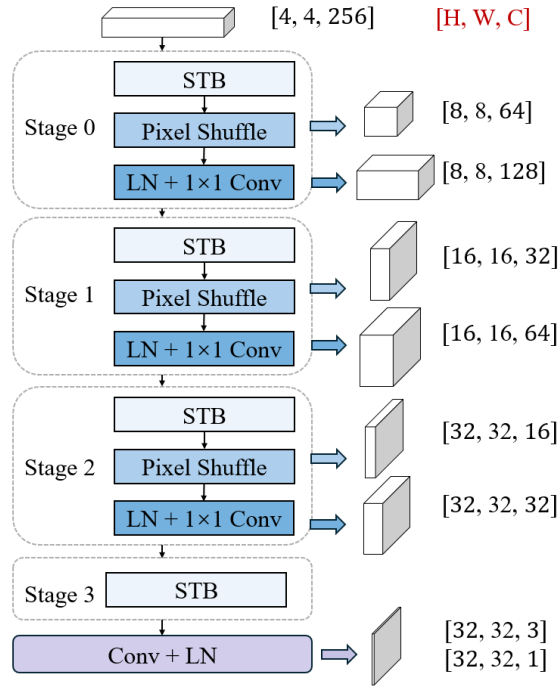


图 3.5 解码器每层输出参数图

### 3.2.3 判别网络

判别网络相当于一个二分类器，判别输入数据是真实数据还是生成数据并计算概率值。判别网络的结构如图 3.6 所示。

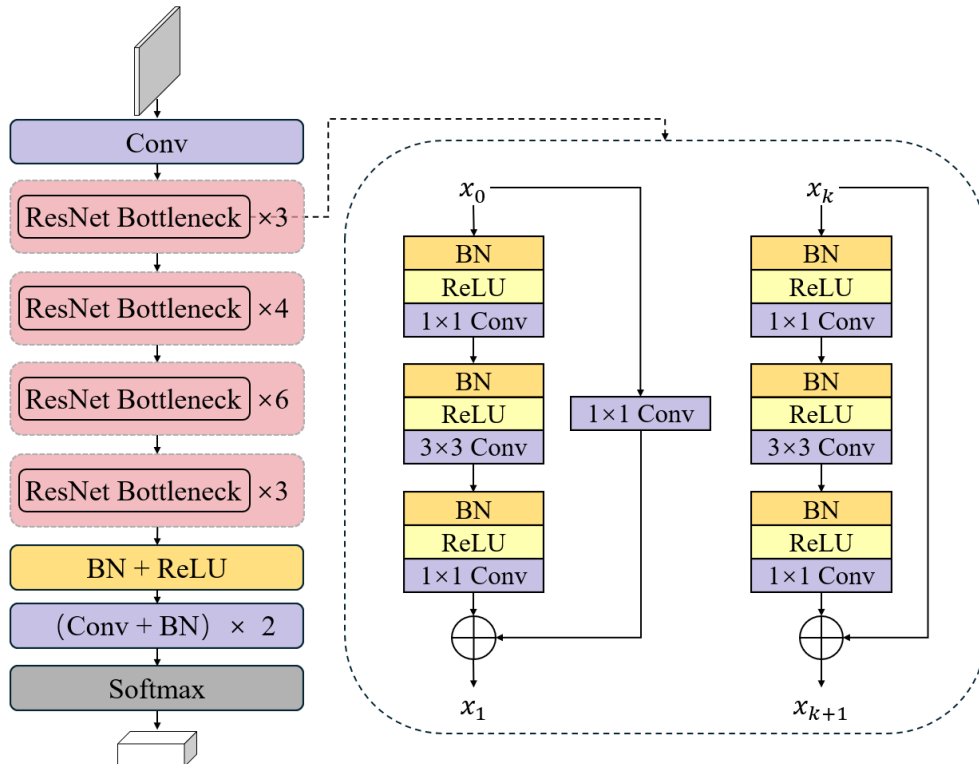


图 3.6 判别网络结构图

本文设计的判别网络的在编码网络的基础上添加了一个分类层，该分类层将预测输入样本为真实样本的可能性，并利用 softmax 函数输出一个 0 到 1 之间的概率值，表示该样本为干净样本的概率。对于判别网络每层的具体参数和输入输出如表 3.2 所示，所有 ResNet 部分参数都和编码器中的一样，在此不做赘述。

表 3.2 判别网络具体参数

Layer	Input size, channels	filter	output size, channels
Conv	$32 \times 32$ , 1 or 3	$3 \times 3$ , 64	$32 \times 32$ , 64
ResNet Stage	$32 \times 32$ , 64	$\begin{bmatrix} 1 \times 1, 8 \\ 3 \times 3, 8 \\ 1 \times 1, 32 \end{bmatrix} \times 3$	$32 \times 32$ , 32
ResNet Stage	$32 \times 32$ , 32	$\begin{bmatrix} 1 \times 1, 16 \\ 3 \times 3, 16 \\ 1 \times 1, 64 \end{bmatrix} \times 4$	$16 \times 16$ , 64
ResNet Stage	$16 \times 16$ , 64	$\begin{bmatrix} 1 \times 1, 32 \\ 3 \times 3, 32 \\ 1 \times 1, 128 \end{bmatrix} \times 6$	$8 \times 8$ , 128
ResNet Stage	$8 \times 8$ , 128	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$4 \times 4$ , 256
BN + ReLU	$4 \times 4$ , 256	—	$4 \times 4$ , 256
(Conv + BN) $\times$ 2	$4 \times 4$ , 256	$\begin{bmatrix} 3 \times 3, 64 \\ 4 \times 4, 1 \end{bmatrix}$	$1 \times 1$ , 1
Sigmoid			

### 3.3 损失函数

本模型提出的防御模型包含生成网络和判别网络，如图 3.1 所示，两个网络分别采用绝对误差损失（L1 Loss）和二元交叉熵损失（BCE Loss）。

生成网络损失由损失 $L_{adv}$ ， $L_{con}$ 和 $L_{enc}$ 组成。

其中， $L_{adv}$ 计算生成样本（ $\hat{x}_{adv}$ ）和干净样本（ $x_{benign}$ ）经判别网络编码后的特征向量（ $\hat{z}_{adv}$ 和 $z_{benign}$ ）之间的绝对误差，目的是使生成样本和干净样本的特征向量尽可能接近，即让生成样本尽量向干净样本靠拢，让生成器生成的图片更加逼真， $L_{adv}$ 损失的公式如下：

$$L_{adv} = E_{z \sim p(z)} ||\hat{z}_{adv} - z_{benign}||_1 \quad (3.1)$$

$L_{con}$  计算生成样本和干净样本之间的绝对误差，目的仍是使生成样本 ( $\hat{x}_{adv}$ ) 和干净样本 ( $x_{benign}$ ) 尽可能接近，其公式如下：

$$L_{con} = E_{x \sim p(x)} \|x_{benign} - \hat{x}_{adv}\|_1 \quad (3.2)$$

$L_{enc}$  由  $L_{enc1}$  和  $L_{enc2}$  共同组成，目的也是为了生成样本更加真实可信。 $L_{enc1}$  用于计算对抗样本 ( $x_{adv}$ ) 和干净样本 ( $x_{benign}$ ) 经生成网络编码后的特征向量 ( $z_{adv}$  和  $z_{benign}$ ) 之间的绝对误差，其公式如下：

$$L_{enc1} = E_{z \sim p(z)} \|z_{adv} - z_{benign}\|_1 \quad (3.3)$$

$L_{enc2}$  用于计算对抗样本和生成样本经生成网络编码后的特征向量之间的绝对误差，其目的在于最小化对抗样本经过两次编码后潜在空间差异。这一目标的背后逻辑在于，干净样本经两次编码后的特征向量差距小，而对抗样本往往呈现出较大的差异。若能有效减小对抗样本经两次编码后的特征向量差距，就意味着模型在某种程度上学习到了干净样本的固有特性，促使对抗样本向干净样本的特征靠拢，进而实现对对抗样本的重建和优化。该损失的公式如下：

$$L_{enc2} = E_{z \sim p(z)} \|z_{adv} - \hat{z}_{adv}\|_1 \quad (3.4)$$

$L_{enc}$  为  $L_{enc1}$  和  $L_{enc2}$  的均值，其公式如下：

$$L_{enc} = (L_{enc1} + L_{enc2}) * 0.5 \quad (3.5)$$

对于生成网络的总损失，就可以表示为：

$$L = w_{adv}L_{adv} + w_{con}L_{con} + w_{enc}L_{enc} \quad (3.6)$$

本模型使用的  $w_{adv} = 0.1$ ， $w_{con} = w_{enc} = 1$ 。

对于判别网络，其损失函数和 LS-GAN 的判别器损失相同，在此不多介绍。

### 3.4 本章小结

本章针对对抗样本带来的安全威胁和当前对抗防御策略中存在的问题，提出了一种基于 GANomaly 的对抗样本防御模型。首先详细介绍了模型框架结构、防御原理以及本模型可以有效防御多种对抗攻击的可能性。其次具体介绍了生成网络和判别网络的网络结构与实现，利用 ResNet 的特征提取能力构造编码器，说明了编码器利用 ResNetV2 残差块结构的优势、编码网络层的具体参数和输入输出维度；利用 Swin Transformer 的上下文建模能力构造解码器，说明了解码器利用 Swin Transformer Block 结构的优势，并详细介绍了像素重排的具体过程以及解码网络的具体参数和输入输出维度。然后介绍了判别网络的判别原理和网络结构。最后详细介绍模型各网络的优化目标即损失函数计算。



## 4 实验内容及结果分析

### 4.1 实验设置

本章设计了三个实验测试本模型的防御性能，分别为主体实验、对比实验和泛化能力测试。

#### 4.1.1 实验内容

##### （1）实验一 防御主体实验

在三种源模型上进行六种对抗攻击，生成对应的对抗样本训练集和测试集。待模型训练结束后，测试并对比重建前后的干净样本和对抗样本的分类准确率，评估模型的防御性能。

##### （2）实验二 防御对比实验

为了更加客观地测试本模型的防御性能，将选取五种防御方法进行对比实验，观察这些防御方法和本模型在 ResNet50 模型上的防御性能并进行结果分析。

##### （3）实验三 泛化能力实验

为验证本模型防御性能的泛化性，需选择六种不包含在训练集里的对抗攻击生成相应的对抗样本，并测试防御模型对这些对抗样本的防御效果。

#### 4.1.2 实验环境

实验环境如表 4.1 所示。

表 4.1 实验环境

硬件环境	CPU&内存	AMD Ryzen Threadripper 2990WX, 128G
	GPU	RTX 2080 Ti, 12G
软件环境	操作系统	Ubuntu 20.04.6
	编程工具	Visual Studio Code
	编程语言	Python 3.9.19
	CUDA 环境	CUDA 11.1, cuDNN 8.0.5
	深度学习框架	PyTorch 1.8.0

#### 4.1.3 采取数据集

本文使用 MNIST<sup>[34]</sup>和 CIFAR-10<sup>[35]</sup>两个公共数据集。

MNIST 是一个包含 10 个类的手写数字识别数据集，训练集中包含 60000 张图片，测试集中包含 10000 张图片。且每张图片都是分辨率为  $28 \times 28$ ，通道数为 1 的灰度图像。

CIFAR-10 是一个包含 10 个类的用于识别普适物体的小型数据集，训练集中包含 50000 张图片，测试集中包含 10000 张图片。且每张图片都是分辨率为  $32 \times 32$ ，通道数为 3 的 RGB 图像。

## 4.2 源模型准备

本文使用 ResNet18、ResNet34 和 ResNet50 三种分类器。

由于预训练模型是在大规模数据集上进行预先训练得到的模型，这些模型凭借大量的数据训练，已经获得了强大的特征表征能力。我们可以利用预训练模型的参数作为网络的初始权重，通过对少量网络参数进行调整，就可以高效完成小批量数据集的训练任务。

本文利用基于 ImageNet 数据集的预训练模型库 pretrainedmodels 在 MNIST 和 CIFAR-10 数据集上做微调，完成 ResNet18、ResNet34 和 ResNet50 这三种分类器的训练。

在微调过程中，需要调整模型的初始卷积层、池化层和最后一个线性层。对于 MNIST 数据集，需要将第一个卷积层的输入通道数改成 1，CIFAR-10 数据集则将通道数改成 3，然后取消池化操作，将最大池化层替换为 Identity 层，最后修改最后一个线性层的输出维度，即将输出类别改为 MNIST 和 CIFAR-10 数据集的类别数量。训练时使用随机梯度下降（SGD）优化器，设置学习率为 0.01，并使用适用于多分类任务的交叉熵损失（CrossEntropyLoss）。为进一步优化学习率的调整，使用余弦退火学习率调度器（CosineAnnealingLR），根据训练周期变化按照余弦函数的规律对学习率进行衰减，从而在训练过程中自适应地调整学习步长。

训练完的结果如表 4.2 所示。

表 4.2 分类器训练结果（%）

	ResNet18	ResNet34	ResNet50
MNIST	98.97	99.33	99.30
CIFAR-10	94.52	94.40	95.18

### 4.3 数据集准备

为了测试本模型是否能够防御多种对抗攻击, 实验在 ResNet18、ResNet34 和 ResNet50 这三种源模型上进行 BIM、PGD、FGSM、RFGSM、MIFGSM、AutoAttack 这六种攻击, 并得到对应的训练集和测试集 1 进行主体实验, 同时选择 FFGSM、SINIFGSM、TPGD、UPGD、APGD-CE、APGD-DLR 这六种不包含在训练集里的对抗攻击生成用于泛化能力测试的对抗样本测试集 2。

因此训练集应含有 6 种对抗攻击, 测试集应含有 12 种对抗攻击。对于 MNIST 数据集来说, 训练集中包含 60000 张图片, 对应生成  $3 * 6 * 60000 = 1080000$  张对抗样本, 测试集中包含 10000 张图片, 对应生成  $3 * 12 * 10000 = 360000$  张对抗样本, 一共需生成 1440000 张对抗样本。而对于 CIFAR-10 数据集来说, 其训练集中包含 50000 张图片, 测试集中包含 10000 张图片, 因此 CIFAR-10 数据集需生成  $3 * 5 * 60000 + 3 * 12 * 10000 = 1260000$  张对抗样本。

本文利用 torchattacks 对抗攻击公开库生成对抗样本, 对于 MNIST 数据集, 设计的扰动系数大小为 80/255, 对于 CIFAR-10 数据集, 扰动系数大小为 8/255。

表 4.3 对抗样本分类准确率 (%)

Attack	ResNet18		ResNet34		ResNet50	
	MNIST	CIFAR-10	MNIST	CIFAR-10	MNIST	CIFAR-10
BIM	15.82	4.19	1.62	5.54	4.90	6.42
PGD	15.63	4.10	1.05	5.46	6.62	7.03
FGSM	11.35	10.72	13.61	9.41	14.31	12.27
RFGSM	15.89	3.79	1.60	4.92	4.94	5.93
MIFGSM	8.04	2.79	0.76	3.01	1.65	3.98
AutoAttack	14.37	35.69	2.84	27.24	8.37	39.51
FFGSM	33.98	11.79	38.23	10.69	15.80	15.06
SINIFGSM	8.44	10.07	1.64	15.00	6.35	12.78
TPGD	30.42	41.68	25.84	47.63	68.65	47.98
UPGD	8.09	2.81	0.77	2.98	17.30	3.76
APGD-CE	14.32	36.38	2.91	27.44	8.82	39.27
APGD-DLR	26.68	46.32	15.39	36.20	17.26	53.32

为验证对抗攻击有效性，通过 ResNet18、ResNet34 和 ResNet50 三个分类器测试 MNIST 和 CIFAR-10 对抗样本数据集的分类准确率，得到结果表 4.3 所示。

#### 4.4 防御主体实验及结果分析

由于训练集中包含的对抗样本数量庞大，本模型训练时仅从对抗样本训练集中随机选取 10000 张进行训练。训练防御模型时，设计训练批次大小为 64，训练周期为 600，并使用 AdamW 优化器，将初始学习率设置为 0.0002。

测试防御模型时，使用干净样本测试集和对抗样本测试集 1 中的所有样本进行测试，并对比这些样本经重建前后的分类准确率，在 MNIST 测试集 1 上的防御结果如表 4.4 所示，在 CIFAR-10 测试集 1 上的防御结果如表 4.5 所示。其中，No defense 和 Defense 分别表示未经过和经过防御的样本分类准确率；Clean 行表示干净样本的分类准确率，其余每一行都表示一种对抗攻击，将 ResNet18、ResNet34 和 ResNet50 三种分类器作为实验的攻防目标模型。

表 4.4 在 MNIST 测试集 1 上的防御效果（%）

Attack	ResNet18		ResNet34		ResNet50	
	No defense	Defense	No defense	Defense	No defense	Defense
Clean	98.97	98.60	99.33	99.02	99.30	98.99
BIM	15.82	96.50	1.62	96.40	4.90	96.66
PGD	15.63	97.74	1.05	97.57	6.62	98.13
FGSM	11.35	94.17	13.61	94.91	14.31	95.07
RFGSM	15.89	96.48	1.60	96.46	4.94	96.56
MIFGSM	8.04	95.29	0.76	94.74	1.65	95.08
AutoAttack	14.37	87.13	2.84	96.07	8.37	96.87

表 4.4 展示了在 MNIST 数据集上的防御效果。干净样本在经过图像重建后，三种目标模型的分类准确率都有轻微下降，但下降程度控制在 0.31%~0.37 内，说明干净样本的分类准确率几乎不受防御模型的影响。进一步观察可以看出，本模型将 BIM、PGD、FGSM、RFGSM、MIFGSM 这五种攻击生成的对抗样本的分类准确率恢复到了 94.17%~98.13%之间，其中，对 PGD 攻击的防御效果最好，在 ResNet50 上达到了 98.13%的准确率，对 FGSM 攻击的防御效果稍差，可能是因为 FGSM 作为单步攻击，生成的扰动较为明显，提取的特征向量中包含的噪

声或扰动更多，导致重建后的准确率相对较低，但分类精度也能达到 94.17%以上。对于 AutoAttack 攻击，在 ResNet34 和 ResNet50 两种目标模型上的防御性能效果很好，达到了 96.07% 以上，但在 ResNet18 上的防御效果最差，仅有 87.13%。尽管如此，本防御模型仍能有效防御多种对抗攻击，在 MNIST 数据集上防御效果佳，无明显短板。

表 4.5 在 CIFAR-10 测试集 1 上的防御效果（%）

Attack	ResNet18		ResNet34		ResNet50	
	No defense	Defense	No defense	Defense	No defense	Defense
Clean	94.52	85.72	94.40	86.29	95.18	87.49
BIM	4.19	80.26	5.54	78.43	6.42	81.35
PGD	4.10	80.92	5.46	79.70	7.03	82.07
FGSM	10.72	77.51	9.41	73.37	12.27	77.69
RFGSM	3.79	80.90	4.92	78.71	5.93	81.54
MIFGSM	2.79	79.73	3.01	76.13	3.98	79.55
AutoAttack	35.69	82.99	27.24	82.25	39.51	83.94

表 4.5 展示了在 CIFAR-10 数据集上的防御效果。干净样本经防御后的分类准确率下降程度在 7.69%~8.8% 之间，对于 BIM、PGD、RFGSM、MIFGSM、AutoAttack 五种对抗攻击的防御效果较好，分类准确率达到 76.13%~83.94%。其中，对于 AutoAttack 攻击的防御效果最好，分类精度在 82.25%~83.94% 之间，对于 MIFGSM 攻击的防御效果稍差，但也在 76.13%~79.73% 之间。对于 FGSM 攻击，在 ResNet18 和 ResNet50 上的防御性能效果较好，达到了 77.51% 以上，在 ResNet34 上的防御效果最差，仅有 73.37%。总体来看，本模型在 CIFAR-10 数据集上的防御表现较好，但相较于 MNIST 数据集的防御效果较差，这是因为 CIFAR-10 数据集具有三通道，且图像类型为 RGB 图像，样本比 MNIST 更复杂，在重建过程中对生成器的挑战更大。

从以上结果可以看出，本防御模型仅使用少量训练图像就可以有效抵抗多种对抗攻击，并且对于干净样本的分类影响较小，无明显防御短板。

#### 4.5 防御对比实验及结果分析

本节选取 5 种防御方法与本文防御模型进行对比实验，分别是：位深度缩减

（Bit Depth Reduction, BDR）、JPEG 压缩（JPEG Compression, JC）、随机化、像素偏转<sup>[36]</sup>（Pixel Deflection, PD）和 ComDefend<sup>[25]</sup>。

表 4.6 总结了这五种防御模型和本模型在 CIFAR-10 测试集 1 上 ResNet50 分类器的表现。可以看到，位深度缩减方法对于干净样本的影响最小，但其防御对抗攻击的能力最弱。尽管本模型对于干净样本的分类精度降低程度稍大于其他防御模型，但是对对抗攻击的防御性能明显强于其他防御模型，可以极大的还原对抗样本图像，有效恢复对抗样本的分类准确率。综合来说，本文提出的防御模型在 CIFAR-10 数据集上的防御性能要优于其他防御模型。

表 4.6 ResNet50 模型上各防御方法的防御效果对比（%）

Attack	No defense	BDR	JC	Random	PD	ComDefend	本章方法
Clean	95.18	93.65	85.84	93.58	39.34	91.54	87.49
BIM	6.32	9.66	40.5	30.05	19.26	14.96	81.35
PGD	6.94	10.74	45.07	34.87	20.04	17.14	82.07
FGSM	12.83	14.33	27.85	27.56	18.19	16.71	77.69
RFGSM	5.91	9.9	43.72	32.33	19.97	15.78	81.54
MIFGSM	3.89	6.05	32.94	25.16	17.13	9.58	79.55
AutoAttack	39.67	40.21	56.78	57.38	22.98	43.17	83.94

#### 4.6 泛化能力实验及结果分析

为客观测试模型的防御性能，泛化能力常常作为评估防御模型的一个重要指标。一个鲁棒的防御模型不仅要在训练数据上展现卓越的防御性能，还应该具有良好的泛化能力，即能够防御未出现在训练集中的攻击方法。本节采用 FFGSM、SINIFGSM、TPGD、UPGD、APGD-CE、APGD-DLR 攻击来验证本防御模型的泛化能力。理论上，FFGSM、SINIFGSM 作为 FGSM 的迭代变体，TPGD、UPGD、APGD-CE、APGD-DLR 作为 PGD 的变体，这些攻击方法都依赖于梯度攻击，具有一定的相似性，本防御模型应该在这些攻击上表现出良好的防御性能。

测试泛化能力时，使用上文制作的测试集 2 中的所有样本，并对比样本重建前后的分类准确率。测得在 MNIST 测试集 2 上的泛化防御效果如表 4.7 所示，在 CIFAR-10 测试集 2 上的泛化防御效果如表 4.8 所示。

表 4.7 在 MNIST 测试集 2 上的泛化防御效果（%）

Attack	ResNet18		ResNet34		ResNet50	
	No defense	Defense	No defense	Defense	No defense	Defense
Clean	98.97	98.60	99.33	99.02	99.30	98.99
FFGSM	33.98	98.21	38.23	98.39	15.80	98.24
SINIFGSM	8.44	96.28	1.64	95.52	6.35	96.48
TPGD	30.42	97.79	25.84	97.71	68.65	98.02
UPGD	8.09	95.36	0.77	94.72	17.30	95.20
APGD-CE	14.32	87.19	2.91	96.18	8.82	96.95
APGD-DLR	26.68	87.28	15.39	96.48	17.26	97.15

表 4.8 在 CIFAR-10 测试集 2 上的泛化防御效果（%）

Attack	ResNet18		ResNet34		ResNet50	
	No defense	Defense	No defense	Defense	No defense	Defense
Clean	94.52	85.72	94.40	86.29	95.18	87.49
FFGSM	11.79	79.38	10.69	75.90	15.06	79.97
SINIFGSM	10.07	81.10	15.00	78.46	12.78	81.23
TPGD	41.68	84.46	47.63	84.51	47.98	85.65
UPGD	2.81	79.87	2.98	76.78	3.76	79.96
APGD-CE	36.38	83.11	27.44	82.05	39.27	84.23
APGD-DLR	46.32	83.14	36.20	82.71	53.32	84.38

可以看到, MNIST 测试集 2 经防御后的分类准确率在 87.19%~98.39%之间, CIFAR-10 测试集 2 经防御后的分类准确率在 75.90%~85.65%之间, 面对这六种未知的对抗攻击, 本防御模型仍能保持良好的防御效果。

#### 4.7 样本展示

图 4.1 和图 4.2 分别展示了部分 MNIST 数据集和 CIFAR-10 数据集的干净样本、经对抗攻击后的对抗样本和经防御模型重建后的对抗样本。可以看到对抗样本上存在着一些噪声点, 经防御模型重建后可以消除大部分噪声, 得到的样本与干净样本更相似。

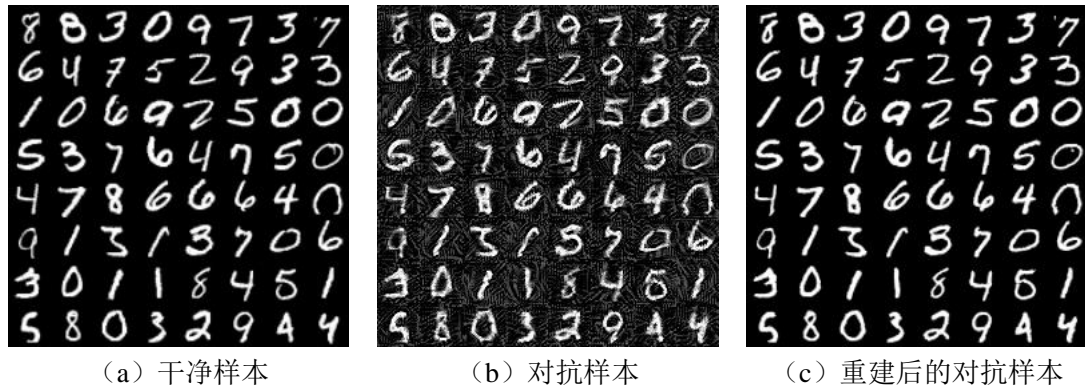


图 4.1 MNIST 数据集样本展示



图 4.2 CIFAR-10 数据集样本展示

#### 4.8 本章小结

本章在 MNIST 和 CIFAR-10 数据集上进行了主体实验、对比实验和泛化能力实验测试本模型的防御性能。在实验之前需训练源模型和制作对抗样本数据集。通过预训练模型在 MNIST 和 CIFAR-10 数据集上做迁移学习完成分类器的训练。并采用 BIM、PGD、FGSM、RFGSM、MIFGSM、AutoAttack 六种攻击方法生成对抗样本训练集和测试集 1，FFGSM、SINIFGSM、TPGD、UPGD、APGD-CE、APGD-DLR 攻击方法生成测试集 2。主体实验利用测试集 1 测试防御性能，从防御结果表明，本文提出的防御模型可以有效防御多种对抗攻击，且对干净样本的分类准确率影响较小。对比实验证明了本模型无明显防御短板，且防御性能明显优于其他防御模型。在泛化能力测试实验中，对测试集 2 的样本进行重建并比较重建前后的分类准确率，可以看出本模型的防御性能具有良好的泛化性，展现出一定的鲁棒性。



## 5 总结与展望

近年来，神经网络凭借其强大的数据处理能力和特征提取表达能力，在众多领域取得了显著的成就。然而，随着神经网络的广泛应用，其安全隐患也逐渐浮现。例如对抗样本的存在对深度学习模型在多个关键安全领域的应用构成了严重威胁，成为了一个亟待解决的重要问题。并且随着对抗样本技术的不断革新和进步，现有的防御策略可能无法防御升级后的对抗样本的攻击，同时多数对抗防御的方法往往聚焦于特定的攻击类型，因此它们在面对多样化的攻击时普遍缺乏泛化性。

针对目前对抗防御存在的问题，本文提出一种基于 GAN 的对抗样本防御模型。该模型在目标检测模型 GANomaly 的网络结构的基础上，利用 ResNetV2 残差块的特征提取能力构造编码器，提取对抗样本中的显著特征，将对抗样本压缩为特征向量，并利用 Swin Transformer 的上下文建模能力，将原下采样部分改为上采样，构造解码器，用于将特征向量重建为图片。同时，在模型训练的过程中引入干净样本，学习干净样本特征分布的固有特性，通过生成网络和判别网络的优化目标，让生成样本即重建后的对抗样本越来越接近于干净样本，最终使得对抗样本分类正确，达到防御对抗攻击的目的。

为多方面验证本防御模型的防御有效性，本文设计主体实验、对比实验和泛化能力测试实验。在进行实验之前，需要完成源模型的训练和对抗样本数据集的制作。通过预训练模型在 MNIST 和 CIFAR-10 数据集上做迁移学习完成分类器的训练。并采取六种攻击方法生成对抗样本训练集，十二种攻击方法生成测试集，在这众多训练样本中随机选取 10000 张进行模型训练。准备工作完成后进行防御性能的测试，主体实验利用测试集 1 测试防御性能，从结果表明，本防御模型对于干净样本的分类准确率的影响较小，几乎不影响 MNIST 数据集的干净样本，且可以有效防御多种对抗攻击，MNIST 对抗样本数据集重建后的平均分类准确率达到 95.65%，CIFAR-10 对抗样本数据集重建后的平均分类准确率达到 79.83%，防御效果较好。对比实验证明了本防御模型的防御能力优于绝大多数现有的基于预处理的防御方法。最后在泛化能力测试实验中，本模型表现出良好的泛化性能，对于未知的对抗攻击，本模型仍能消除这些对抗样本中的大多数对抗扰动。

本文提出的防御模型在 MNIST 数据集上的防御表现强于 CIFAR-10 数据集，并且在现实中所面临的数据集却更为复杂多样，潜在的安全威胁同样难以预知，

因此，本文提出的防御模型仍有许多改进空间，未来的研究可以从以下几个维度展开：

（1）对于 CIFAR-10 数据集，减少防御模型对干净样本的分类准确率的影响，进一步提高重建后对抗样本的分类准确率，优化模型提高防御性能。

（2）考虑对模型网络结构进一步优化和拓展，以期望在面对更庞大更复杂的数据集时，本防御模型仍能展现出良好的防御能力。

（3）本防御模型具有良好的防御性能，但相比于其他基于数据预处理的防御模型，计算成本较低，因此可以在保障优异的防御性能的同时，进一步提升计算效率。

## 参考文献

- [1] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[J]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 770-778.
- [2] Kumar A, Irsoy O, Ondruska P, et al. Ask me anything: Dynamic memory networks for natural language processing[J]. International conference on machine learning, 2016: 1378-1387.
- [3] Kim J, Park C. End-to-end ego lane estimation based on sequential transfer learning for self-driving cars[J]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017: 30-38
- [4] Madugalla A K, Rajapakse R N, Amarasinghe I U, et al. FaceID: A 3D computer graphic application for forensic medicine: A novel semi-automated muscle based digital sculpting initiative for forensic facial reconstruction in Sri Lanka. International Conference on Computer Medical Applications, 2013: 40-49
- [5] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing properties of neural networks[J]. Computer Science, 2013(4): 1312-1322.
- [6] Yuan X, He P, Zhu Q, et al. Adversarial examples: Attacks and defenses for deep learning[J]. IEEE transactions on neural networks and learning systems, 2019, 30(9): 2805-2824.
- [7] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. International Conference on Learning Representations, 2015: 647-659.
- [8] 刘瑞祺, 李虎, 王东霞, 等. 图像对抗样本防御技术研究综述[J]. 计算机科学与探索, 2023, 17(12): 2827-2839.
- [9] Moosavi-Dezfooli S M, Fawzi A, Frossard P. Deepfool: a simple and accurate method to fool deep neural networks[J]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 2574-2582
- [10] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale[J]. International Conference on Learning Representations, 2017: 335-355.
- [11] Madry A, Makelov A, Schmidt L, et al. Towards deep learning models resistant to adversarial attacks[M]. International Conference on Learning Representations, 2018:1132-1141.
- [12] Papernot N, McDaniel P, Goodfellow I, et al. Practical black-box attacks against machine learning[M]. Proceedings of the 2017 ACM on Asia conference on computer and communications security, 2017: 506-519.

- [13] Chen P Y, Zhang H, Sharma Y, et al. ZOO: Zeroth Order Optimization based Black-box Attacks to Deep Neural Networks without Training Substitute Models[C]. Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 2017: 15-26.
- [14] Tramèr F, Kurakin A, Papernot N, et al. Ensemble Adversarial Training: Attacks and Defenses[M]. 2017.
- [15] Shafahi A, Najibi M, Ghiasi A, et al. Adversarial Training for Free![M]. 2019: 3353-3364.
- [16] Gu S, Rigazio L. Towards deep neural network architectures robust to adversarial examples[M]. International Conference on Learning Representations, 2015: 876-888
- [17] Papernot N, McDaniel P, Wu X, et al. Distillation as a defense to adversarial perturbations against deep neural networks[M]. 2016 IEEE Symposium on Security and Privacy (SP). IEEE, 2016: 582-597
- [18] Hendrycks D, Gimpel K. Early Methods for Detecting Adversarial Images[M]. 2017
- [19] Lu J, Issarano T, Forsyth D. Safetynet: Detecting and rejecting adversarial examples robustly[J]. Proceedings of the IEEE International Conference on Computer Vision, 2017: 446-454
- [20] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[J]. Proceedings of the 27th international conference on machine learning (ICML-10), 2010: 807-814.
- [21] Frosst N, Sabour S, Hinton G. DARCCC: Detecting adversaries by reconstruction from class conditional capsules[J]. arXiv preprint arXiv:1811.06969, 2018.
- [22] Xie C, Wang J, Zhang Z, et al. Mitigating adversarial effects through randomization[J]. arXiv:1711.01991, 2017.
- [23] Liao F, Liang M, Dong Y, et al. Defense against adversarial attacks using high-level representation guided denoiser[C]. Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Jun 18- 23, 2018. Washington: IEEE Computer Society, 2018: 1778-1787.
- [24] Jia X, Wei X, Cao X, et al. Comdefend: An efficient image compression model to defend adversarial examples[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019: 6084-6092.
- [25] Dongyu Meng, Chen Hao. Magnet: a two-pronged defense against adversarial examples[C]. Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, 2017: 135-147.
- [26] Samangouei P, Kabkab M, Chellappa R. DefenseGAN: protecting classifiers against adversarial attacks using generative models[J]. arXiv:1805.06605, 2018.

- [27] Akcay S , Atapour-Abarghouei A , Breckon T P .GANomaly : semi-supervised anomaly detection via adversarial training[J]. Newcastle University, 2018.
- [28] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[M]. Advances in neural information processing systems, 2014: 2672-2680.
- [29] 何永庆. 基于 VAE-GAN 的对抗样本防御方法研究[D]. 华中科技大学, 2020.
- [30] Croce F, Hein M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks[M]. 2020.
- [31] XU W, EVANS D, QI Y. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks[C]. 25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018. The Internet Society, 2018.
- [32] He K , Zhang X , Ren S ,et al.Identity Mappings in Deep Residual Networks[J].Springer International Publishing, 2016.
- [33] Liu Z , Lin Y , Cao Y ,et al.Swin Transformer: Hierarchical Vision Transformer using Shifted Windows[J]. 2021.
- [34] LeCun Y, Cortes C, Burges C J C. The MNIST database of handwritten digits. online: <http://yann. lecun. com/exdb/mnist>, 1998.
- [35] Krizhevsky A, Nair V, Hinton G. The cifar-10 dataset. online: <http://www. cs. toronto. edu/kriz/cifar. html>, 2014.
- [36] Prakash A, Moran N, Garber S, et al. Deflecting Adversarial Attacks with Pixel Deflection[C]. 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. Computer Vision Foundation / IEEE Computer Society, 2018:8571-8580.

## 致谢

至此，本科生涯即将落下帷幕。

回望这四年，有开心有痛苦，有迷茫有收获，经历了许多更成长了许多。

首先我要感谢我的导师王昕老师，感谢您在过去四年对我的帮助与指导，感谢您总是耐心地为我们提供细心的指导和合理的建议。也感谢信计学院的各位老师，总是热心地帮助我们解决问题，陪伴了我们四年的时光。

然后，我想感谢师兄对我的指导与鼓励，不厌其烦地帮助我解决问题，并提出许多宝贵的意见。最后，感谢家人、朋友、同学对我的帮助和支持，是你们对我的信任，让我有勇气一直走下去。

相信我们一定会越来越好。

作者：尤量子

2024 年 5 月 30 日