分 类 号_____ 学号 M201772696
学校代码 10487 密级_____

華中科技大學

硕 士 学 位 论 文

脑机接口分类问题中的
通用对抗扰动

学位申请人： 刘子涵
学 科 专 业： 控制科学与工程
指 导 教 师： 伍冬睿 教授
答 辩 日 期： 2020 年 3 月 10 日

A Thesis Submitted in Partial Fulfillment of the Requirements
For the Degree of Master of Engineering

Universal Adversarial Perturbations for CNN Classifiers in EEG-Based BCIs

Candidate : Zihan Liu
Major : Control Science and Engineering
Supervisor : Prof. Dongrui Wu

Huazhong University of Science & Technology

Wuhan 430074, P. R. China

March, 2020

独创性声明

本人声明所呈交的学位论文是我个人在导师的指导下进行的研究工作及取得的研究成果。尽我所知，除文中已标明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期：　　年　　月　　日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本论文属于　保密□，在 ____ 年解密后适用本授权书。

　　不保密 □。

（请在以上方框内打“√”）

学位论文作者签名：　　指导教师签名：

日期：　　年　　月　　日　　日期：　　年　　月　　日

摘 要

通用对抗扰动是一种十分微小且对所有数据完全相同的扰动，它拥有足够强的攻击能力可以使得CNN模型的分类性能显著降低。目前，有多个专门针对脑机接口中EEG数据的CNN分类器被提出，这些模型已经被发现非常容易受到对抗样本的攻击，揭示了脑机接口系统的一个关键的安全问题。然而，考虑到攻击的因果性和EEG信号的时序性，一般的对抗攻击方法不方便应用到脑机接口系统中，而通用对抗扰动能够同时解决上述问题，实时的攻击脑机接口系统。本文主要关注脑机接口分类问题中通用对抗扰动，也就是如何设计出通用对抗扰动成功攻击脑机接口系统以及如何防御它。主要分为如下三个部分：

(1) 首先介绍了基于DeepFool的通用对抗扰动设计算法，首次将通用对抗扰动的思想引入到脑机接口系统中。我们针对EEG数据设计了一种迭代式的通用对抗扰动，验证了这种扰动在非目标攻击场景中对三种常用的CNN分类器上攻击的有效性。

(2) 进一步提出了基于总损失最小化的通用对抗扰动设计算法，使用优化的方式来生成通用对抗扰动，并成功将通用对抗扰动从非目标攻击扩展到目标攻击。实验验证了在非目标攻击场景中，我们提出的方法能够在扰动更小的情况下取得更强的攻击性能。另外，实验结果表明在目标攻击场景中，我们提出的方法取得了接近100%的目标率，能够迫使CNN模型将EEG数据分类到任意指定类别。据我们所知，这是第一项进行通用对抗扰动目标攻击的研究。

(3) 首次在脑机接口系统中研究了通用对抗扰动的防御机制，从对抗样本的检测和模型对抗鲁棒性两个方面对防御能力进行了分析，并通过实验验证了搭建通用对抗扰动检测模块和使用投影梯度下降方法训练模型这两种方式能够分别在不同应用场景中成功防御通用对抗扰动。

.

关键词： 脑机接口　　脑电图　　卷积神经网络　　通用对抗扰动

Abstract

Universal adversarial perturbations (UAPs), which are small and example-independent, yet powerful enough to degrade the performance of a CNN model, when added to a benign example. Multiple convolutional neural network (CNN) classifiers have been proposed for electroencephalogram (EEG) based brain-computer interfaces (BCIs). These CNN models have been found vulnerable to adversarial examples, exposing a critical security concern of BCIs. However, considering the causality of attack and the time sequence of EEG signal, the common adversarial attacks are not convenient to be applied to BCIs. The UAP can solve the above problems simultaneously, and can attack BCIs in real-time. This thesis mainly focuses on universal adversarial perturbations for CNN Classifiers in EEG-Based BCIs, that is, how to craft a UAP to attack BCI and defense it. It is mainly divided into three parts:

(1) Firstly, a DeepFool based algorithm for generating a UAP is introduced. The idea of UAP is introduced into BCIs for the first time. We designed an iterative UAP for EEG data and verified its effectiveness on three popular CNN classifiers in non-target attack scenarios.

(2) We further propose a novel total loss minimization (TLM) approach to generate UAPs by using a optimized method, and our approach can be applied to both target and non-target attacks. Experiments demonstrated that our proposed can achieve better attack performance with a smaller perturbation, compared with the traditional DeepFool based approach. In addition, the experiments also demonstrated that our proposed can achieves nearly 100% target rate in the target attack scenario, which can force the CNN model to classify all EEG data into any specified class. To our knowledge, this is the first study on UAPs for target attacks.

(3) We study the defence of UAPs in BCIs for the first time, and analyse the performance from the detection of adversarial example and adversarial robustness of model. Experiments demonstrated that detection module and projected gradient descent method can defence against UAPs in different application scenarios.

.

Key words: Brain-computer interface electroencephalogram convolutional neural network universal adversarial perturbation

目　录

1 绪论

1.1 研究意义

脑机接口是一种能够让人的大脑信号与计算机进行直接交互的系统。脑电图（Electroencephalogram，EEG)）通过记录大脑头皮上的电活动，因其低廉的成本和便捷的操作方式，是脑机接口应用中最为广泛的输入信号。在基于脑电图的脑机接口系统中，常见的范式有P300诱发电位[1–4]、运动想象[5]（Motor imagery，MI）、稳态视觉诱发电位[6]（Steady-state visual evoked potentials，SSVEP）等。

近年来，深度学习由于其自动提取特征的能力，在脑机接口的脑电信号解码中得到了越来越广泛的应用。针对基于EEG的脑机接口系统，提出了多种卷积神经网络（Convolutional neural network，CNN）分类器。Lawhern 等人提出了EEGNet，这是一种紧凑的CNN模型，在几个基于EEG的BCI任务中表现良好。Schirrmeister等人[7]提出了深层模型（DeepCNN）和浅层模型（ShallowCNN）用于脑电图分类。也有研究将脑电图信号转换成光谱图或地形图，然后将其输入到深度学习分类器中[8–10]。本文主要研究以原始脑电图信号为输入的CNN分类器，该方法也可以推广到其他形式的输入。

最近，在图像领域中，CNN模型已经被发现非常容易受到对抗样本的攻击：给正常图片添加一个精心设计的微小扰动，就能够使得CNN模型将这张图片分类错误。然而，脑机接口作为一种将大脑与计算机连接在一起进行信息交互的系统，已经在很多应用中被证实可以提高人类的生活质量。如果对抗样本也能成功的攻击脑机接口系统，那么会直接造成系统的决策错误，显著的降低使用者的用户体验，甚至会将使用者推入到危险的环境中，造成十分严重的影响。因此，本文基于脑机接口的分类问题，从攻击和防御两个角度出发，对通用对抗扰进行了研究，希望本文的工作能为设计出更安全的脑机接口系统提供思想和帮助。

1.2 脑机接口

脑机接口是指通过在人类或动物大脑与外部设备(比如计算机、机器人等)间建立直接连接通路，来实现神经系统和外部设备间信息交互与功能整合的技术。简单来说，就是通过大脑信号来控制机器的一种技术。一个脑机接口系统既包含了将大脑信号传输到外部硬件的接口，也包括指向大脑发送信号的技术。它主要测量和使用中枢神经系统（Central nervous system，CNS）产生的信号。认为脑机接口是读心术的设备是一种误解，脑机接口并不像从毫无戒心或不情愿的用户那里提取信息那样读取人的思想，而是让用户通过大脑信号而不是肌肉来对世界采取行动。用户通

常在经过一段时间的训练后，生成对意图进行编码的脑信号，而脑机接口也在训练后对信号进行解码，并将其转换为命令，发送到一个输出设备，以实现用户的意图。脑机接口技术的实现意味着，人与机器之间的主要交互方式，除了手工输入和语音交互之外，也可以通过大脑直接对机器进行指令操作。

一些最早的脑机接口技术只是简单地记录来自大脑的信号，随着时代的发展与科技的进步（例如虚拟现实技术），各种不同类型的脑机接口技术得到了发展。在现代的脑机接口技术中，可以通过手术在大脑中植入芯片或电极，用于提高与某些大脑活动相关的技能，如视觉、听觉或者其他人类或动物功能。这些被称为神经修复术的系统设备已经被开发出来，并在世界各地作为辅助人类能力的功能性手段而使用。

脑机接口技术根据系统的输入信号通常可分为非侵入式、半侵入式和侵入式三种。这三种类别的位置和特征如下所示：

(1) 非侵入式：通常是一个可以让人穿戴的设备（例如脑电帽），这种接口不需要动手术，直接从大脑外部采集大脑信号。这种技术避免了昂贵和危险的手术，但是由于颅骨对于大脑信号的衰减作用，以及对于神经元发出的电磁波的分散和模糊效应，使得记录到的信号强度和分辨率并不高，很难确定发出信号的脑区或者相关的单个神经元的放电。还需要注意的是，由于用户个体之间存在差异性，使用前往往需要进行校准。

(2) 半侵入式：接口一般植入到颅腔内，但是位于灰质外，其空间分辨率不如侵入式脑机接口，但是优于非侵入式。其另一优点是引发免疫反应和愈伤组织的几率较小，主要基于皮层脑电图（Electrocorticography，ECoG）进行信息分析。

(3) 侵入式：需要通过手术在大脑的灰质中植入电极或者芯片。人的大脑中有上千亿个神经元，通过植入电极，可以记录局部场电位（Local Field Potential，LFP）、单个神经元的活动（即动作电位/锋电位, Spike）和多个神经元活动（Multiunit Activity，MUA）。侵入式脑机接口获取的大脑信号质量好，时间和空间解析度高，有可能取得比非侵入式脑机接口系统更好的性能。但是因为手术风险，目前侵入式脑机接口系统在动物上应用较多，对人类的研究多限于瘫痪病人等临床特殊群体。

常用的非侵入式信号有头皮脑电（EEG）、功能近红外光谱（Functional near-infrared spectroscopy，fNIRS）和功能核磁共振成像（Functional magnetic resonance imaging，fMRI）等，其中以EEG最为常见。EEG通过记录大脑头皮上的电活动，因其低廉的成本和便捷的操作方式，是脑机接口应用中最为广泛的输入信号。在基于EEG的脑机接口系统中，常见的范式有P300诱发电位[1–4]、运动想象[5]（MI）、稳态视觉诱发电位[6]（SSVEP）：

(1) 基于P300的BCI系统：当人脑受到一个特定的、少见的刺激时，会在刺激产生之后300毫秒产生一个比较大的正向电位峰，称为P300。诱发P300事件相关电位（Event Related Potential, ERP）的特定事件称为oddball范式（小概率刺激范式）。事

件发生的概率越小，P300的峰值越高。一般的P300范式有听觉P300和视觉P300，目前应用较多的是视觉P300范式。

(2) 基于MI的BCI系统：肢体运动的执行和想象会影响大脑特定区域感觉运动皮层的节律活动的变化。比如对于左手的想象动作会导致大脑右半球激活强烈，右手的想象动作会导致大脑左半球激活强烈，而脚的想象动作会导致大脑顶部激发强烈。根据这些特征可以把运动想象转化成输出指令，用于BCI 系统的控制。

(3) 基于SSVEP的BCI系统：稳态视觉诱发电位是由快速重复刺激诱发的脑电的稳定振荡。一般的刺激源有闪光灯、发光二极管和显示器的棋盘格模式等。当人眼注视闪烁的刺激源时，脑电的振荡频率会趋近于刺激源的闪烁频率。用不同的闪烁频率编码不同的指令，我们就可以通过注视不同频率的刺激源发出不同的指令。

如图 1-1所示，一个脑机接口系统一般包含4个模块：信号采集、信号预处理、机器学习和控制动作。其中，如果使用的是传统的机器学习算法，机器学习模块通常包含特征提取和分类（回归）这两个步骤。然而，由于深度学习的出现，特征提取和分类（回归）能够被无缝的集成到一个深度神经网络模型中，从而大大的节省手工提取特征的代价。

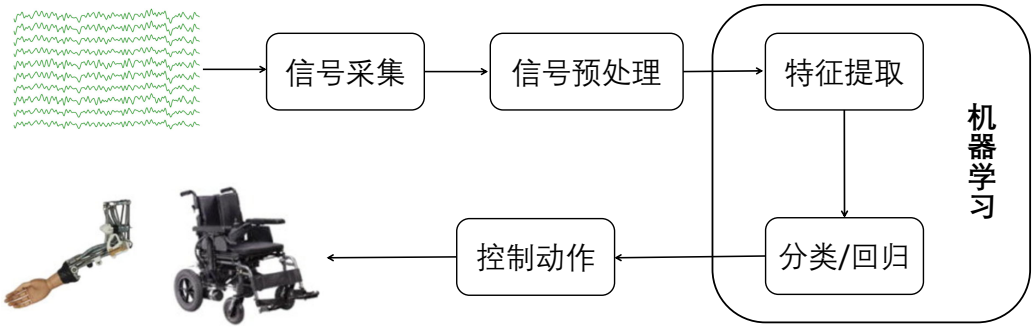


图 1-1 脑机接口系统使用传统机器学习算法的过程。如果使用深度学习，则不需要手动提取特征。

可观测的脑电信号能否作为信息载体用于人机通信，或用于控制诸如假肢之类的装置? 这是Vidal[11]在1973年提出的问题。他的脑机接口项目是一个早期的尝试，目的是评估在人机对话中使用神经元信号的可行性，从而使计算机成为大脑的假肢。虽然在20世纪60年代末对猴子的研究[12]表明，来自单个皮层神经元的信号可以用来控制仪表指针，但对人类的系统研究真正开始于20世纪70年代。人类脑机接口研究的最初进展缓慢，并且受到计算机能力和我们自己的大脑生理学知识的限制。到1980年，Elbert等人[13]证明，在脑电图（EEG）活动中，给予慢皮层电位生物反馈的人可以改变这些电位，从而控制火箭图像在电视屏幕上的垂直运动。1988 年，Farwell和Donchin[14]展示了如何利用P300事件相关电位让普通志愿者在电脑屏幕上拼写单词。自20世纪50年代以来，在感觉运动皮层上记录的mu和beta节律（即感觉

运动节律）被认为与运动或运动表象有关。20 世纪70年代末，Kuhlman[15]研究表明，EEG反馈训练可以增强mu节律。从这些信息开始，Wolpaw等人[16–18]培训多名志愿者接受训练，控制感觉运动节律振幅，并使用它们在计算机屏幕上精确地移动光标。2006年，一名年轻男子在C3-C4颈椎损伤后完全四肢瘫痪，在他的初级运动皮层中植入了一个微电极阵列。利用从这个电极阵列获得的信号，脑机接口系统使患者能够打开模拟的电子邮件、操作电视、打开和关闭假肢，以及使用机械手臂执行基本动作[19]。2011年，Krusienski和Shih[20]证明，直接从皮层表面记录的信号（ECoG）可以被脑机接口翻译，使人能够在计算机屏幕上准确地拼写单词。脑机接口的研究正在以极快的速度增长。

近年来，国内外有许多关于脑机接口的研究。2005年，Cyberkinetics公司对病人进行了临床试验，研究表明通过在大脑中植入电极芯片，除了能够控制电脑鼠标移动以外，还能够让瘫痪病人通过运动想象直接控制机械臂。2011年，华盛顿大学医学院的研究人员[21]通过改造脑机接口来监听大脑中控制语言的区域，使用瘫痪病人曾经使用过的大脑区域来模拟他们自己的声音来打招呼，这种研究有助于恢复患者因脑损伤或残疾而丧失的能力。2012年，加州大学洛杉矶分校的一项研究[22]表明可以通过刺激大脑的一个关键节点来增强人类患者的记忆，可能会为早期阿尔茨海默症(Alzheimer’s disease)患者提供一种提高记忆力的新方法。2012年，巴西世界杯期间，研究人员展示了使用脑机接口系统和机械外骨骼，帮助截肢残疾者完成一次开球。2013年，匹兹堡大学的研究人员[23]表明，通过植入芯片使得一名30岁的瘫痪男子仅用思想就能控制电脑屏幕上一个字符的三维运动，这也是他自从七年前在一次摩托车事故中受伤以来，第一次能够移动机械手臂来触摸朋友的手,这种侵入式脑机接口系统不仅可以操作机器手臂，还能够控制其完成手部抓取的动作。2014年，Grau等人[24]的研究描述了两个相距5000 英里的人类实验对象的完整头皮之间通过互联网成功传输信息的过程，使用联网脑电图（EEG）和机器人辅助和图像引导的经颅磁刺激（TMS）技术，通过计算机介导的脑-脑传输，成功地将单词“hola”和“ciao”从印度的一个地点传输到法国的一个地点，让人的脑与脑之间的直接交流称为可能。2015年，加州大学欧文分校的研究[25]表明，脑机接口系统可以帮助脊髓损伤患者通过EEG信号控制腿重新行走，该系统从参与者的大脑中获取电信号，然后将其传输到膝盖周围的电极上，以产生运动。2016年，Christian等人[26]使用脑机接口系统成功解码来自大脑信号的语言，将癫痫患者的想法解码成实际的语音或文字。2017年，在斯坦福大学领导的一项研究报告[27]中，三个有运动障碍的参与者仅仅通过想象自己的手部运动来控制屏幕上的光标，脑机接口系统可以让瘫痪的人通过大脑的直接控制计算机，以迄今为止报道的最高速度和准确度打字，这种点击式操作方法可以应用于各种计算设备，包括智能手机和平板电脑，无需进行大量修改。2018年，Perdikis等人[28]研究表明当人与机器都被允许学习时，使用脑机接口的人效率更高，研究人员训练了两名四肢瘫痪的使用者参加国际脑机接口竞赛。两个人

都在不断地学习如何控制脑机接口，并在比赛中取得了最好的成绩，证实了研究者的假设，即相互学习在脑机接口训练中起着基础作用。2019年，布朗大学的研究人员[29] 从非人类灵长类动物大脑记录的神经信号中重建英语单词，他们通过记录与灵长类动物听特定词汇相关的次级听觉皮层的复杂神经兴奋模式，然后利用这些神经数据来重建这些单词的声音，并保持高保真度，这项研究可能会导致新的神经修复手术产生，帮助听力受损的人生活。

无论是那种脑机接口应用，其当前可实现的性能距离人们在科幻作品中的设想还有很长的路要走。工程上仍存在难题需要解决。除了计算机技术和传感技术的限制以外，脑接接口的应用主要还受到脑科学和神经科学发展的影响。由于我们目前对于大脑机制的了解还十分有限，对人脑的研究还需要进一步的探索。值得一提的是，我国的脑计划“脑科学和类脑智能”强调了脑研究和人工智能（Artificial intelligence，AI）的结合，可能会为脑机接口的研究带来突破口。此外，脑机接口的发展还涉及到安全性的问题，目前，侵入式脑机接口在很多应用取得了一些成果，由于脑电信号的采集离人脑越近会越准确，需要往人脑内植入芯片和电极，往往会对人脑造成不可避难的创伤和损坏，这一点也是需要科学的不断进步和创新，才能减少非患者用户使用脑机接口的恐惧以及提高正常人利用脑机接口的效率，如何设计出更安全的脑机接口系统也将成为未来领域的研究重点。

1.3 深度神经网络

机器学习是人工智能的一种应用，它为系统提供了自动学习和从经验中改进的能力，而无需显式编程。机器学习侧重于开发能够访问数据并自己使用数据的计算机程序。它能够发现并学习数据或现象之间的复杂规则然后加以利用，是一种将信息转化为知识的技术。随着人工智能技术与设备的发展，传统的机器学习算法还是存在一些没能良好解决的问题，处理原始数据的能力有限。例如在图像处理、语音识别、自然语言处理等领域，我们使用传统机器学习方法去解决问题一般分为以下三个步骤：从数据源获取数据，然后经过数据预处理、特征提取和特征选择，最后使用算法进行推理、预测或者识别。最后一个部分也就是我们所说的机器学习模块，如今有着许多相关的研究和算法在不同的应用方向都取得了非常不错的性能，例如支持向量机（Support vector machine，SVM）、Logistic回归、多层感知机（Multi layer perceptron，MLP）、K-means等等。中间的部分可以概括为特征表达，一个良好的特征表达会对之后机器学习的决策结果起到至关重要的影响。然而，特征表达往往是人工完成的，通过手工特征提取之后，再输入到搭建好的机器学习算法模型中进行性能评估和测试，这一部分会耗费大量的时间。除了耗时以外，手工提取特征还需要相当多的专业知识，挑选出良好的特征一般需要行业经验和运气。于是，为了免去手工提取特征这一步骤，表示学习（Representation learning）得到了迅速的

发展，它允许向模型直接输入原始数据并自动提取出检测或分类任务所需的特征。其中，深度学习是一种多层表示的表示学习方法。

当前，深度学习成为了机器学习领域中一个研究热点，有时也被称为深度神经学习或深度神经网络（Deep neural network，DNN）。它使用多层神经网络，能够自动的在原始输入数据上提取出高层次的特征，从而免去人工选取特征的过程，大大的简化了预处理过程。例如，在图像处理领域中，DNN模型中较低的层可以自动学习出物体的边缘等特征信息，随着层数的逐渐增加，较高的层得到的特征则会学习到越来越抽象（比如字母、数字或者人脸）。DNN模型结构包含输入层、隐层和输出层，其中每一层都包含了许多神经元节点，用于转换和特征提取，后一层都以前一层的输出作为输入。DNN模型学习的过程一般可分为两个阶段，即训练阶段和推理阶段。训练阶段通过学习大量的数据来确定合适的特征以及模型参数，使用反向传播算法来指示机器应该如何改变其内部参数从而来发现大数据集中复杂的结构；而推理阶段则利用模型学习到的知识对新的未标记的数据进行决策或识别。使用不同的抽象阶段和多种结构的表示，模型的学习过程既可以在有监督条件，也可以在无监督条件下进行。

深度神经网络模型主要分为自动编码机（Auto encoder，AE）、卷积神经网络（Convolutional neural network，CNN）、深度信念网络（Deep brief network，DBN）、深度堆叠网络（Deep stacking network，DSN）和循环神经网络（Recurrent neural network，RNN）这五种类型。其中，CNN和RNN在实际应用中最为广泛。

CNN是一种特殊的深度前馈网络，近年来在许多领域特别是计算机视觉上取得了巨大的成功。第一个CNN由LeCun等人[30]提出，之后经过各项研究，发展出了许多新的结构与框架[31]。CNN主要用于处理多维矩阵形式的数据，例如由三个二维矩阵组成的彩色图像，包含了RGB 三个彩色通道中的像素点信息，除此以外，还可以用于一维的信号数据、二维的音频或文本数据，以及三维的视频数据。CNN中的核心思想是局部连接、权值共享、池化以及使用多层网络，前三种结构思想结合起来获得了某种程度的位移、尺度、形变不变性。一个典型的CNN网络结构如图1-2所示。其中，CNN的主要由卷积层（Convolutional layer）、池化层（Pooling layer）以及全连接层（Fully connected layer）组成，用于提取特征。卷积层的是使用几种不同的滤波器（Filter）对输入数据进行卷积运算，每个过滤器与前一层的输出局部相连，卷积运算之后这个局部加权和的结果会通过一个非线性激活函数，例如ReLU。为了减少计算量以及更有效的进行特征抽取，卷积层中所有滤波器权值共享。进行卷积层操作的原因如下：首先，对于像图像这中多维数据，局部数据点通常是高度相关的，有着容易被检测到的局部信息；其次，图像和其他信号局部信息统计不受位置的影响，例如一个图形如果出现在图片中的某个地方，那么它在其他图片中就可以出现在任何地方，因此，不同的滤波器实行了权值共享，并且在不同的彩色通道之间也采用了相同的模式。卷积层提取到前一层数据的局部特征信息之后，池化

层将相近的特征信息进行合并。池化层最直接的作用就是引入了不变形，例如最常见的最大池化（Max pooling）操作，只保留区域内的最大值，那么这个得到的新特征不会随着区域内部特征位置的变化而改变。在重复进行多次卷积- 池化操作之后，最后使用全连接层，与输出信息相连。和常规的神经网络一样，CNN使用反向传播算法进行训练。

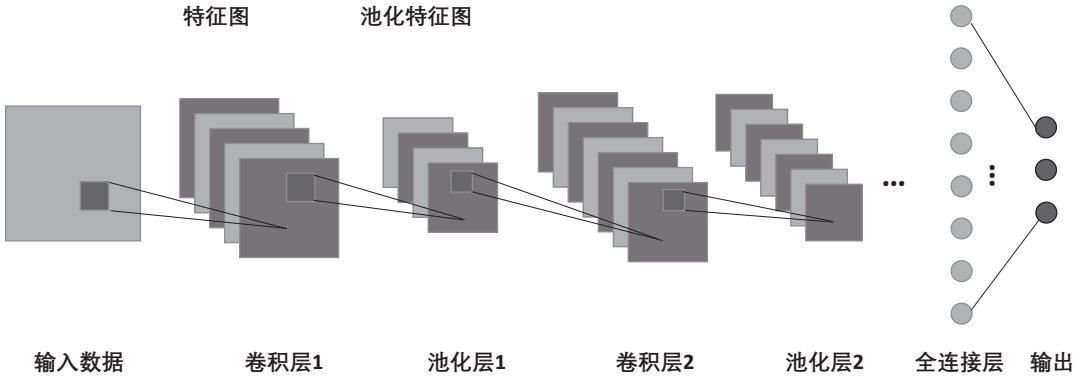


图 1-2 CNN结构示意图。

RNN是一种可以处理序列数据的神经网络。它将前一步骤的输出作为当前步骤的输入，是一种有记忆能力的网络。对于自然语言处理领域，在CNN或者传统的神经网络中，所有的输入和输出都是相互独立的，如果我们需要预测句子下一个单词的情况下，那么久需要记住前面的单词。RNN就是这样产生的，它利用一个隐层来解决这个问题。RNN 最主要和最重要的特征是隐藏状态，它能记住关于序列的一些信息。RNN通常会有一个记忆单元，它能记住所有计算过的信息。它对每个输入使用相同的参数，同时对所有输入或隐藏层执行相同的任务以生成输出。与其他神经网络不同，这降低了参数的复杂性。其中，长短期记忆网络（Long short term memory，LSTM）[32]是RNN中常见的一种模型，其结构如图 所示。LSTM中使用了一种特殊的隐藏单元被称为记忆细胞（Memory cell），它是一种可以在足够长的时间内保存其值，并使用以前时间步输出信息和当前时间步的输入信息都作为输入的函数，该作用是可以长时间记忆输入。细胞单元由输入门，输出门以及遗忘门组成，然后，使用反向传播算法通过组合以前的状态、当前记忆和输入来训练模型。事实证明，记忆细胞单元在捕获长期依赖关系方面非常有效。

近年来，CNN和RNN经过不断的发展，在计算机视觉，语音识别，自然语言处理等领域都取得了非常优秀的性能[31]。在计算机视觉领域，2012年Alex等人[33]设计出了一个深层的卷积神经网络AlexNet，获得了2012年ImageNet LSVRC的冠军，且分类正确率远超第二名（top5错误率为15.3%，第二名为26.2%），它的好处和优越性在于它的可扩展性和对图像处理器（Graphics processing unit，GPU）充分的

利用。由于使用了GPU, AlexNet具有很高的处理和训练速度，开创了深度神经网络高效训练的先河。2014 年牛津大学视觉图形组和Google DeepMind公司的研究员一起提出了VGGNet[34]，该模型由宽的底层和深的顶层组成，网络结构呈金字塔状。VGGNet研究了CNN的深度与其性能之间的关系，证明了增加网络的深度能够在一定程度上提高网络最终的性能。同年，Szegedy等人[35]提出了GoogLeNet，它包括22层，而VGGNet有16-19 层。GoogLeNet内部包含了一种叫Inception网络结构，通过一种“基础神经元”结构，来搭建一个稀疏性、高计算性能的网络结构。Inception结构凭借着其训练的高效性和在相同计算量下能提取出更多的特征，在之后的研究中也得到了进一步的发展[36–38]。随着深度神经网络的层数的逐渐加深，发现CNN网络达到一定深度后一味的继续增加层数并不能进一步提高分类器的性能，反而会导致网络收敛速度变得很慢。为了解决这个问题，2016年，He等人[39] 提出了深度残差网络ResNet，首次引入了残差模块，把网络层数扩展到了很深。ResNet由许多连续的残差模块组成，通过一种跳跃式的“捷径连接”的方式，直接把输入传到输出作为初始结果，相当于ResNet 将学习目标变为了目标值与输入值之间的差值，也就是所谓的残差。这种跳跃式的残差结构，打破了之前神经网络中后一层只与前一层相连的惯例，使前面层的输出可以跳跃几层后再作为后面层的输入。除了上述这些经典的CNN网络结构在图像分类上取得了优异的能力以外，CNN还在目标检测、图像分割领域都大放异彩。Girshick等人[40]在2014年提出了边缘卷积神经网络（Regions with convolutional neural network，RCNN），通过在图像中的对象上设计一个边界框，并识别图像中给定的对象。2016年，Redmon 等人[41] 提出了YOLO目标检测算法。YOLO 将图像划分为边界框，然后执行对所有框通的识别算法，最后进行框的合并，得到目标周围最佳的边界框。2017年，Redmon等人[42]又发表了YOLO v2，进一步提升了算法的检测精度和速度。在图像分割领域，Badrinarayanan等人[43]在2015年提出了SegNet，其核心由编码器网络、相应的解码器网络以及像素级分类层组成，SegNet在提高分辨率的同时对于边界的定位较为准确，在场景理解的任务中有着更高效的应用。同年，Ronneberger等人[44]提出了U-Net，基于编码器-解码器的结构，将不同层的特征进行拼接，结构简单且稳定，在医疗图像分割领域取得了非常不错的成绩。对于循环神经网络RNN，2013年，Graves等人[45]在提出了Deep LSTM RNNs，通过将多个LSTM作为声学和语言模型联合训练，在公开数据集上取得了当时最好的语音识别正确率。在2014年Graves等人[46]基于深度双向LSTM，直接用文本来转录语音数据，不需要中间语音表示法；他们设计出的语音识别系统与基准系统相结合，在华尔街日报语料库上取得了6.7%的错误率。2015年，Kalchbrenner 等人[47] 提出了Grid LSTM，使用该方法设计了一个新的二维翻译模型，在汉译英任务中对比其他方法取得了更好的结果。

除此以外，最近有多种深度学习模型，特别是基于CNN的深度学习模型被提出来用于脑电信号的分类。Lawhern等人提出了一种紧凑的CNN模型EEGNet[48]，使用

了深度可分离卷积（Depthwise and separable convolution）来代替传统卷积，用于脑机接口系统中脑电图特征的提取，该模型在几个基于EEG的脑机接口任务中取得了良好的表现。Schirrmeister等人[7]设计了一种深层的CNN模型（DeepCNN）和一种浅层的CNN模型（ShallowCNN）来进行端到端的脑电波解码，使得CNN的前面一些层能够学习到EEG信号的空间关系，更深的层学习到EEG信号的局部和全局调制的时序关系；实验表明，这两种网络结构对于不同任务的脑电信号特征提取有通用性，拥有很强的泛化能力，可以作为EEG信号解码的通用工具。除此以外，也有一些将EEG信号转换成图片，然后再使用深度神经网络模型进行EEG分类的工作[8–10]。为了与脑机接口系统的实际应用相结合，考虑到更常见的真实场景，本文主要研究的是接受原始脑电信号作为输入的CNN模型，具体来说，即上述的EEGNet、DeepCNN和ShallowCNN。

1.4 对抗攻击

尽快深度学习模型表现出色，但是它们非常容易受到对抗扰动的攻击[49]。在对抗攻击中，攻击者通过将精心设计的微小扰动（甚至很难被人类肉眼察觉）添加到正常的数据样本上，使得深度学习模型将这个样本分类错误，导致模型的整体性能得到大幅度的下降。这种加入了对抗噪声的数据样本，我们称之为对抗样本。对于图像分类领域，对抗扰动可以愚弄CNN分类器，使得预测结果发生巨大的改变，如图1-3所示。

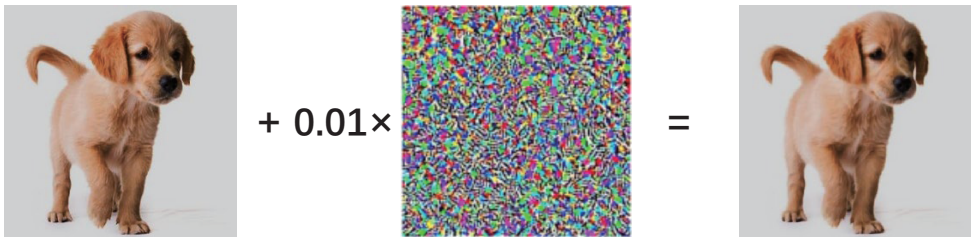


图 1-3 对抗扰动的作用。给原始分类为狗的图片加入人眼不容易察觉的扰动后，被CNN分类器分类为鱼。

2013年，Szegedy等人[49]首次发现了对抗攻击这一现象，很快就受到了极大的关注。2014 年，Goodfellow等人[50]成功的愚弄了一个深度学习模型，让模型将一个原本被正确分类为熊猫的图片，在加入了对抗噪声之后，被错误的分类到了长臂猿。Kurakin等人[51]发现深度学习系统甚至会在预测已经打印出来的对抗样本的照片时出错，这一发现表明对抗攻击是容易被应用在实际生活中的。Brown等人[52]提出了一

种对抗补丁的方法，能够在图片的任意位置加入一个补丁，使得深度学习模型对该图片预测错误。Athalye等人[53]提出了一种期望转换的方法，使得深度学习模型从不同的观测角度和随机的拍摄姿势，都能够将一个3D打印出来的海龟预测成来复枪。除此以外，对抗攻击也可以被应用到语音识别系统中，例如，攻击者给一段语音加入了对抗噪声，这段语音听上去与正常的语音相同，但是语音识别系统会将这段语音信号识别成其他短语，甚至是攻击者任意想要的短语[54]。

最近有一些关于通用对抗扰动的研究。Moosavi-Dezfooli等人[55]第一次发现了通用对抗扰动的存在，他们使用DeepFool[56]来解决一个复杂的优化问题，构建出了通用对抗扰动，并且表明他们的方法能够愚弄在图像分类领域中最先进的机器学习模型。相同的思想被应用到了语音识别系统[57]。Behjati等人[58]提出可一种基于梯度投影的方法在文本分类任务中来构建通用扰动。Mopuri等人[59]提出了一种在独立于数据、有泛化能力的通用对抗扰动，并在不同的图像任务中都取得了不错的攻击效果。

根据攻击者对于目标模型信息的访问程度，攻击方式可以分为以下三种：

(1) 白盒攻击。假定攻击者可以访问目标模型的所有信息，包含其结构以及参数，那么攻击者可以针对这个模型设计特定的对抗扰动，这种攻击方式称为白盒攻击。白盒攻击的算法一般是基于优化策略或者梯度策略，例如L-BFGS[49]，DeepFool[56]，C&W方法[60]，快速梯度符号法[50]（Fast gradient sign method，FGSM），基本迭代法（Basic iterative method，BIM）[51]等等。

(2) 黑盒攻击。这种攻击方式假定攻击者完全不知道目标模型的结构和参数，只能观察到模型对于输入数据的响应，整个目标模型是一个黑盒子。Papernot等人[61]在2016年提出了一种黑盒攻击方法，该方法可以通过与目标模型交互并训练一个替代模型来生成对抗样本。Su等人[62] 在2017年通过改变图像的一个像素点，成功的愚弄了三个不同的模型。Brendel等人在2017年提出了一种黑盒攻击方法，通过从一个大的对抗扰动开始，然后尝试减少扰动大小的同时，保持扰动的对抗性。

(3) 灰盒攻击。假定攻击者只知道目标模型的一部分信息，但不是全部的信息，例如，攻击者有权限访问训练目标模型所需要的训练数据，但不知道目标模型的架构和参数，因此攻击者只能根据训练数据来优化对抗扰动。这种攻击方法通常是攻击者在已知的训练集上训练出一个替代模型，根据替代模型来使用白盒攻击的方法来生成出对抗扰动，然后再应用到目标模型上。

最新研究表明，Zhang和Wu[63]第一次发现在基于EEG信号的脑机接口系统中同样存在对抗样本。在3 种不同的应用场景（白盒攻击、灰盒攻击和黑盒攻击）中，他们的方法对3个CNN分类器（EEGNet，DeepCNN和ShallowCNN）都攻击成功。他们的实验结果表明了在基于EEG信号的脑机接口系统中存在一个关键的安全问题，这是以往没有相关研究注意到的。正如他们在文中指出：基于脑电图的脑机接口系统可用于控制轮椅或外骨骼的残疾[64]，在那里，对抗性攻击可能使轮椅或外骨骼

故障。其后果可能从简单的使用户产生困惑或者沮丧的心情，到显著降低用户的生活质量，甚至可能通过对抗攻击故意将用户推入危险的环境中来伤害用户。在意识障碍患者意识评估和检测的临床应用中，对抗性攻击可能导致脑机接口系统产生误诊。

1.5 本文主要研究内容

尽管Zhang和Wu在脑机接口中的分类问题中取得了不错的对抗攻击效果[63]，但是他们的方法还存在以下的约束：

(1) 每个输入的脑电图信号都需要专门计算一个特定的对抗扰动，这在实际应用中不方便。

(2) 为了计算出特定的对抗扰动，攻击者需要提前知道完整的EEG信号，因此不可能在一开始进行EEG信号传输时就进行实时的攻击。

这两项约束可能会导致一般的样本依赖的对抗攻击方法需要等待一个完整的EEG信号传输结束，才能设计出合适的对抗扰动然后才能添加到原始信号上。这样的操作会出现**因果性**的问题：EEG信号是时序信号，等待信号传输完毕后，脑机接口系统已经接受到之前的数据，无法再进行数据的篡改操作，从而无法给原始EEG信号添加对抗扰动。因此，我们期望能够设计出一种可以实时添加到EEG信号中的特殊扰动，考虑到通用对抗扰动（Universal adversarial perturbation，UAP）是一种可以提前进行离线计算并且对于所有数据完全相同的扰动，这种扰动的攻击和防御为本文的主要研究内容。

本文主要研究了脑机接口分类问题中的UAP，这是一种离线计算的通用扰动模板，只要确定信号的起始位置之后，它能够实时的被加入到任意的EEG信号中，从而不用担心信号因果性的问题，如图1-4所示。UAP能够同时解决上文中在提到的Zhang和Wu所提出的方法存在的两种约束。

本文研究了基于脑电图EEG的脑机接口系统中的通用对抗扰动。我们作出了以下四项贡献：

(1) 我们首次将UAP的思想引入到了基于EEG的脑机接口系统中，针对EEG信号设计了基于DeepFool的的UAP攻击方法（DF-UAP），在不同的应用场景下成功的攻击了3个CNN分类器，使得模型的分类准确率显著下降。这项工作使得对抗攻击在脑机接口系统中会更加的灵活和方便的应用，能够实时进行攻击。

(2) 我们提出了一种总损失最小（Total loss minimization，TLM）的方法来给EEG信号设计UAP（TLM-UAP）。与DF-UAP相比，我们提出的TLM-UAP能够在扰动更小的情况下取得更强的攻击性能，并且拥有更好的扩展性和通用性，能够适用于不同应用场景。

(3) 我们提出的TLM方法首次将UAP从非目标攻击（不用将样本分类位指定类

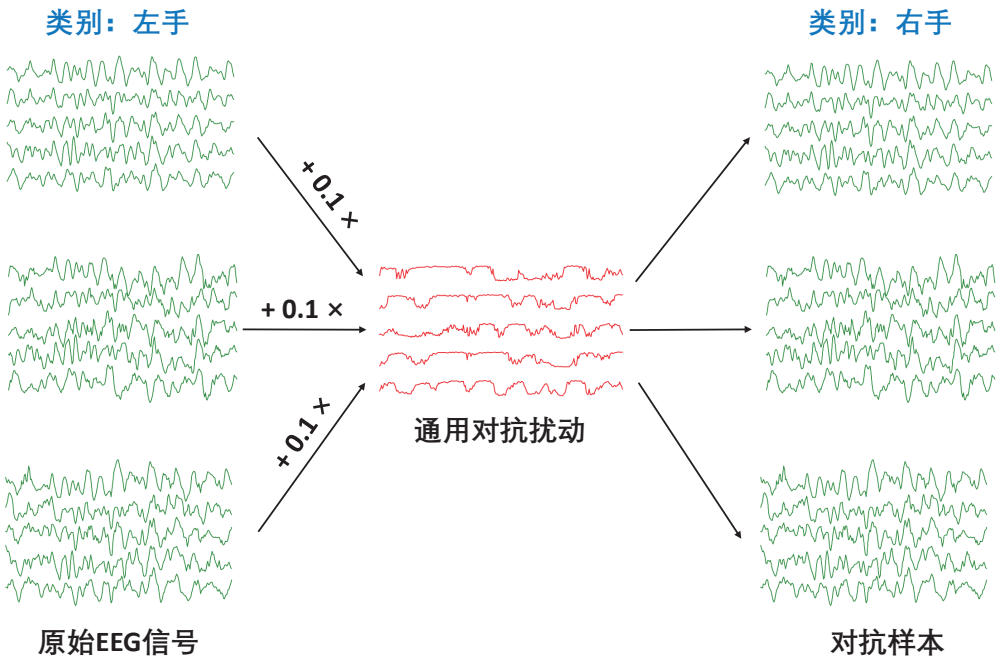


图 1-4 给原始EEG信号添加通用对抗扰动。每个EEG信号添加的对抗扰动完全相同，且不容易被人眼察觉，添加通用对抗扰动之后，原本被分类为左手的EEG信号被脑机接口系统中的CNN模型分类为右手响应。

别）扩展到了目标攻击（需要将样本分类为指定类别），并在目标攻击场景中取得了接近100%的目标率（样本被分类到指定类别占总体样本的比率）。之前的一些设计通用扰动的方法都只能进行非目标攻击，据我们所知，这是第一项进行目标攻击的UAP研究。

(4) 我们首次在脑机接口系统中进行了UAP防御机制的研究。从通用对抗样本的检测和模型对抗鲁棒性两个方面对UAP防御能力进行了分析，通过搭建UAP检测模块和使用投影梯度下降（Projected gradient descent，PGD）方法训练模型这两种方式能够分别在不同应用场景中成功防御UAP。据我们所知，还没有人在脑机接口系统中做针对UAP防御机制的研究，本文的工作将对设计出更安全的脑机接口分类器提供帮助。

1.6 本文组织结构

全文共五章，第二章至第四章为本文的主要内容，是作者硕士阶段，在脑机接口分类问题中的UAP研究的主要工作。论文的主要内容概括如下：

第二章介绍了一种样本依赖的对抗攻击算法DeepFool和基于DeepFool的UAP生成算法。然后通过实验验证了该算法生成的DF-UAP在不同的应用场景下成功的攻击了3个CNN分类器，使得模型的分类准确率得到了显著的下降，表明了UAP在脑机

接口系统中攻击的可行性。我们首次将UAP的思想引入到了基于EEG的脑机接口系统中，这项工作使得在脑机接口系统中对抗攻击会更加的实用和方便。

第三章介绍了我们提出的TLM-UAP设计算法，并将该算法从非目标攻击扩展到了目标攻击，使攻击者使用我们提出的算法生成的TLM-UAP攻击基于EEG数据的CNN分类器后，能够将所有数据样本分类到任意他们所指定的分类结果。我们通过实验表明了在非目标攻击场景，我们提出的TLM-UAP对比DF-UAP能够在扰动更小的情况下取得更好的攻击性能。然后对TLM-UAP的特性以及迁移性进行了具体地分析和研究。最后我们使用TLM-UAP设计了全通道相同的扰动，但是攻击效果不显著。我们提出的TLM-UAP设计算法同时考虑到了非目标攻击和目标攻击着两种不同的应用形式，之前的生成UAP方法只能进行非目标攻击。据我们所知，这是第一项进行目标攻击的通用对抗扰动攻击的研究。

第四章介绍了UAP的防御机制，我们首先考虑了理想条件下（已知某些数据是对抗样本），通过搭建UAP检测模块来实现对抗样本的识别，然后我们在真实场景下（未知某些数据是对抗样本）通过PGD的方法来训练出了对于UAP更鲁棒的模型。据我们所知，还没有人在脑机接口系统中来做针对UAP检测和防御的研究，我们期望这项工作能够对设计出更安全的脑机接口系统提供帮助。

本文框架如图1-5所示。

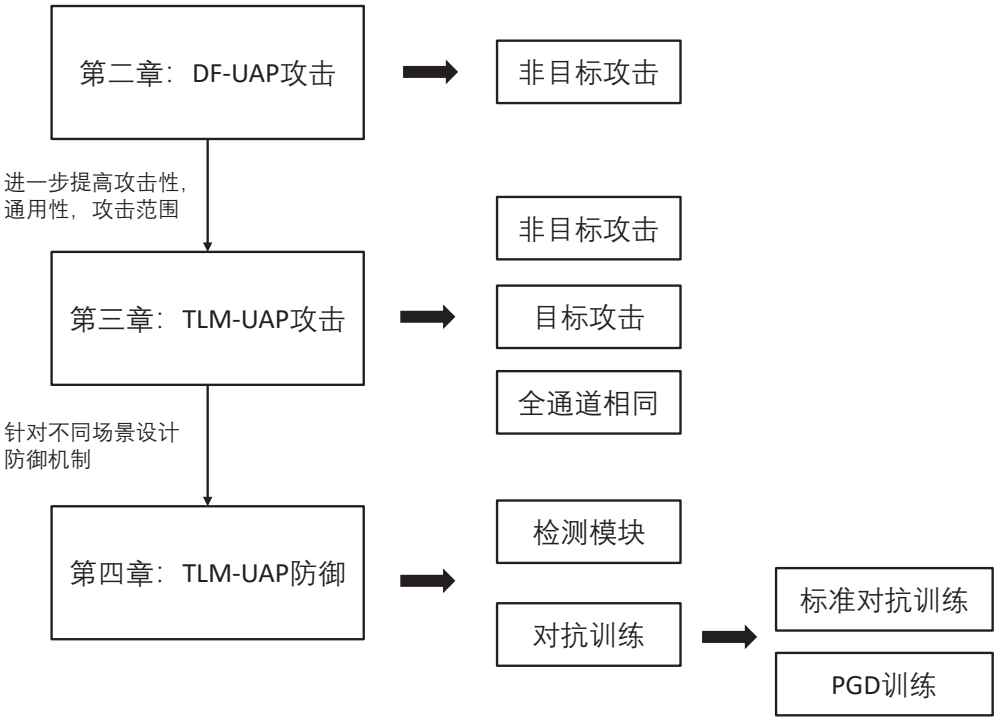


图 1-5 本文框架。

2 基于DeepFool的通用对抗扰动

常用的对抗攻击方法都是基于样本独立的：针对每个数据样本都设计出不一样的对抗扰动，这样的攻击方法往往能取得非常优异的攻击效果，但是由于需要考虑到系统的实时性和数据的因果性，这些攻击手段在脑机接口系统的实际应用并不方便进行实现（传统的对抗扰动不太容易进行实时的添加，需要等待一个完整的数据样本输入结束）。因此，我们在脑机接口系统中引入了通用对抗扰动的思想，给所有的脑电波数据都设计出相同的对抗扰动，使得添加对抗扰动后的大部分数据都能被神经网络模型分类出错，达到实时攻击脑机接口系统的目的。本章首先介绍了一种针对单个数据样本的对抗攻击方法DeepFool，然后基于DeepFool对EEG数据设计出了一种通用对抗扰动，使得对抗攻击在脑机接口系统中更加的实用和方便。

2.1 问题设置

为了能够成功攻击一个脑机接口系统，对抗扰动需要被实时的加入到原始的EEG信号中。

我们定义$X_i \in \mathbb{R}^{C\times T}$是第$i$个原始EEG样本，其中$C$是EEG信号的通道数，T是EEG 信号的时长。为了方便后文的表述，我们定义$\boldsymbol{x} \in \mathbb{R}^{C\cdot T\times 1}$是$X_i$的向量形式，即将$X_i$的所有列拼接成一列。$k(\boldsymbol{x}_i)$为目标CNN 模型的预测估计标签，$\boldsymbol{v} \in \mathbb{R}^{C\dot{T}\times 1}$为通用对抗扰动，$\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i + \boldsymbol{v}$为加入了通用对抗扰动后的对抗EEG样本。那么，一个能够成功愚弄分类器的通用对抗扰动$\boldsymbol{v}$需要满足如下条件：

$$\left.\begin{array}{rcl} \frac{1}{n}\sum_{i=1}^{n} I\left(k(\boldsymbol{x}_i+\boldsymbol{v}) \neq k(\boldsymbol{x}_i)\right) & \geq & \delta \\ \|\boldsymbol{v}\|_p & \leq & \xi \end{array}\right\}, \tag{2.1}$$

其中，$\|\cdot\|_p$表示L_p范数，$I(\cdot)$是指示函数，当内部条件满足时，值为1，否则为0。参数$\delta \in (0,1]$表示我们设置的期望的攻击成功率（Attack success rate，ASR），参数ξ用来约束扰动$\boldsymbol{v}$ 大小。简单来讲，上诉第一个约束式子要求通用对抗扰动能够达到我们期望的攻击成功率，第二个式子确保通用对抗扰动要很小，不容易被人察觉。

然后，下面我们将介绍如何针对EEG数据构建出一个可行的通用对抗扰动。我们首先介绍了一种DeepFool 的白盒攻击方法（攻击者需要拥有目标模型的所有信息，包含其模型结构及参数），这种方法是针对每一个数据样本都设计出特定的不相同的对抗扰动。然后基于DeepFool引入了一种迭代式的通用对抗扰动生成方法，为整个EEG数据设置出一种通用扰动，能够使大多数数据在加入通用对抗扰动之后被目标模型分类错误。

2.2 DeepFool的对抗攻击算法

DeepFool是一种给单个数据样本设计出对抗扰动的攻击方法。

我们首先考虑二分类问题，假设所有数据样本的类别标签属于$\{-1,1\}$，定义$\boldsymbol{x}$为一个数据样本，$f(\boldsymbol{x})=\boldsymbol{w}^T\boldsymbol{x}+b$是一个分类的映射函数，其中$\boldsymbol{w}$是函数的权重，$b$是函数的偏置，那么数据的预测标签由映射函数$f$的符号进行判断，即：

$$k(\boldsymbol{x})=\operatorname{sign}f(\boldsymbol{x}) \tag{2.2}$$

那么，能够将样本$\boldsymbol{x}$移动到决策超平面$\mathcal{F}=\{\boldsymbol{x}^*:\boldsymbol{w}^T\boldsymbol{x}^*+b=0\}$的最小的对抗扰动$\boldsymbol{r}^*$为：

$$\boldsymbol{r}^*=-\frac{f(\boldsymbol{x})}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w}. \tag{2.3}$$

由于CNN分类器是非线性的，而我们上述考虑到的场景是线性分类器，因此，Moosavi-Dezfooli等人[56]提出了一种名为DeepFool的迭代策略来构建对抗扰动。该方法通过在第t次迭代中，假设映射函数f在数据样本$\boldsymbol{x}_t$ 周围是近似线性的，那么，在第t 次迭代中的最小对抗扰动可以按如下公式计算。

$$\begin{aligned}&\arg\min_{\boldsymbol{r}_t}\ \|\boldsymbol{r}_t\| \\ &\text{s.t. }\ f(\boldsymbol{x}_t)+\nabla f(\boldsymbol{x}_t)^T\boldsymbol{r}_t=0.\end{aligned} \tag{2.4}$$

根据公式(2.3)，我们计算出在第t次迭代中的扰动$\boldsymbol{r}_t$，然后在下一轮迭代中，新的数据样本按如下式子更新：

$$\boldsymbol{x}_{t+1}=\boldsymbol{x}_t+\boldsymbol{r}_t \tag{2.5}$$

当更新后的$\boldsymbol{x}_t$ 能够开始改变最终的分类标签时，迭代停止。DeepFool的伪代码如算法 1 所示：

算法1能够通过使用one-vs-all策略来寻找出离数据样本最近的分割超平面，将DeepFool 从二分类问题扩展到多分类问题。

文献[56]将DeepFool同时推广到了多分类问题。在这种场景下，分类器有c个输出的预测概率，其中，c是类别的数量。因此，分类器f 被定义为映射函数$f:\mathbb{R}^n\to\mathbb{R}^c$，最后预测的分类结果通过下式可得：

$$\hat{k}(\boldsymbol{x})=\arg\max_k f_k(\boldsymbol{x}) \tag{2.6}$$

其中，$f_k(\boldsymbol{x})$对应于$f(\boldsymbol{x})$第k类的输出，即分类器将输入$\boldsymbol{x}$分类为第k类的预测概率，$f_k(\boldsymbol{x})$也可以看做是第k个子分类器。与二分类问题的情况相似，文献[56]先考虑分类器为线性场景进行推导，然后再扩展到非线性的场景下。

Algorithm 1: DeepFool算法伪代码.

Input: $\boldsymbol{x}$, 一个输入数据样本;
f, 一个二分类器.
Output: $\boldsymbol{r}^*$, 通用对抗扰动UAP.
初始化$\boldsymbol{x}_0 = \boldsymbol{x}$;
初始化迭代次数$t = 0$;
while $\mathrm{sign} f(\boldsymbol{x}_t) = \mathrm{sign} f(\boldsymbol{x}_0)$ **do**
  计算扰动$\boldsymbol{r}_t = -\frac{f(\boldsymbol{x}_t)}{\|\nabla f(\boldsymbol{x}_t)\|_2^2}\nabla f(\boldsymbol{x}_t)$;
  更新数据$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \boldsymbol{r}_t$;
  $t = t + 1$;
end
对抗扰动$\boldsymbol{r}^* = \sum_{i=0}^{t} \boldsymbol{r}_i$.

首先，考虑分类器是多分类线性场景，假设$f(\boldsymbol{x})$是一个线性分类器的映射函数，即：

$$f(\boldsymbol{x}) = \mathbf{W}^T\boldsymbol{x} + b \tag{2.7}$$

其中，$\mathbf{W}$和b分别对应线性分类器的权重矩阵和偏置。我们定义$\hat{k}$是one-vs-all的多分类形式，如公式(2.6)所示。要使分类器的分类结果改变，需要保证在加入扰动之后至少存在一个非原始类别的预测概率大于原始类别的分类函数结果，因此能够愚弄分类器的最小扰动$\boldsymbol{r}$ 可以由下式可得：

$$\begin{aligned} &\arg\min_{\boldsymbol{r}} \ \|\boldsymbol{r}\|_2 \\ &\text{s.t. } \boldsymbol{w}_k^T(\boldsymbol{x}_0 + \boldsymbol{r}) + b_k \geq \boldsymbol{w}_{\hat{k}(\boldsymbol{x}_0)}^T(\boldsymbol{x}_0 + \boldsymbol{r}) + b_{\hat{k}(\boldsymbol{x}_0)} \end{aligned} \tag{2.8}$$

其中$\boldsymbol{w}_k$是权重矩阵$\mathbf{W}$的第k个列向量，即第k个子分类器$f_k(\boldsymbol{x})$的权值向量。那么第k个分类边界为$\mathcal{F}_k = \{\boldsymbol{x} : f_k(x) - f_{\hat{k}(\boldsymbol{x}_0)}(x) = 0\}$，从几何上来看，求解上式对应于计算出$x_0$ 与凸区域P 边界的距离，P为:

$$P = \bigcap_{k=1}^{c} \{\boldsymbol{x} : f_{\hat{k}(\boldsymbol{x}_0)}(x) \geq f_k(x)\} \tag{2.9}$$

其中x_0在区域P内部。如图2-1所示，凸区域代表当f输出原始标签$\hat{k}(\boldsymbol{x}_0)$所在的区域空间（由各分类边界围绕）。

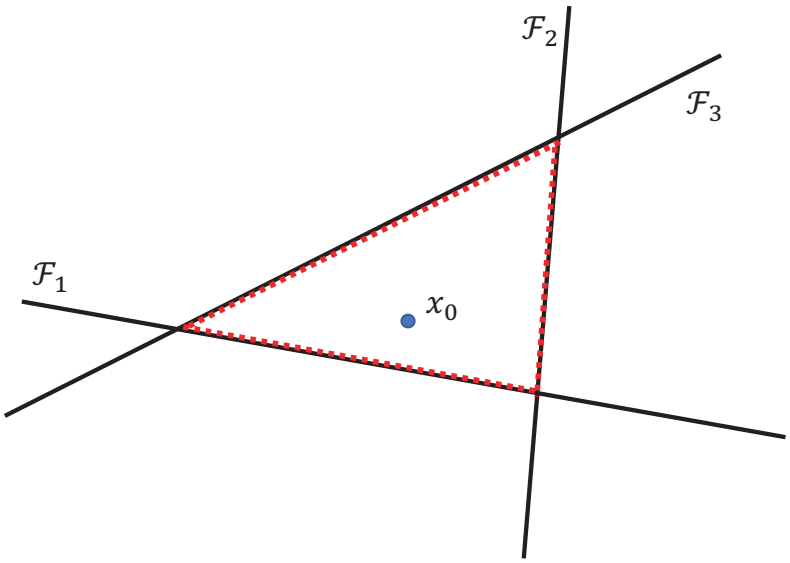


图 2-1 凸区域P示意图。这里假设$\boldsymbol{x}_0$属于类别4，则其它类别的分类超平面$\mathcal{F}_k = \{\boldsymbol{x} : f_k(x) - f_4(x) = 0\}$。实线表示超平面$\mathcal{F}_k$，红色虚线表示P的边界。

因此求解公式(2.8)等价于求解$\boldsymbol{x}_0$离其他边界（非原始类别边界）的最短距离，我们定义该最短距离$\hat{l}(\boldsymbol{x}_0)$为：

$$\hat{l}(\boldsymbol{x}_0) = \arg\min_{k \neq \hat{k}(\boldsymbol{x}_0)} \frac{|f_k(\boldsymbol{x}_0) - f_{\hat{k}(\boldsymbol{x}_0)}(x_0)|}{\|\boldsymbol{w}_k - \boldsymbol{w}_{\hat{k}(\boldsymbol{x}_0)}\|_2} \tag{2.10}$$

那么，能够改变数据点$\boldsymbol{x}_0$分类结果的最小扰动$r(\boldsymbol{x}_0)$为将$\boldsymbol{x}_0$在超平面上沿着$\hat{l}(\boldsymbol{x}_0)$方向上的投影向量，即：

$$r(\boldsymbol{x}_0) = \frac{|f_{\hat{l}(\boldsymbol{x}_0)(\boldsymbol{x}_0)} - f_{\hat{k}(\boldsymbol{x}_0)}(x_0)|}{\|\boldsymbol{w}_{\hat{l}(\boldsymbol{x}_0)} - \boldsymbol{w}_{\hat{k}(\boldsymbol{x}_0)}\|_2^2} (\boldsymbol{w}_{\hat{l}(\boldsymbol{x}_0)} - \boldsymbol{w}_{\hat{k}(\boldsymbol{x}_0)}) \tag{2.11}$$

换句话说，我们找到了数据点$\boldsymbol{x}_0$在凸区域P上的最近投影。

之后，通过迭代的思想，我们可以将DeepFool从线性场景推广到非线性：在每轮迭代更新扰动$r(\boldsymbol{x}_0)$时，假定数据点$\boldsymbol{x}_0)$在各个分类边界函数上是线性可导的，然后逐渐逼近真实分类边界。所以我们使用区域P_i来近似估计在第i次迭代中的真实凸区域P：

$$P = \bigcap_{k=1}^{c} \{\boldsymbol{x} : f_{\hat{k}(\boldsymbol{x}_0)}(x) \geq f_k(x)\} \tag{2.12}$$

具体来讲，在DeepFool算法的每次迭代中，计算到达多面体区域P_i边界的扰动向量，并更新当前估计的扰动向量。需要注意的是，DeepFool算法以贪婪的方式运行，不能保证收敛到公式(2.8)中的最优扰动。文献[56]表示DeepFool可以生成出非常小的对抗扰动，使得分类器结果改变，我们认为该算法对最小扰动有着很好的近似。

除此以外，DeepFool还可以通过简单的调整更新式子，从l_2扩展到任意l_p范式（$p \in [1, \infty)$）。在文献[56]中的实验表明，DeepFool能在扰动更小的情况下，对常见的4种CNN模型取得与FGSM[50]相近的攻击效果。

2.3 基于DeepFool的通用对抗扰动

Moosavi-Dezfooli等人[55]在图像分类领域提出了一种通用对抗扰动（Universal adversarial perturbation，UAP），这种扰动能够在大多数图像上欺骗最先进的CNN分类器，即在加入相同的通用对抗扰动之后，可以使CNN分类器对于大部分图像分类错误。Moosavi-Dezfooli等人基于DeepFool提出了一种迭代式的算法来生成通用对抗扰动。因此，通用对抗扰动是根据多个数据样本来设计的，使之满足条件(2.1)，能够让大部分数据样本分类错误。

在一个脑机接口系统中，信号的采集与读取与一般的图像分类系统是不同的。图像数据往往是非时序信号（读取一张图片，数据即输入完毕），但是脑机接口信号通常是一种时序信号，随着时间来不断读取数据，这样就会导致传统的为单一样本设计对抗扰动的攻击方法存在因果性的问题：如果等待脑机接口信号输入完毕再去设计对抗扰动，数据信号已经被系统读取完毕，很难再次将设计好的扰动添加到原始信号中。正如2.1 部分问题设置所示，为了能够成功攻击一个脑机接口系统，对抗扰动需要被实时的加入到原始的EEG信号中。在本文第一章中我们提到传统的针对单一样本设计出对抗扰动方法（例如DeepFool，FGSM等）在实际的脑机接口中应用可能会存在问题：1.每个输入的脑电图信号都需要专门计算一个特定的对抗扰动；2.为了计算出特定的对抗扰动，攻击者需要提前知道完整的EEG信号，因此不可能在一开始进行EEG信号传输时就进行实时的攻击。因此，我们期望设计出一种能够无视数据样本，有泛化能力的通用对抗扰动，然后通过给EEG信号添加一个通用模板的形式来将该对抗扰动添加到脑机接口系统中。

首先，一个通用对抗扰动是针对数据集$\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ 中的所有样本来被特定设计出来的，作为一种迭代式的生成算法，在每次迭代中，我们使用DeepFool来计算当前的单一扰动点（加入了对抗扰动之后的数据点）$\boldsymbol{x}_i + \boldsymbol{v}$ 的最小对抗扰动$\triangle \boldsymbol{v}_i$，然后将扰动$\triangle \boldsymbol{v}_i$逐渐累加得到最终的通用对抗扰动$\boldsymbol{v}$，直到该扰动能使大部分数据点分类错误，如图2-2 所示。

更具体来讲，如果当前的通用对抗扰动$\boldsymbol{v}$在数据点$\boldsymbol{x}_i$上不能欺骗分类器，为了使数据点$\boldsymbol{x}_i$分类错误，那么我们通过计算如下式所示的优化问题来计算得到一个额外的最小扰动$\triangle \boldsymbol{v}_i$：

$$\min_{\triangle \boldsymbol{v}_i} \| \triangle \boldsymbol{v}_i \|_2, \quad \text{s.t. } k(\boldsymbol{x}_i + \boldsymbol{v} + \triangle \boldsymbol{v}_i) \neq k(\boldsymbol{x}_i). \tag{2.13}$$

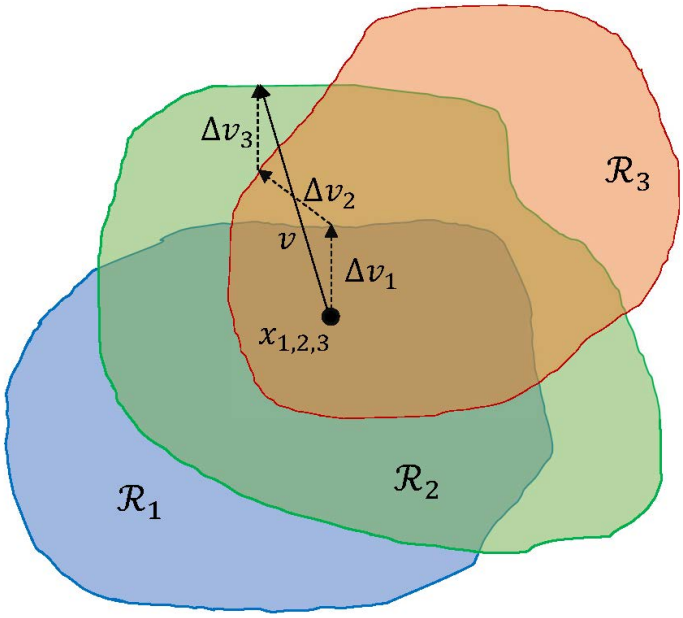


图 2-2 计算通用对抗扰动UAP的算法示意图。在本示意图中，将数据点$\boldsymbol{x}_1$，$\boldsymbol{x}_2$和$\boldsymbol{x}_3$叠加在一起，分类区域$\mathcal{R}_i$使用不同的颜色表示，最终的扰动$\boldsymbol{v}$通过不断累加$\triangle\boldsymbol{v}_i$使得数据点$\boldsymbol{x}_i$跳出原始分类区域。

为了约束通用对抗扰动的大小，使之满足$\|\boldsymbol{v}\|_p \leq \xi$，更新后的通用扰动需要投影到中心为0，半径为$\xi$的超球体上（$\ell_p$范数），然后根据投影函数的约束条件求出超球体内部离当前扰动最近的新扰动，投影函数$\mathcal{P}_{p,\xi}$定义如下：

$$\mathcal{P}_{p,\xi}(\boldsymbol{v}) = \arg\min_{\|\boldsymbol{v}'\|_p \leq \xi} \|\boldsymbol{v} - \boldsymbol{v}'\|_2. \tag{2.14}$$

然后，在每次迭代中，通用对抗扰动通过$\boldsymbol{v} = \mathcal{P}_{p,\xi}(\boldsymbol{v} + \triangle\boldsymbol{v}_i)$ 进行更新。这个迭代过程将在整个数据集上一直持续，直到达到设置的最大迭代次数，或者期望的攻击成功率ASR 超过设定的阈值$\delta \in (0, 1]$，即：

$$ASR(\boldsymbol{X}_v, \boldsymbol{X}) = \frac{1}{n}\sum_{i=1}^{n} I\left(k(\boldsymbol{x_i} + \boldsymbol{v}) \neq k(\boldsymbol{x_i})\right) \geq \delta. \tag{2.15}$$

基于DeepFool来生成通用对抗扰动的算法伪代码如算法2所示。该算法在每一次迭代过程中，最多求解公式(2.13)中的m个数据样本的优化问题。当k是一个标准的分类器时（例如深度神经网络），该优化问题往往不是一个凸优化问题，但是存在一些有效的近似方法来解决该问题[49,56]，因此我们采用了DeepFool[56]作为求解方案。值得一提的是，算法2的目标不是为了找出能够使大多数数据点分类错误的最小通用扰动，而是找到一个满足大小约束且足够有效的通用扰动。

Algorithm 2: 基于DeepFool的通用对抗扰动设计算法伪代码。

Input: $\boldsymbol{X}=\{\boldsymbol{x}_i\}_{i=1}^{n}$，$n$个输入数据样本;
k，分类器;
ξ，UAP大小约束参数ℓ_p;
δ，期望目标攻击率ASR;
M，最大迭代次数。
Output: $\boldsymbol{v}$，一个通用对抗扰动。
初始化$\boldsymbol{v}=\mathbf{0}$;
初始化$\boldsymbol{X}_v=\boldsymbol{X}$;
for $m=1,...,M$ **do**
  if $ASR(\boldsymbol{X}_v,\boldsymbol{X})<\delta$ **then**
    for 每个$\boldsymbol{x}_i\in\boldsymbol{X}$ **do**
      if $k(\boldsymbol{x}_i+\boldsymbol{v})==k(\boldsymbol{x}_i)$ **then**
        使用DeepFool计算在公式(2.13)中的最小扰动$\triangle\boldsymbol{v}_i$;
        通过公式(2.14)更新扰动: $\boldsymbol{v}\leftarrow\mathcal{P}_{p,\xi}(\boldsymbol{v}+\triangle\boldsymbol{v}_i)$;
      end
    end
    $\boldsymbol{X}_v=\{\boldsymbol{x}_i+\boldsymbol{v}\}_{i=1}^{n}$;
  else
    Break;
  end
end
Return $\boldsymbol{v}$.

2.4 实验部分

该部分主要介绍了实验使用的BCI数据集、攻击的CNN模型、具体的实验设置以及DF-UAP的攻击性能。

2.4.1 BCI数据集

P300 evoked potentials（P300）数据集: 该数据集由文献[65]首次提出，一共采集了8个被试者用户的脑电数据，是一个视觉P300范式数据集。在采集实验中，每个被试者面对一台笔记本电脑，电脑屏幕上随机闪过6 张图片，通过电极采集得到被试者的P300响应。这个数据集的目的是通过采集到被试者的EEG信号来判断被试者当前看到的图片是目标/非目标（Target/Non-target）。其中，32个通道的EEG信号被降采样到256Hz，然后使用带通滤波器提取出$[1,40]$Hz 的信号，最后记录每张图片闪过$[0,1]$s的EEG信号作为数据样本。我们使用$\frac{x-\text{mean}(x)}{10}$对数据集进行了归一化处理，

并把结果值裁剪到$[-5,5]$。每个用户包含了3300个数据样本，整个P300数据集一共有$3300\times 8=26400$个样本。

Feedback error-related negativity（ERN）数据集： ERN数据集[66]是来自于Kaggle竞赛①的一个比赛数据集。该数据集采集了26个用户的EEG信号数据，分为差反馈/良好反馈（Bad feedback/Good feedback）两种类别。整个数据集被划分为一个训练集（16个用户）和一个测试集（10个用户）。由于比赛官方没有公开测试集，我们只使用了包含16个用户训练集数据。我们将56通道的EEG信号降采样到200Hz，然后同样使用带通滤波器提取了$[1,40]$Hz 的信号，使用用户在接受刺激后的$[0,1.3]$的脑电信号作为数据样本，并对数据进行了Z-score标准化。ERN数据集一共包含5440个样本，每个用户拥有340个样本。

Motor imagery（MI）数据集： MI数据集来自于BCI Competition IV[67]中的Dataset 2A②，是一种运动想象数据集。该数据集采集了9位用户的EEG信号，用户通过大脑想象肢体的运动，分为4个类别：左手，右手，双脚和舌头。我们将22通道的EEG信号降采样至128Hz，使用带通滤波器得到$[4,40]$Hz的信号，然后提取每个想象后的$[0,2]$s作为数据样本，并使用了衰减系数为0.999 的指数滑动平均窗口对数据进行了标准化。每个用户包含576个样本（4个类别，每个类别有144次试验），一共有5184个样本。

2.4.2 CNN模型

为了与脑机接口系统的实际应用相结合，考虑到更常见的真实场景，本文主要研究的是接受原始脑电信号作为输入的CNN分类模型，具体细节如下：

EEGNet： EEGNet[48]是专门针对基于EEG信号的脑机接口系统而设计的一个紧凑的CNN模型，它由两个卷积模块以及一个分类模块组成。为了减少模型的参数量，EEGNet使用了深度可分离卷积[68]来代替CNN中传统的卷积。

DeepCNN： DeepCNN[7]包含四个卷积模块和一个用于分类的softmax模块，该模型比EEGNet 模型层数要更深。它的第一个卷积模块是经过专门设计来用于处理EEG信号输入，其余三个卷积模块则与标准的卷积模块相同。

ShallowCNN： 灵感来源于滤波器中公共空间模式（Common spatial pattens，CSP），ShallowCNN[7]是一种专门用于解码频带能量特征的模型，与DeepCNN相比，ShallowCNN的卷积层中使用了更大的时间卷积核，并在之后接上了空间滤波器、平方非线性处理、平均池化层和对数激活函数。

2.4.3 实验设置

在本文的实验部分中，我们考虑到了两种实验设置：

① https://www.kaggle.com/c/inria-bci-challenge
② http://www.bbci.de/competition/iv/

Within-subject实验：在每个数据集中，我们将每个单独的用户的EEG数据样本随机打乱，然后划分80%的数据作为训练集，20%的数据作为测试集。其中，我们从训练集中随机选择出25%的数据作为验证集，来实现早停（Early stopping）。

Cross-subject实验：对于每个数据集，我们使用留一法（Leave-one-subject-out）来进行交叉验证，得到最终的实验结果。例如，我们假设一个数据集有N个用户，那么第N个用户的数据将作为测试集，之前的$N-1$个用户的数据将全部合在一起并进行随机打乱，然后我们将这$N-1$个用户的数据取75% 作为训练集，25%作为验证集来实现早停。

由于P300数据集和ERN数据集存在类别不平衡的问题，为了减少类别不平衡对模型分类精度的影响，我们在训练目标模型的过程中，根据不同的类别的样本在训练集中所占比例的倒数，给不同类别的数据分别加入了类别权重。在目标模型训练过程中，我们使用交叉熵作为损失函数，选择Adam作为梯度下降的优化器，同时使用了早停来减少模型的过拟合。

2.4.4 性能指标

在本文中，我们使用了原始分类准确率（Raw classification accuracy，RCA）和平衡分类准确率（Balanced classification accuracy，BCA）作为性能指标。RCA表示被分类正确的数据样本占全部样本的比率，BCA表示在不同类别中RCA的平均值。

值得一提的是，统计BCA指标在本文中是有必要的，因为在很多脑机接口范式（例如P300）中存在明显的类别不平衡，因为只评价RCA指标时，容易产生误导。

2.4.5 实验结果

首先，我们统计了3个CNN模型在干净的（没有添加扰动的）EEG数据上的基准结果，实验结果如表2.1 所示。从表2.1的结果可以看出，对于所有的数据集和CNN分类器，Within-subject实验的RCAs 和BCAs 要比与之对应的Cross-subject实验的结果都要高。造成这样的实验结果是由于不同用户的EEG信号样本之间存在个体的差异性，因此在相同用户上的测试结果会比在不同用户上的测试结果要好。

考虑到通用对抗扰动也是一种噪声，如果给原始EEG数据加入相同大小约束的随机噪声也能使得分类器的分类精度显著的下降，那么就没有加入通用对抗扰动的必要。因此，为了合理的对比UAP 的攻击性能，我们做了额外的实验：给原始的EEG 数据中加入了相同大小约束的随机噪声$0.2 \cdot \mathcal{U}(-1,1)$，评价随机噪声对模型分类精度的影响。实验结果如表2.1中噪声数据一列所示。从结果上来看，在大多数情况下随机噪声并不能降低分类器的RCAs/BCAs，表明这三种CNN分类器对于随机噪声都有着一定的鲁棒性，分类性能不太容易受随机噪声的干扰，因此我们需要特定的设计出对抗扰动，来实现攻击模型的效果。

表 2.1 三种CNN分类器在三个BCI数据集上的RCAs/BCAs结果。其中通用对抗扰动大小约束参数$\xi = 0.2$。

实验	数据集	目标模型	基准结果		白盒攻击
			干净数据	随机噪声	DF-UAP
Within -Subject	P300	EEGNet	.79/.79	.81/.79	**.17**/**.50**
		DeepCNN	.84/.80	.84/.81	**.18**/**.51**
		ShallowCNN	.80/.77	.80/.77	**.52**/**.64**
	ERN	EEGNet	.69/.63	.69/.64	**.32**/**.51**
		DeepCNN	.72/.71	.72/.71	**.41**/**.54**
		ShallowCNN	.69/.65	.70/.66	**.50**/**.52**
	MI	EEGNet	.50/.50	.47/.48	**.30**/**.29**
		DeepCNN	.55/.54	.55/.54	**.33**/**.33**
		ShallowCNN	.76/.76	.68/.68	**.36**/**.36**
Cross -Subject	P300	EEGNet	.68/.63	.69/.63	**.19**/**.51**
		DeepCNN	.69/.64	.70/.64	**.20**/**.51**
		ShallowCNN	.67/.62	.66/.62	**.27**/**.54**
	ERN	EEGNet	.67/.65	.67/.65	**.31**/**.51**
		DeepCNN	.65/.63	.64/.63	**.31**/**.50**
		ShallowCNN	.68/.64	.68/.64	**.49**/**.56**
	MI	EEGNet	.44/.44	.38/.38	**.28**/**.28**
		DeepCNN	.47/.47	.44/.44	**.34**/**.34**
		ShallowCNN	.47/.47	.43/.43	**.27**/**.27**

这个部分我们研究了通用对抗扰动的攻击性能。我们在ℓ_∞范数和大小约束$\xi = 0.2$的条件下给三个BCI数据集生成了通用对抗扰动。一个EEG数据样本在加入通用对抗扰动之前和之后的例子如图1所示。可以发现，通用对抗扰动非常小，加入了通用扰动的数据与原始数据重合在一起，非常不容易被察觉出来。

通用对抗扰动攻击之后的实验结果如表2.1所示。首先我们考虑了白盒攻击场景（攻击者拥有目标模型的所有信息，包括其模型结构和参数）。从表2.1中可以看出：

(1) 加入通用对抗扰动之后，三个CNN分类模型在三个BCI数据集上的分类精度都明显下降，表明了通用对抗扰动攻击的有效性。

(2) 通用对抗扰动攻击之后，P300和ERN数据集的BCA指标都接近0.5，同时RCA指标明显低于0.5，表明大部分的EEG数据样本都被CNN分类器分到了少数类。因为根据基于DeepFool的UAP生成方法，如果将那些原始分类结果为多数类的数据分到少数类，能够取得更优异的攻击性能（详见2.3 节）。

(3) 在MI数据集上，RCA和BCA指标都明显低于基准结果，表明通用对抗扰动

在多分类问题上也能够对三种CNN模型攻击成功。

在算法2中，通用扰动的大小约束ξ是一个非常重要的参数，我们通过实验评价了UAP在不同大小约束下的攻击性能（RCA指标），如图2-3所示。从图中可以看出，随着约束大小参数ξ的增加，RCA指标急剧下降直至收敛稳定。这是由于更大的ξ意味着更大的通用对抗扰动，所以可以取得更好的攻击性能。

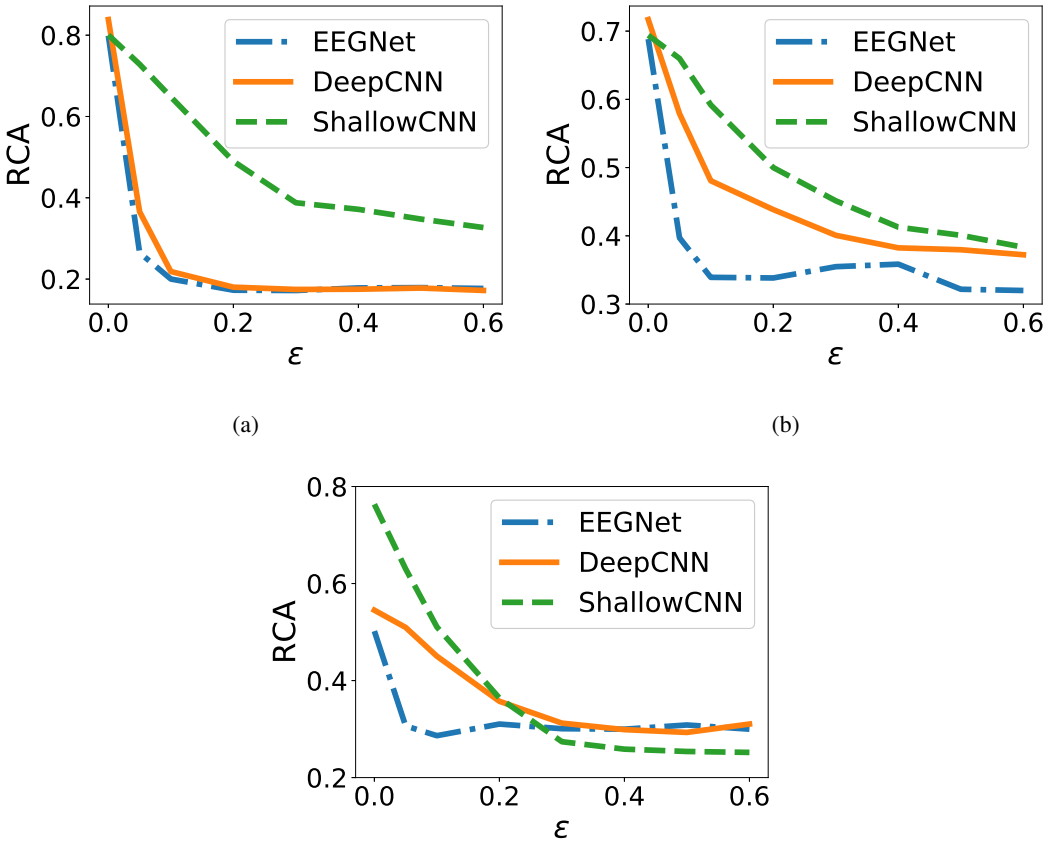


图 2-3 白盒非目标攻击场景中，添加不同大小约束ξ的TLM-UAP后，目标模型RCA结果图。(a) P300 dataset；(b) ERN dataset；(c) MI dataset。

2.4.6 通用对抗扰动的迁移性

在对抗样本的研究中，对抗扰动的迁移性是最具有危害性的一种性质，它表示我们使用某一个模型生成的对抗扰动，也能够对其它模型进行攻击，使得其它模型的性能下降。我们研究了通用对抗扰动在EEG数据上的迁移性，考虑了灰盒攻击场景：攻击者只知道训练数据，而不知道目标模型的结构和参数。我们可以在已知的训练集上训练出一个替代模型，根据替代模型来生成出通用对抗扰动，然后应用到

目标模型上。迁移性的实验结果如表2.2 所示。

表 2.2 三种CNN分类器在三个BCI数据集上的RCAs/BCAs结果。其中通用对抗扰动大小DF-UAP约束参数$\xi = 0.2$.

实验	数据集	目标模型	基线		DF-UAP的迁移性		
			干净数据	噪声数据	EEGNet	DeepCNN	ShallowCNN
Within -Subject	P300	EEGNet	.79/.79	.81/.79	.25/.54	.21/.52	.59/.71
		DeepCNN	.84/.80	.84/.81	.30/.57	.20/.56	.56/.69
		ShallowCNN	.80/.77	.80/.77	.67/.72	.65/.71	.54/.66
	ERN	EEGNet	.69/.63	.69/.64	.63/.64	.51/.59	.67/.63
		DeepCNN	.72/.71	.72/.71	.67/.69	.50/.56	.66/.67
		ShallowCNN	.69/.65	.70/.66	.67/.64	.60/.60	.58/.56
	MI	EEGNet	.50/.50	.47/.48	.40/.40	.37/.37	.40/.40
		DeepCNN	.55/.54	.55/.54	.54/.53	.43/.43	.48/.47
		ShallowCNN	.76/.76	.68/.68	.62/.61	.48/.48	.38/.37
Cross -Subject	P300	EEGNet	.68/.63	.69/.63	.24/.53	.25/.53	.30/.55
		DeepCNN	.69/.64	.70/.64	.32/.55	.21/.51	.28/.54
		ShallowCNN	.67/.62	.66/.62	.41/.58	.35/.57	.36/.57
	ERN	EEGNet	.67/.65	.67/.65	.57/.61	.53/.58	.41/.56
		DeepCNN	.65/.63	.64/.63	.54/.55	.41/.53	.44/.56
		ShallowCNN	.68/.64	.68/.64	.67/.61	.66/.59	.53/.59
	MI	EEGNet	.44/.44	.38/.38	.32/.32	.32/.32	.33/.33
		DeepCNN	.47/.47	.44/.44	.37/.37	.40/.40	.36/.36
		ShallowCNN	.47/.47	.43/.43	.30/.30	.27/.27	.36/.36

从表中可以看出，所有的RCA和BCA指标都比基准正确率要低，表明在EEG数据上，通用对抗扰动能够在不同的模型之间进行迁移，取得不同的攻击性能。

2.4.7 通用对抗扰动的特性

我们还做了额外的实验，对DF-UAP的特性进行了研究。首先，我们通过随机打乱训练集的方式，生成的四种不同的DF-UAP并进行了可视化，如图2-4所示。

从图中可以看出，这四种通用对抗扰动肉眼上看上去非常相似，但是在细节上不完全相同。另外，我们计算了这四种DF-UAP的相关系数矩阵，来分析它们之间的相似性，如图2-5所示，可以看出DF-UAP在相关系数上也还是存在明显差异。该实验结果表明，对于一个给定的EEG数据集，可以设计出的通用对抗扰动是不唯一的。同样的，我们在之后的针对UAP的防御工作中，由于UAP的多样性，如果检测机制

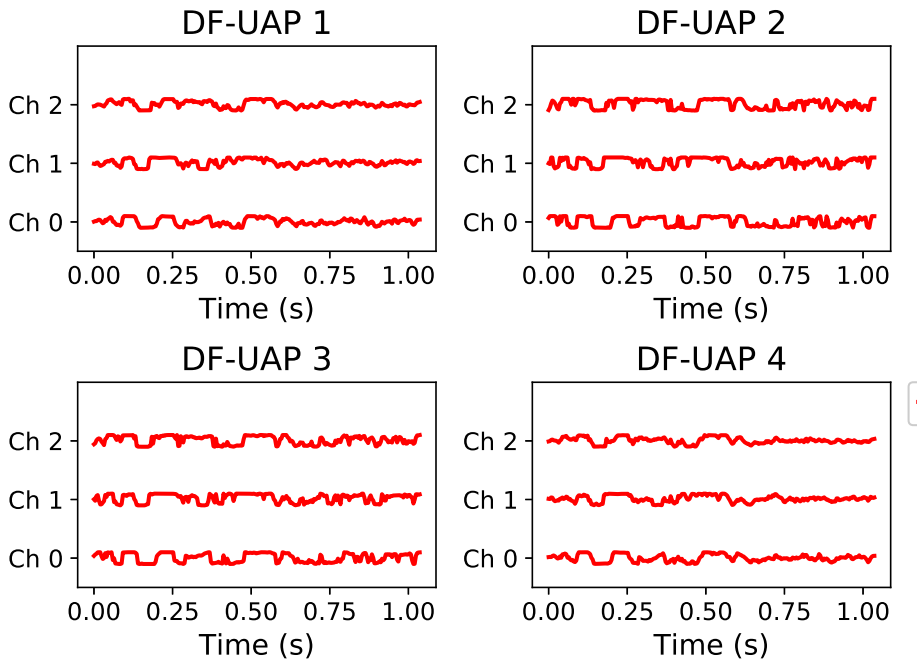


图 2-4 DF-UAP的多样性。四种不同的DF-UAP，在ERN数据集上针对EEGNet设计（随机打乱训练集四次）。

或者防御模型只能对某一种UAP 有防御能力，那么说明该防御方法还存在欠缺，仍然有需要提升的空间。

由于通用对抗扰动是基于训练数据集中生成的，研究其攻击性能与训练数据集的大小之间的关系是非常有趣的。图2-6为在Cross-subject实验下的通用对抗扰动攻击性能与不同训练集大小之间的关系。从图中我们可以看出，我们只使用50个EEG数据样本设计通用扰动时，DF-UAP的攻击成功率已经达到了40%，训练集数据样本数在200时，DF-UAP的攻击效果就非常不错了，该实验结果表明我们不需要一个大的数据集就能够生成出一个十分有效的通用对抗扰动，在文献[55]中的针对图片的UAP实验中也观察到了同样的现象。

图2-7展示了在三个EEG数据集上，加入通用对抗扰动之前和之后，被CNN分类器分到每个类别中的EEG样本数。从图中可以看出，添加DF-UAP之后，三个数据集上的预测类别比例发生了显著改变，测试样本往往被CNN模型从多数类被分到少数类。P300和ERN这两个二分类数据集上，由某一类别变成另外一个类别；MI这个四分类数据集中，测试样本大多数被分到Feet类别。该实验结果是可以理解的，我们假设少数类样本占总体样本的比例为$p\%$ $(p < 50)$，那么如果将少数类的数据样本全部分类为多数类将取得$p\%$的攻击成功率，但是将多数类的数据样本全部分为少数类将取得$(100-p)\%$ 的攻击成功率，显而易见的是，后者的攻击成功率更大。

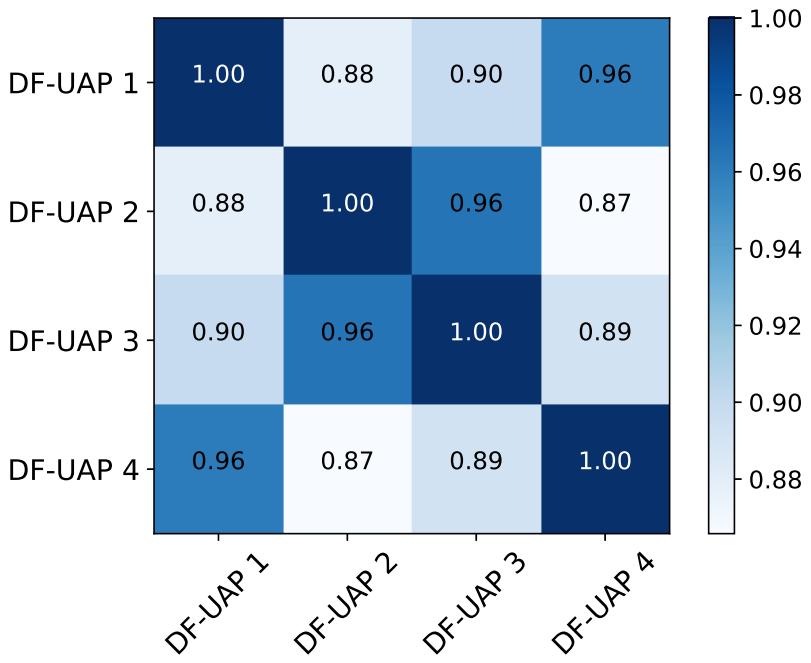


图 2-5 四种DF-UAP的相关系数矩阵。

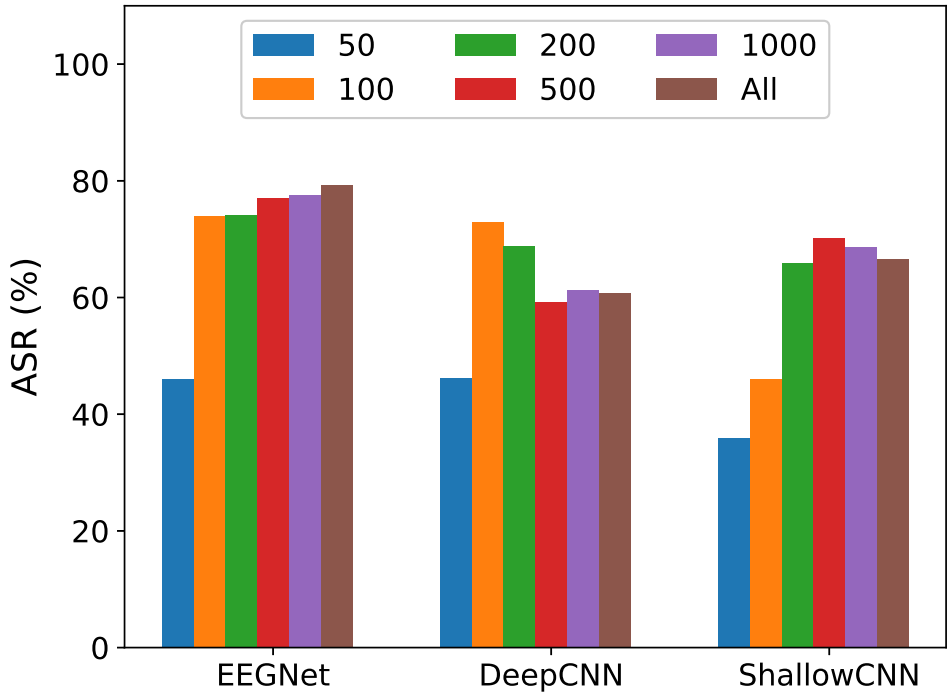
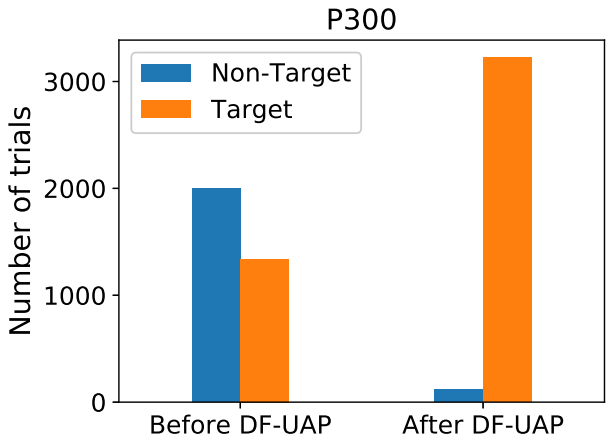
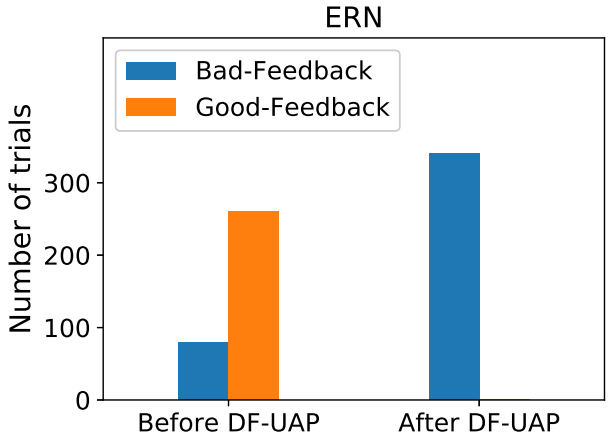


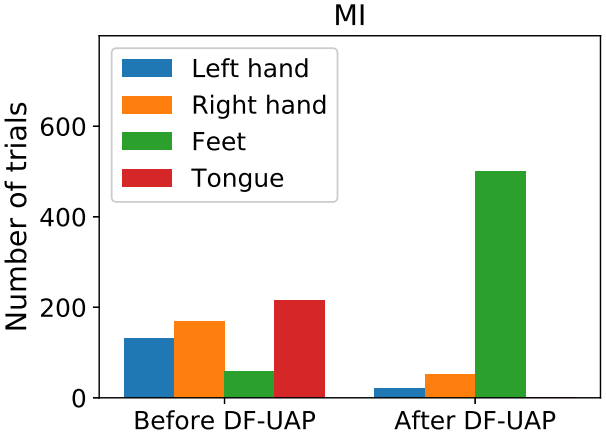
图 2-6 攻击成功率与训练集数据量大小之间的关系图。图为Cross-subject实验中DF-UAP在MI数据集上的攻击结果；”All”代表算法2在设计DF-UAP时，使用了MI数据训练集中所有4608个EEG数据样本。



(a)



(b)



(c)

图 2-7 在白盒非目标攻击场景，添加DF-UAP之前和添加之后，被分类器EEGNet分到每个类别的样本数量. a) P300数据集；b) ERN数据集；c) MI数据集。

2.5 本章小结

本章介绍了一种基于DeepFool的通用对抗扰动设计算法，并使用该算法第一次成功设计出通用对抗扰动DF-UAP攻击脑机接口系统。首先我们介绍了一种样本独立的对抗攻击算法DeepFool，然后基于DeepFool为EEG数据设计出了一种通用对抗扰动DF-UAP，将UAP的思想成功的推广应用到脑机接口系统中。实验表明我们设计出的DF-UAP能够成功攻击脑机接口系统，显著降低CNN分类器的分类性能。在白盒攻击场景中，我们设计的DF-UAP在脑机接口中两种不同的实验设置（Within-sunject和Cross-subject）上都取得了非常不错的攻击性能，使得三种常用的CNN分类器在三个BCI数据集上的分类准确率都明显降低。对于灰盒攻击场景，我们对DF-UAP在脑电数据上的迁移性进行了研究，实验表明通过其他替代模型生成的DF-UAP也能够降低目标模型的分类能力，DF-UAP在不同的攻击场景中都对脑机接口系统攻击成功。最后，我们研究了DF-UA的特性，包含：多样性、与训练数据集样本量大小的关系、以及添加DF-UAP 前后对分类器分类结果的具体影响。总之，基于DeepFool设计的通用对抗扰动在脑机接口系统中是可行的。为了进一步提高通用对抗扰动的攻击成功率和通用性，我们将在下一章介绍我们提出的基于总损失最小化的通用对抗扰动生成算法（TLM-UAP），让通用对抗扰动的设计方式更加的简洁，可扩展性更强。

3 基于总损失最小化的通用对抗扰动

上一章介绍的DF-UAP的实验结果表明，脑机接口系统的CNN分类器非常容易受到UAP的攻击，使得模型的性能大幅下降。然而，DF-UAP设计算法只能应用在非目标攻击中，对于目标攻击场景，还没有相关的UAP攻击研究。如果UAP能够使模型输出任意指定的类别，那么其危害性更大。为了进一步提高DF-UAP的攻击性能，来设计出攻击范围更广，通用性更好的扰动，我们提出了基于总损失最小化（Total loss minimize，TLM）的通用对抗扰动设计算法，不仅适用于非目标攻击场景，也可以在目标攻击场景应用。

3.1 总损失最小化

对于一般的CNN模型，目的是正确的识别数据，正常训练过程是使用梯度下降法来优化模型的损失函数，使损失最小化。一般使用交叉熵作为损失函数，如下式所示：

$$l(\boldsymbol{x}, y) = -\log(p_{y_t}(\boldsymbol{x})), \tag{3.1}$$

其中，y_t是数据样本$\boldsymbol{x}$对应的真实标签。p_{y_t}表示模型预测样本$\boldsymbol{x}$的属于类别y_t的概率。

对抗攻击的目的是让目标模型对数据错误分类，使得原始的预测结果发生改变。它与正常训练是一个相反的过程，相当于加入精心设计的对抗扰动后，模型在对抗样本上的损失最大化。然而，最大化问题是能够通过改变待优化目标的符号把它变成对偶问题（最小化问题）来求解。文献[53]提出了一种期望转换（Expectation over transformation，EOT）的方法，考虑了视角转换、摄像机噪声和其他自然转换，作者使用优化的方式设计对抗扰动，然后应用在3D打印出来的物体，使得对抗攻击在现实生活中也能成功应用。那么，我们能否也通过优化的方式来得到可以攻击分类器的UAP呢?

与基于DeepFool的通用对抗扰动不同，TLM将UAP视为可优化参数，直接通过优化一个目标函数来设计出UAP。具体来讲，对一个batch的数据进行梯度下降，使得目标函数总损失最小：在白盒攻击场景中，由于受害模型的参数是已知并且固定的，因此我们能够将通用扰动视为一个可优化的变量，在整个已知的训练集上应用梯度下降来最小化目标损失函数，以此来设计出我们期望的UAP。

更具体的讲，我们通过求解如下的优化问题来设计UAP：

$$\min_{\boldsymbol{v}} E_{\boldsymbol{x}\sim\boldsymbol{D}} l(\boldsymbol{x}+\boldsymbol{v}, y) + \alpha \cdot C(\boldsymbol{x}, \boldsymbol{v}), \quad \text{s.t.} \|\boldsymbol{v}\|_p \leq \xi \tag{3.2}$$

其中，$l(\boldsymbol{x}+\boldsymbol{v},y)$是一个损失函数，$y$是数据样本$\boldsymbol{x}$的（真实或预测）标签，$C(\boldsymbol{x},\boldsymbol{v})$为在通用对抗扰动$\boldsymbol{v}$上的约束，$\alpha$为正则化系数。我们提出的TLM方法是非常灵活的：攻击者能够根据不同的任务需求，选择不同的优化器、损失函数以及约束条件。

值得一提的是，我们提出的方法可以通过简单的修改损失函数l，就能够同时应用到非目标攻击以及目标攻击中。

对于非目标攻击，攻击者的目的只想让模型对数据样本的输出结果改变，不用指定被错误分到哪一类。本文中CNN使用的损失函数都是交叉熵，那么，损失函数l能够被定义为：

$$l(\boldsymbol{x},y)=\log(p_y(\boldsymbol{x})), \tag{3.3}$$

其中，$p_y(\boldsymbol{x})$是分类器对于数据样本$\boldsymbol{x}$与真实标签y相关联的预测概率。如果在真实标签不可获取的场景下，我们可以使用$\arg\max_j p_j(\boldsymbol{x})$得到数据样本$\boldsymbol{x}$的预测标签来代替公式(3.1)中的真实标签$y$。对比公式（3.1）和公式（3.3）可以发现，交叉熵函数去掉了负号，相当于把CNN训练过程中损失最小化问题转变成了损失最大化。

对于目标攻击，我们强制分类器模型将一个添加对抗扰动之后的对抗样本分类到一个特定的类中，所以损失函数能够被定义为：

$$l(\boldsymbol{x},y)=-\log(p_y(\boldsymbol{x})), \tag{3.4}$$

其中，y是攻击者期望的目标标签（通常上与真实标签不同）。

值得一提的是，公式(3.3)和公式(3.4)的唯一区别为公式(3.4)中多添加了一个负号，这是因为在非目标攻击中，我们想最小化分类器对数据样本分类正确（预测为真实标签）的预测概率，在目标攻击中，我们期望最大化分类器对数据样本分类为目标标签的预测概率。

我们提出的TLM方法除了能够将通用对抗扰动从非目标攻击扩展到目标攻击以外，通过不同的约束条件$C(\boldsymbol{x},\boldsymbol{v})$，可以对通用对抗扰动$\boldsymbol{v}$进行不同需求的调节。例如，在大多数情况下，我们可以简单的设置$C(\boldsymbol{x},\boldsymbol{v})$为在通用对抗扰动$\boldsymbol{v}$上的L1或L2正则化。然而，约束条件$C(\boldsymbol{x},\boldsymbol{v})$也可以为更复杂的函数，例如一种用于检测输入是否为对抗样本的度量函数；当有人提出了一种新的度量函数来检测对抗样本时，我们提出的方法可以用来测试它的可靠性：我们将C设置为新的度量函数，来检查我们的方法是否仍然可以找到攻击性能不错的通用对抗扰动。如果仍然可以找到通用扰动，说明该度量函数依然存在上升的空间。

考虑到度量函数的多样性，本文只考虑C为L1正则化和L2正则化。其他度量函数和对抗样本防御策略将在我们未来的工作中进行研究。我们提出的基于TLM的通用对抗扰动生成算法的伪代码如算法3所示。

Algorithm 3: 基于TLM方法的通用对抗扰动设计算法伪代码。

Input: $\boldsymbol{X}_{train} = \{\boldsymbol{x}_{train,i}\}_{i=1}^{n}$, n个EEG训练数据;
$\boldsymbol{X}_{val} = \{\boldsymbol{x}_{val,i}\}_{i=1}^{m}$, m个EEG验证数据;
k, 分类器;
ξ, 通用对抗扰动ℓ_p范数最大值;
α, 正则化项系数;
δ, 期望攻击成功率ASR;
M, 最大迭代次数;
Output: $\boldsymbol{v}_{best}$, 通用对抗扰动TLM-UAP.

```
v = 0;
r = 0;
for m = 1, ..., M do
    for 每一个小批量数据D ∈ X_train do
        对于数据D，使用一个优化器来更新公式(3.1)中的v;
        使用公式(2.14)来约束v: v ← P_{p,ξ}(v);
    end
    X_{val,v} = {x_{val,i} + v}_{i=1}^{n};
    if ASR(X_{val,v}, X_val) > r then
        r = ASR(X_{val,v}, X_val);
        v_best = v;
    end
    if R > δ then
        Break;
    end
end
Return v_best.
```

3.2 全通道相同的通用对抗扰动

由于我们提出的TLM方法是通过优化的方式来设计通用对抗扰动，因此我们有一个更大胆的想法：能否设计出更通用的对抗扰动? 给一个数据样本中每个通道都添加相同的扰动。首先，我们随机初始化一个通道上的通用扰动模板$\boldsymbol{v}_c \in \mathbb{R}^{1\times T}$，然后将该模板加入到每个数据样本中的每个通道上，新的对抗样本为$\tilde{\boldsymbol{x}}_i = \{\boldsymbol{x}_i^1 + \boldsymbol{v}_c, \ldots, \boldsymbol{x}_i^C + \boldsymbol{v}_c\}$，最后基于公式(3.3) 或者公式(3.4)，利用梯度下降的优化器得到更新后的单通道通用扰动的模板$\boldsymbol{v}_c$。

3.3 实验部分

3.3.1 实验设置

为了便于我们提出的TLM方法设计的通用对抗扰动TLM-UAP的攻击效果与本文第二章的DF-UAP进行比较，我们采用了和2.4.3完全相同的实验设置和性能指标，分别对Within-subject和Cross-subject两个不同的脑机接口应用场景进行了研究。算法2 生成DF-UAP和算法3生成TLM-UAP 的参数如表3.1所示。值得一提的是，实验中TLM-UAP是在L2正则化的约束条件下生成的，因此不用担心通用扰动会随着训练迭代次数的增加而不断增加，同时，我们将攻击成功率的阈值δ设置为1.0且用到早停（patience=10）来作为迭代停止的条件。在整个实验过程，我们没有使用训练集的真实标签，而是使用分类器对于样本的预测标签，在实际的应用场景（未知真实标签）中结果会更加真实。

表 3.1 设计DF-UAP和TLM-UAP的参数对比表. 两种通用对抗扰动都使用无穷范式$\|\boldsymbol{v}\|_\infty$.

	ξ	δ	M	α	约束函数
DF-UAP	0.2	0.8	10	-	-
TLM-UAP	0.2	1.0	500	100	L2

3.3.2 实验结果

考虑到TLM-UAP能够从非目标攻击，目标攻击，全通道相同这三个层面对分类器模型进行对抗攻击，该部分实验结果从上述三个层面进行了研究与讨论。

3.3.2.1 非目标攻击

在白盒攻击场景中，加入TLM-UAP之后，三种分类器在三个BCI数据集上的分类准确率RCA/BCA如表3.2所示，为了方便比较，我们同时列出了表2.1中DF-UAP的白盒攻击结果。

从表3.2中的结果可以看出：

(1) 加入TLM-UAP之后，三个CNN分类模型在三个BCI数据集上的分类精度都明显下降，表明了TLM-UAP攻击的有效性。

(2) 在大多数情况下，TLM-UAP的攻击效果要优于DF-UAP。

(3) TLM-UAP和DF-UAP攻击之后，P300和ERN数据集上的BCA结果接近0.5，表明为了在整体数据集上取得最好的攻击性能，测试集中EEG数据样本基本上都被分到了少数类。

由于TLM-UAP是采用优化的方法对一个批（batch）数据进行求解，而DF-UAP是在每个样本的基础上进行一次迭代，无法并行，因此，生成一个有攻击能力

表 3.2 白盒非攻击场景中，TLM-UAP在三个EEG数据集上攻击三种不同CNN分类器的RCAs/BCAs结果（随机噪声和TLM-UAP大小约束为$\xi = 0.2$）。

实验	数据集	目标模型	基准结果		白盒攻击	
			干净数据	随机噪声	DF-UAP	TLM-UAP
Within -Subject	P300	EEGNet	.79/.79	.81/.79	**.17/.50**	**.17/.50**
		DeepCNN	.84/.80	.84/.81	.18/.51	**.17/.50**
		ShallowCNN	.80/.77	.80/.77	.52/.64	**.34/.56**
	ERN	EEGNet	.69/.63	.69/.64	.32/.51	**.31/.50**
		DeepCNN	.72/.71	.72/.71	**.41/.54**	.50/.59
		ShallowCNN	.69/.65	.70/.66	.50/.52	**.49/.52**
	MI	EEGNet	.50/.50	.47/.48	.30/.29	**.24/.25**
		DeepCNN	.55/.54	.55/.54	.33/.33	**.26/.29**
		ShallowCNN	.76/.76	.68/.68	.36/.36	**.28/.28**
Cross -Subject	P300	EEGNet	.68/.63	.69/.63	.19/.51	**.17/.50**
		DeepCNN	.69/.64	.70/.64	.20/.51	**.18/.50**
		ShallowCNN	.67/.62	.66/.62	.27/.54	**.19/.50**
	ERN	EEGNet	.67/.65	.67/.65	.31/.51	**.29/.50**
		DeepCNN	.65/.63	.64/.63	**.31/.50**	.33/.50
		ShallowCNN	.68/.64	.68/.64	.49/.56	**.29/.50**
	MI	EEGNet	.44/.44	.38/.38	.28/.28	**.25/.25**
		DeepCNN	.47/.47	.44/.44	.34/.34	**.25/.25**
		ShallowCNN	.47/.47	.43/.43	.27/.27	**.25/.25**

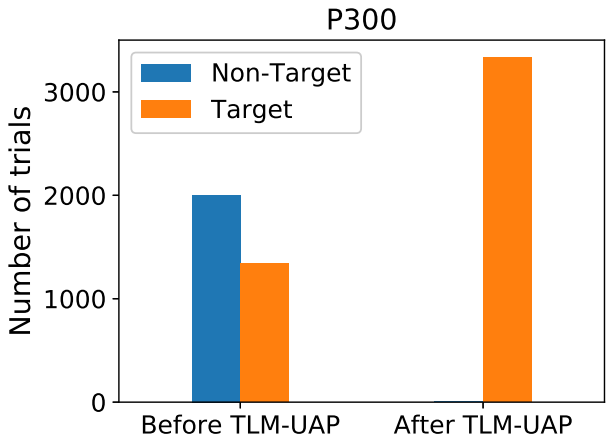
的通用扰动的速度要比DF-UAP快许多。

图3-1为在加入TLM-UAP之前和之后，分类器EEGNet在三个BCI数据集上每个类别的分类结果。从图中可以看出，一般加入TLM-UAP之后，原始的EEG数据样本从原来的属于大多数类被重新分到了少数类。与DF-UAP的结果图2-7相比，TLM-UAP攻击后基本上将所有类别都分到了某一类。

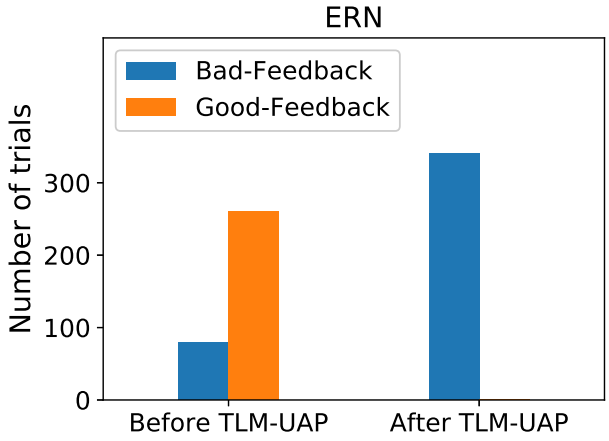
图3-2展示了一个EEG数据样本在添加TLM-UAP之前与之后的的具体形式。从图中可以看出，TLM-UAP是非常小的，加入了TLM-UAP的对抗样本与原始数据近乎重合在一起，非常不容易被察觉出来。

TLM-UAP在灰盒攻击场景中的迁移性实验结果如表3.3所示。从表中可以观察到:

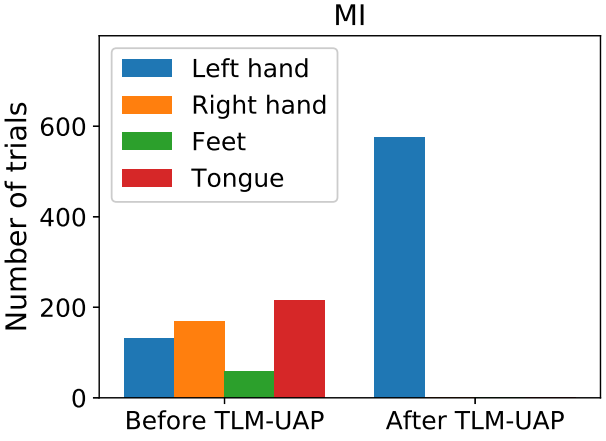
(1) 三种CNN分类器在三个数据集上的分类精度都得到了下降，表明TLM-UAP可以在不同模型之间进行迁移，通过替代模型生成TLM-UAP在目标模型上仍然



(a)



(b)



(c)

图 3-1 在白盒非目标攻击场景，添加TLM-UAP之前和添加之后，被分类器EEGNet分到每个类别的样本数量。a) P300数据集；b) ERN数据集；c) MI数据集。

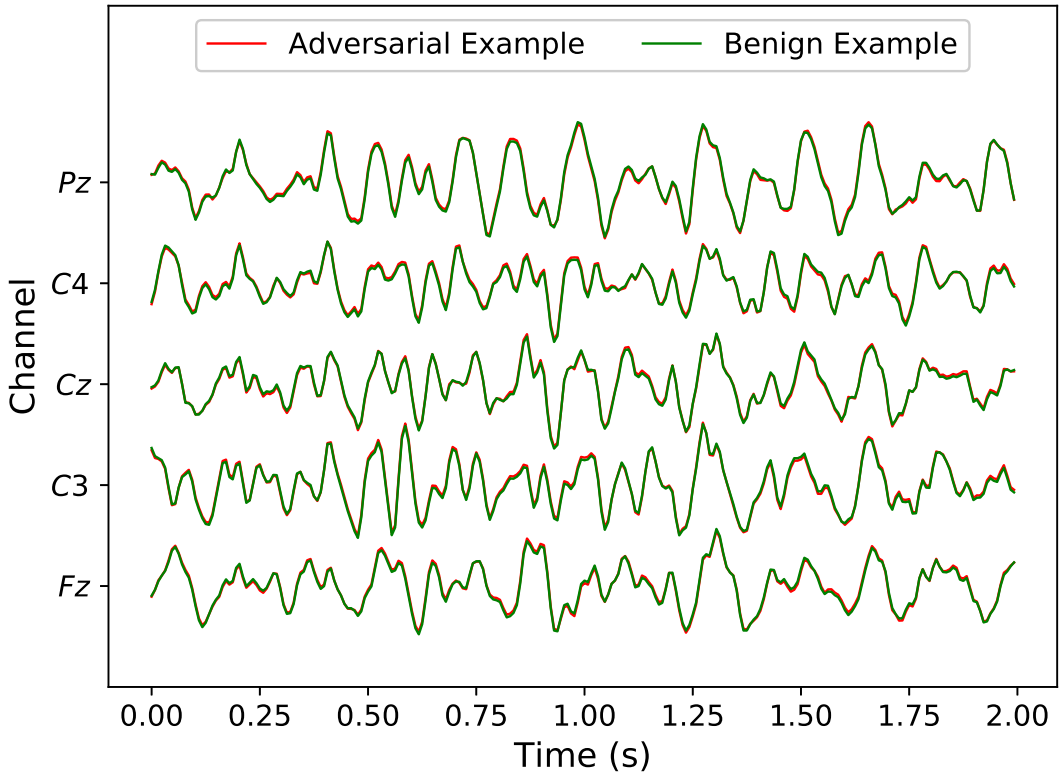


图 3-2 在白盒非目标攻击场景，给MI数据集中一个EEG数据样本添加TLM-UAP之前和添加之后的对比示例（TLM-UAP扰动大小约束为$\xi = 0.2$）。

有一定的攻击效果。

(2) 在大多数情况下，TLM-UAP对比DF-UAP会使分类器的RCA和BCA指标的下降的更多，表明我们提出的方法生成的TLM-UAP的迁移性在大多数情况下要优于DF-UAP。

除此以外，我们还做了额外的实验对TLM-UAP的部分特性做了进一步的研究，包含扰动与原始信号的信噪比以及光谱图。

首先，我们计算了添加扰动之后的EEG数据样本的信噪比（Signal-to-Perturbation Ratio，SPR），扰动包含随机噪声以及白盒攻击下的DF-UAP 和TLM-UAP。我们将原始的EEG数据样本看做干净信号，计算了在cross-subject实验中的SPR，实验结果如表3.4所示。三种CNN分类器在三个BCI数据集上的9组实验结果中，有8组的TLM-UAP的SPR比DF-UAP的SPR要高，表明通过TLM方法设计的UAP对比基于DeepFool设计的UAP有着更小的扰动大小，会使得TLM-UAP 更不容易被人所察觉和识别出来。这是由于除了给扰动添加条件$\|\boldsymbol{v}\|_p \leq \xi$ 以外，TLM-UAP也受到公式(3.1)中约束函数$C(\boldsymbol{x}, \boldsymbol{v})$ 的约束。

之后为了分析通用对抗扰动的时频特性，我们对比了在白盒攻击场景下DF-

表 3.3 灰盒非攻击场景中，TLM-UAP的迁移性实验结果（扰动大小约束$\xi = 0.2$）。

实验	数据集	目标模型	灰盒攻击					
			替代模型（DF-UAP）			替代模型（TLM-UAP）		
			EEGNet	DeepCNN	ShallowCNN	EEGNet	DeepCNN	ShallowCNN
Within -Subject	P300	EEGNet	.22/.52	.21/.52	.59/.71	**.18/.51**	**.18/.51**	**.49/.65**
		DeepCNN	.30/.57	.20/.52	.56/.69	**.21/.52**	**.18/.51**	**.49/.64**
		ShallowCNN	.67/.72	.65/.71	.60/.68	**.55/.67**	**.48/.64**	**.44/.60**
	ERN	EEGNet	.62/.68	**.51/.59**	.67/.63	**.57/.65**	.56/.58	**.64/.63**
		DeepCNN	.67/.69	**.53/.58**	.66/.67	**.66/.69**	.57/.59	**.64/.66**
		ShallowCNN	**.67/.64**	**.60/.60**	**.57/.60**	.68/.65	.63/.61	.60/.62
	MI	EEGNet	**.38/.37**	.37/.37	.40/.40	.42/.42	**.30/.30**	**.36/.36**
		DeepCNN	**.54/.53**	.46/.46	.48/.47	.54/.54	**.33/.33**	**.34/.34**
		ShallowCNN	**.62/.61**	.48/.48	.38/.38	.72/.72	**.42/.42**	**.28/.28**
Cross -Subject	P300	EEGNet	.24/.53	.25/.53	.30/.55	**.17/.50**	**.17/.50**	**.20/.51**
		DeepCNN	.32/.55	.22/.52	.28/.54	**.18/.50**	**.18/.50**	**.20/.52**
		ShallowCNN	.41/.58	.35/.57	.32/.55	**.28/.54**	**.24/.52**	**.21/.51**
	ERN	EEGNet	.53/.58	.53/.58	.41/.56	**.32/.51**	**.34/.51**	**.35/.53**
		DeepCNN	.54/.55	.48/.54	.44/.56	**.30/.50**	**.34/.50**	**.34/.60**
		ShallowCNN	.67/.61	.66/.59	.53/.58	**.53/.59**	**.53/.61**	**.30/.51**
	MI	EEGNet	**.35/.35**	.32/.32	.33/.33	.36/.36	**.31/.31**	**.26/.26**
		DeepCNN	**.37/.37**	.35/.35	.36/.36	.42/.42	**.31/.31**	**.29/.29**
		ShallowCNN	**.30/.30**	**.27/.27**	.37/.37	.44/.44	.31/.31	**.29/.29**

UAP和TLM-UAP在三种CNN分类器上的光谱图。实验结果如图3-3所示。从图中可以看出，DF-UAP和TLM-UAP存在部分相似的光谱表现形式：这两种通用扰动在EEGNet和DeepCNN的光谱图中的能量主要集中在低频区域，在ShallowCNN上则表现的更为分散。当然，更具体的来讲，TLM-UAP和DF-UAP的光谱图之间也还是存在着明显的不同。对于EEGNet，DF-UAP的能量主要集中在$[0.1, 0.9]$s和$[0, 7]$Hz，TLM-UAP则主要集中在$[0.1, 0.8]$s和$[3, 8]$Hz；对于DeepCNN，TLM-UAP影响更长的信号周期，在$[0.4, 0.8]$s之间。对于ShallowCNN，TLM-UAP主要在高频区扰动，其均匀性分布不如DF-UAP。

上述光谱图的分析结果也可以解释表3.3中Cross-subject实验中TLM-UAP在P300数据集上的实验结果：从光谱图上来看，在EEGNet和DeepCNN分类器上设计的TLM-UAP比ShaalowCNN更具有相似性，因此在灰盒攻击场景中，在EEGNet（DeepCNN）上设计的TLM-UAP迁移到DeepCNN（EEGNet）上，拥有更相似的攻击性能，导致了它们的RCA和BCA指标的实验结果更接近。

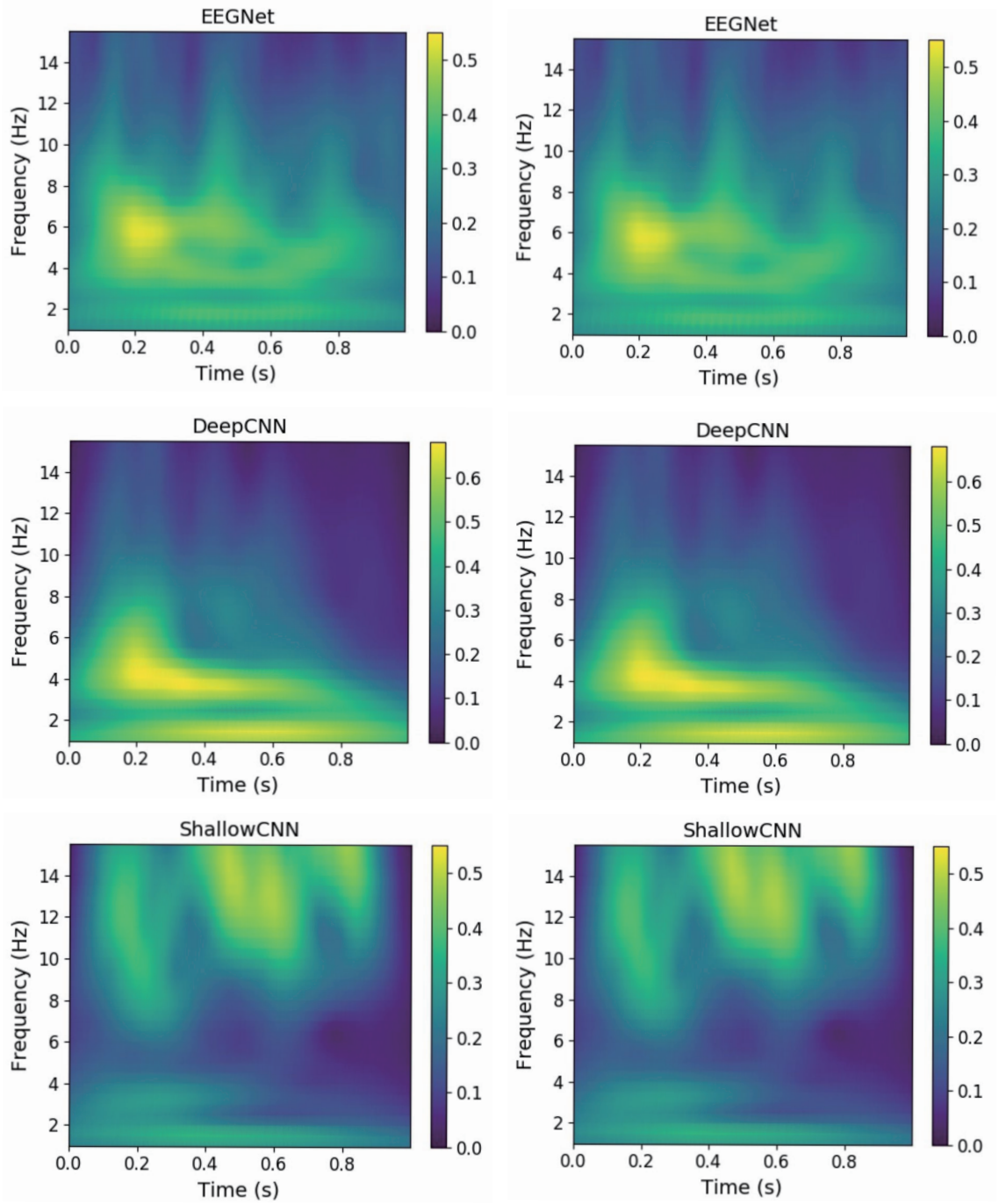


图 3-3 在白盒非目标攻击场景，DF-UAP和TLM-UAP在P300数据集上的光谱图（Within-subject实验）。图为通道C_z上的结果。(a) DF-UAP；(b) TLM-UAP。

表 3.4 在白盒非目标攻击场景，EEG数据样本添加DF-UAP和TLM-UAP扰动之后的信噪比（扰动大小约束为$\xi = 0.2$）。

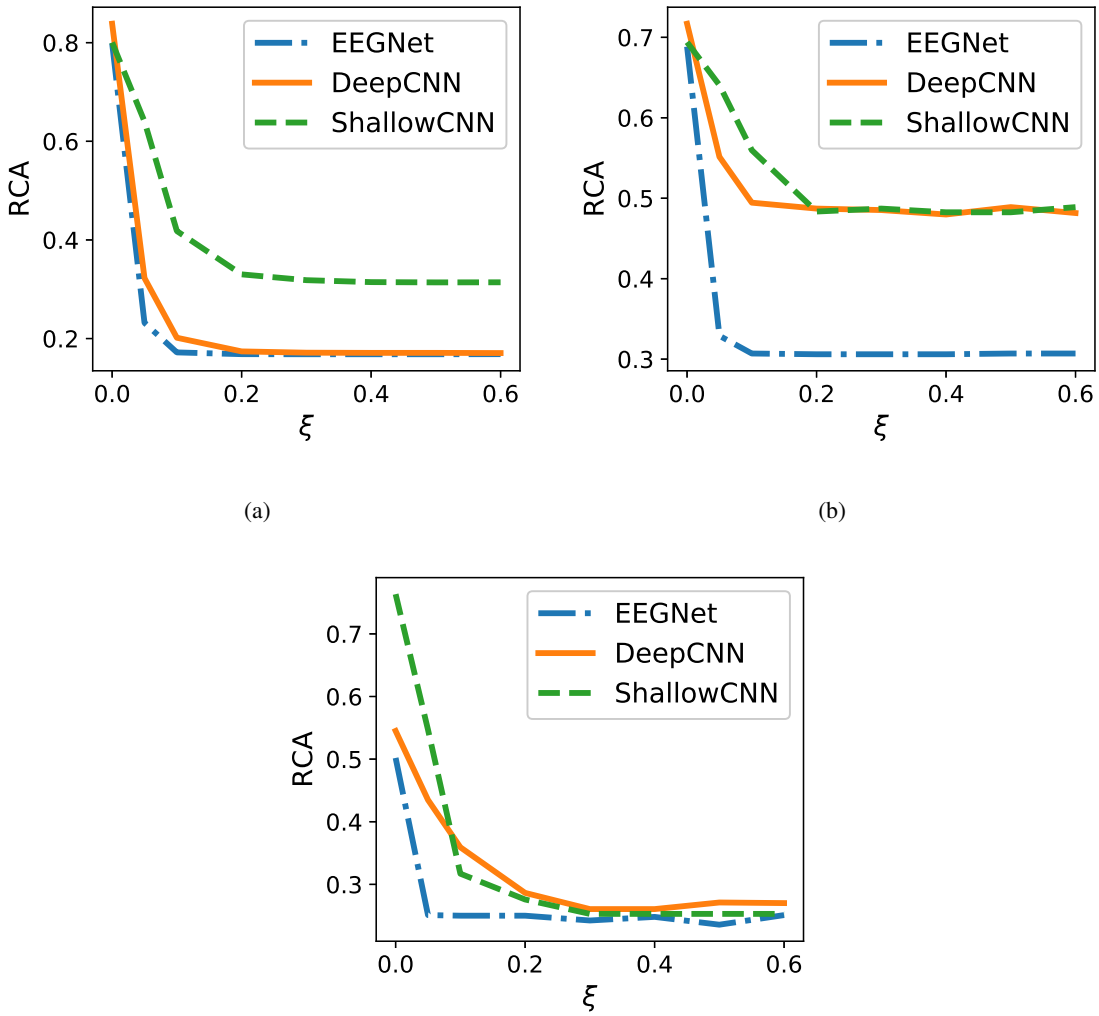
	数据集	EEGNet	DeepCNN	ShallowCNN
DF-UAP	P300	16.99	17.00	17.85
	ERN	16.22	16.73	**17.73**
	MI	21.71	13.08	14.57
TLM-UAP	P300	**21.17**	**19.92**	**20.58**
	ERN	**21.03**	**21.67**	17.72
	MI	**23.48**	**17.85**	**17.80**

我们还对TLM-UAP对于算法中不同的超参数设置的敏感性进行了研究。首先是TLM-UAP扰动大小的约束参数ξ：在算法3中，ξ是一个非常重要的参数，直接设定了扰动的大小上限。我们评价了TLM-UAP在不同的ξ下的攻击性能，实验结果如图3-4所示。随着ξ的增加，RCA快速下降直到收敛（一般在$\xi = 0.2$时收敛），表明即使是一个小的通用对抗扰动，也有着足够强的攻击目标模型的能力。该实验结果与DF-UAP的图2-3的结果吻合。

其次是用于设计TLM-UAP的训练集的数据量大小对TLM-UAP攻击性能的影响。由于我们的TLM算法在设计UAP时也依赖于训练数据，在公式（3.1）里，受到数据分布$\boldsymbol{D}$的影响。图3-5展示了在白盒攻击场景下TLM-UAP在MI数据集上不同训练集的数据量大小的非目标攻击性能（Cross-subject实验）。从图中可以看出，TLM-UAP在仅使用50个训练数据时，就已经达到超过70%的攻击成功率，并且在增加训练数据量后，攻击效果也很稳定。与DF-UAP的结果图2-6相比，TLM-UAP在数据量少的条件下，攻击效果依然要更好。该实验结果表明在我们的方法设计TLM-UAP时不需要很大的数据量就已经可以具有非常强的攻击性。

最后，我们比较了在公式（3.1）中不同约束函数C下设计TLM-UAP的不同表现形式：约束函数C包含无约束（No），L1正则化（$\alpha = 10/10/5$对应于EEGNet/DeepCNN/ShallowCNN），L2正则化($\alpha = 100$)。白盒非目标攻击场景中，不同约束下TLM-UAP在三个数据集上的SPR如表3.5所示，我们同时列出了平均RCA的结果。从表中可以看出，在不同约束条件下设计的TLM-UAP，信噪比SPR会有明显差异，除了有着相似的攻击性能以外，有约束函数条件下设计出的TLM-UAP 的SPR 要高于不带约束的SPR（在‘L1’和‘L2’行的SPR 结果要高于‘No’行的结果）。因此，该实验结果表明我们的TLM方法中加入约束函数能够调节TLM-UAP的形式，可以在保证攻击效果的同时，降低UAP 扰动的大小。

图3-6表明，加入不同的约束函数C后，设计出的TLM-UAP的波形发生了显著变化。在L1正则化约束条件下生成的TLM-UAP使得波形更加稀疏，L2正则化下的TLM-UAP则减少了扰动的大小。除此以外，我们同样研究了TLM-UAP在满足其



(a) (b)

(c)

图 3-4 白盒非目标攻击场景中，添加不同大小约束ξ的TLM-UAP后，三种目标模型RCA结果图。(a) P300 dataset；(b) ERN dataset；(c) MI dataset。

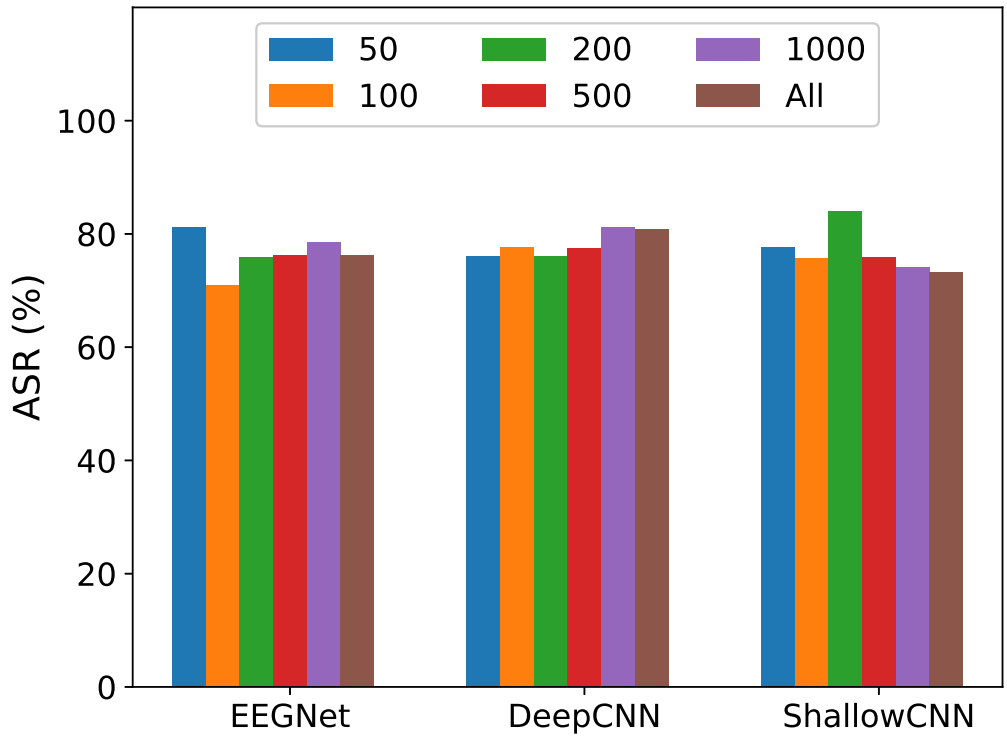


图 3-5 白盒非目标攻击场景中，不同训练集大小下的攻击成功率ASR结果图（Cross-subject实验，MI数据集）。其中'All'代表算法3使用了MI数据的训练集中所有4,608个训练样本。

表 3.5 白盒非目标攻击场景中，在不同约束函数下设计出的TLM-UAP在三个数据集上的SPR (dB) 与平均RCA (%)结果。

数据集	约束	平均RCA	信噪比（TLM-UAP）		
			EEGNet	DeepCNN	ShallowCNN
P300	No	17.18	14.89	14.71	14.45
	L1	17.36	18.39	17.82	17.16
	L2	17.85	21.17	19.92	20.58
ERN	No	30.96	19.91	20.70	17.02
	L1	29.24	21.45	22.05	17.11
	L2	30.66	21.03	21.67	17.72
MI	No	25.05	22.88	15.46	16.11
	L1	25.08	23.35	53.76	16.88
	L2	25.06	23.48	17.85	17.80

他约束函数C下的表现，例如将TLM-UAP添加到特定的EEG通道上，或者是根据检测对抗样本的度量函数，这些实验结果将在我们未来的研究中展示。

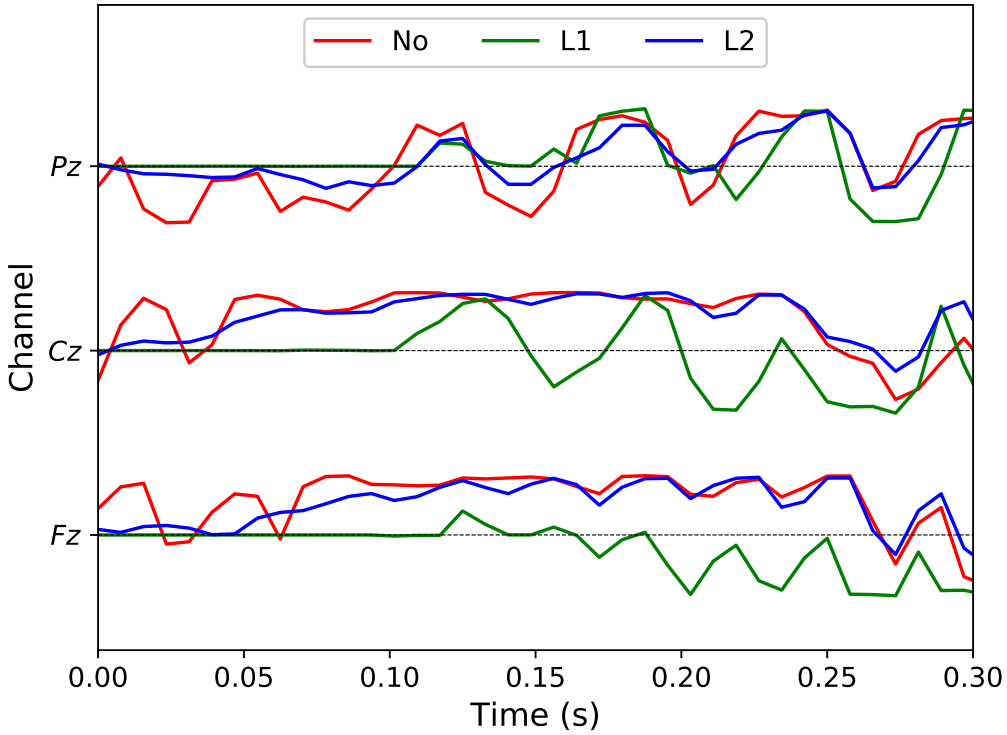


图 3-6 白盒非目标攻击场景，在MI数据集上不同约束函数下设计出的TLM-UAP. 图中使用了通道P_z，C_z和F_z的结果.

3.3.2.2 目标攻击

我们提出的TLM方法同样适用于目标攻击，只需要改动公式(3.1)中的损失函数l的符号。为了评价TLM-UAP在目标攻击场景的表现，我们做了Cross-subject实验，对三个EEG数据集上进行了白盒攻击，并计算出了目标率（被分类到目标类别的样本数与总样本数的比率）。并与添加随机均匀分布噪声的试验结果进行了对比，实验结果如表3.6所示。

从表3.6中可以观察到，TLM-UAP在所有数据集以及三个目标模型的所有类别的目标攻击中取得了近乎100%的目标率，表明我们提出的TLM方法能够轻易的篡改脑机接口系统的分类结果，使其输出攻击者期望的结果，这可能会比非目标攻击造成更严重的危害。例如，在一个脑机接口驱动的轮椅中，TLM-UAP进行目标攻击后，会迫使系统所有的指令都变成一个特定的指令（例如，向前进），这样的话很容易让用户陷入到危险的环境当中。

据我们所知，还没有人进行通用对抗扰动的目标攻击研究，我们提出的方

表 3.6 白盒目标攻击场景，TLM-UAP在三个EEG数据集上攻击三种不同CNN分类器的目标率结果（Cross-subject 实验，随机噪声和TLM-UAP大小约束为$\xi = 0.2$）。

数据集	目标模型	目标类别	基准		TLM-UAP
			干净数据	噪声数据	目标攻击
P300	EEGNet	Non-target	.6627	.6463	.9629
		Target	.3373	.3510	.9572
	DeepCNN	Non-target	.6755	.6637	.9416
		Target	.3245	.3116	.9373
	ShallowCNN	Non-target	.6505	.6597	.8904
		Target	.3495	.3499	.8306
ERN	EEGNet	Bad	.3537	.3741	.9980
		Good	.6463	.6300	.9971
	DeepCNN	Bad	.3770	.3309	.9912
		Good	.6230	.6739	.9976
	ShallowCNN	Bad	.3033	.2910	.9741
		Good	.6967	.7160	.9888
MI	EEGNet	Left	.3152	.1350	.9821
		Right	.2830	.2056	.9850
		Feet	.1545	.1954	.9994
		Tongue	.2473	.5380	1.000
	DeepCNN	Left	.2535	.1765	.8839
		Right	.3491	.2207	.9238
		Feet	.2282	.3155	.9659
		Tongue	.1692	.2544	.9938
	ShallowCNN	Left	.2872	.1952	.9151
		Right	.2537	.1746	.9443
		Feet	.2647	.2838	.9819
		Tongue	.2124	.3673	.9983

法TLM第一次将通用对抗扰动从非目标攻击扩展到了目标攻击。

3.3.2.3 全通道相同的通用对抗扰动

除了非目标攻击和目标攻击以外，我们还做了一组额外的实验，来研究全通道相同的通用对抗扰动的可行性。通过给每个数据样本的每个通道固定相同的通用扰动模板，我们采用与TLM-UAP相同的实验设置来设计出通道型通用对抗扰动，称为Channel TLM-UAP。该扰动在白盒攻击场景中的非目标攻击实验结果如表3.7 所示。

与表3.2中基准结果相比，加入Channel TLM-UAP之后，大多数情况下分类器的分类精度都得到了一定程度上的下降；但是Channel TLM-UAP的分类器RCA/BCA要比DF-UAP和TLM-UAP 的实验结果要高5%-40%，表明在提高了扰动的通用性之后，一定程度上牺牲了部分攻击性能，很难找到合适的全通道相同的通用对抗扰动使得在全部数据集上都取得不错的攻击效果。不过，虽然Channel TLM-UAP的攻击效果并没有很强，但是仍然表明在这种通用性需求更强的应用场景中，对抗攻击还是存在可行性的，需要综合考虑攻击性能和通用性。

表 3.7 白盒非目标攻击场景，Channel TLM-UAP在三个EEG数据集上攻击三种不同CNN分类器的RCAs/BCAs结果（Channel TLM-UAP大小约束为$\xi = 0.2$）。

实验	数据集	目标模型	Channel TLM-UAP
Within -Subject	P300	EEGNet	.47/.66
		DeepCNN	.60/.72
		ShallowCNN	.64/.72
	ERN	EEGNet	.48/.58
		DeepCNN	.53/.59
		ShallowCNN	.56/.61
	MI	EEGNet	.36/.37
		DeepCNN	.52/.51
		ShallowCNN	.74/.74
Cross -Subject	P300	EEGNet	.36/.57
		DeepCNN	.40/.58
		ShallowCNN	.49/.61
	ERN	EEGNet	.53/.62
		DeepCNN	.46/.56
		ShallowCNN	.57/.61
	MI	EEGNet	.33/.33
		DeepCNN	.33/.33
		ShallowCNN	.38/.38

3.4 本章小结

本章介绍了我们提出的TLM方法来设计通用对抗扰动。首先我们提出了一种基于优化的方法来更快的设计出一种可以降低目标模型分类能力的通用对抗扰动TLM-UAP，然后将TLM-UAP从非目标攻击扩展到目标攻击和全通道相同的通用对抗扰动。实验表明，相比于DF-UAP，我们提出的方法设计出的TLM-UAP生成速度更快，有更强的攻击性能以及更良好的扩展性。首先，对于非目标攻击，TLM-UAP能够在白盒攻击场景下有效的降低三种CNN分类器在三种EEG数据集上的分类精度，并且在大多数情况下攻击性能要优于DF-UAP，同时，TLM-UAP在灰盒攻击场景下也有更强的迁移性；其次，对于目标攻击场景，TLM-UAP在所有的数据集和分类器上取得了接近100%的目标率，表明了我们提出的方法能够轻易的篡改脑机接口系统的分类结果，使其输出攻击者期望的结果，这可能会造成比非目标攻击更严重的危害。值得一提的是，我们提出的TLM方法第一次将通用对抗扰动从非目标攻击扩展到了目标攻击，据我们所知，这是第一项进行通用对抗扰动目标攻击的研究。最后，我们研究了全通道相同的通用对抗扰动的可行性，使用我们提出的TLM方法给EEG数据中的每个通道添加完全相同的扰动，实验结果表明提高了扰动的通用性之后，会一定程度上牺牲扰动的部分攻击性能，总之，在对抗攻击在脑机接口系统的应用中，我们提出的TLM-UAP能够根据具体任务需求来设计出更合适的通用对抗扰动。为了设计出更安全的脑机接口系统来抵御这两种通用对抗扰动，我们将在下一章进行通用对抗扰动的检测与防御的研究。

4 通用对抗扰动的防御方法

我们的研究表明在脑机接口系统中，针对EEG数据的CNN分类器是非常容易受到TLM-UAP的攻击。这些加入了通用对抗扰动之后对抗样本在肉眼上看上去与原始EEG信号几乎完全相同，但是又会直接改变CNN模型的分类结果，可能会给脑机接口使用者带来一系列的严重的危害（例如攻击者可以使用UAP来控制轮椅到危险的环境中）。因此，我们希望针对EEG数据，搭建出能防御TLM-UAP的CNN模型，从而设计出更安全的脑机接口系统。本章从TLM-UAP的检测和模型鲁棒性研究两个层面对TLM-UAP的防御进行了分析。

4.1 对抗攻击的防御机制

目前，在对抗攻击的防御中，主要可分为四个研究方向：

(1) 输入检测。在模型外部加入检测模型，针对输入来区分正常样本和对抗样本，对于对抗样本采取过滤处理。

(2) 对抗训练。主要思想是在训练模型的过程中，把对抗样本和原始数据结合在一起，训练出更鲁棒的模型。

(3) 修改模型。修改模型的结构，让神经网络模型的结构变得更加简单以及加入一些随机性，全面提高模型的鲁棒性，对对抗扰动的容忍度变高。

(4) 去燥网络。在模型外部添加额外的去噪模块，先对输入数据进行检测和处理，去除其中的扰动信息，尽量还原成原始数据，使其不具有模型攻击能力。

进一步的，上面的防御方式又可以分为两种，一种是数据防御：提前对数据进行检测或预处理，可以对对抗样本实现识别和预警，并拒绝输入给模型；或者直接去掉对抗样本的扰动，还原成正常样本；一种是模型防御：通过某种方式能够得到鲁棒的模型，使模型能够直接防御住对抗样本的攻击。

考虑到在脑机系统的研究中，还没有针对对抗攻击防御的相关的研究，下面我们将从UAP的检测和模型鲁棒性（对抗训练）两个方面对通用对抗扰动进行研究。

4.2 通用对抗扰动的检测

UAP的检测在对抗攻击的防御中属于一种非常直接的手段。在本文第二章和第三章对于DF-UAP和TLM-UAP 的研究中，与其他针对单个样本设计的对抗攻击方法不同，通用对抗扰动由于是单一扰动，相当于整个数据在样本空间沿着同一方向做了一次平移变换，数据的流形变化相对来说更简单。因此，对于通用对抗扰动的检

测，我们需要训练出一个检测模型来对正常样本和对抗样本进行识别，并拒绝对抗扰动的输入操作，如图4-1所示。

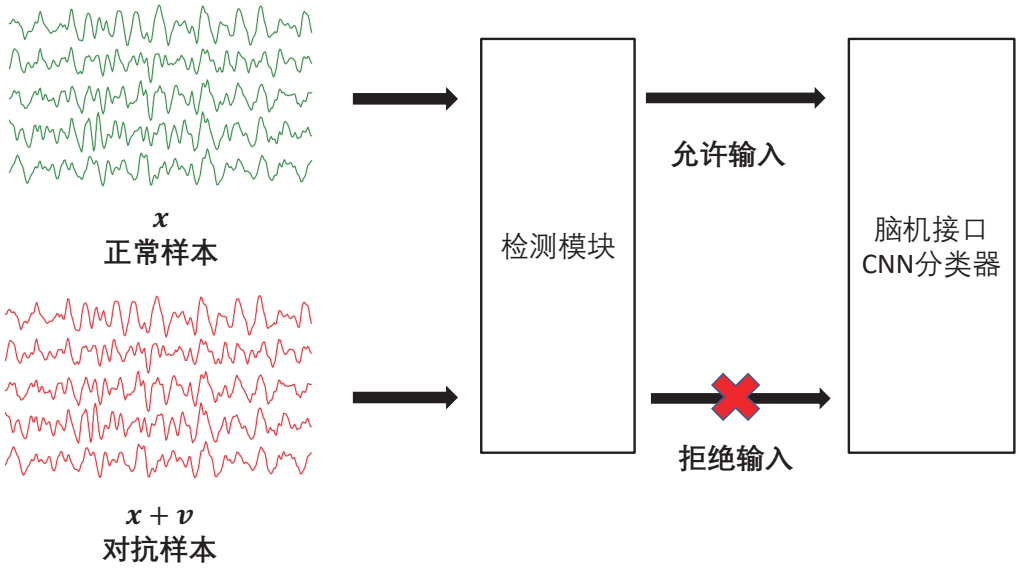


图 4-1 通用对抗扰动的检测模块。

然而，上述检测方法存在的一个弊端是需要知道对抗样本的存在并拥有对抗样本的数据，才能提前训练出检测模型，这种方式在真实场景中的应用可能会受到限制，因为攻击者对系统的攻击往往是带有随机性的，我们通常不能提前获取对抗样本的信息。所以我们更期望能够通过某种方式来得到对对抗样本鲁棒的分类模型，这种防御方式的应用场景会更广泛。

4.3 模型的鲁棒性研究

对抗样本的发现表明我们所使用的深度神经网络模型并没有以一种健壮的方式学习到数据底层的概念，如何学习出一个对于对抗样本具有鲁棒性的模型是一个关键的问题。目前，针对模型的鲁棒性研究是一个非常有趣的研究点。对抗扰动本质上是一种微小的噪声，但是这种噪声可以改变分类器的分类结果。对于一个对对抗样本有鲁棒性的模型，如果对输入数据加一个微小扰动，那么输出也应该变化很小。

在提高神经网络模型对对抗样本鲁棒性的研究中，对抗训练（Adversarial training）是一种最简单最直接的方法。顾名思义，它通过把已经添加完对抗扰动的对抗样本直接加入到训练集中，和原始训练数据一同训练得到最终的模型。这种训练方法不仅提高了模型防御对抗样本的能力，也一定程度上提升了模型的泛化能力。在图像领域中，数据增强（Data augmentation）是一种非常流行的正则化方法，例如

对图片进行水平翻转或者随机旋转；通常上，一般的数据增强方法通过转化训练集数据，一是可以扩充数据量，二是可能将测试集的某种变化提前暴露给模型，使得模型的泛化能力和最终的性能得到提升。对抗训练可以看作是一种特殊的数据增强形式：原始数据中混合了对抗样本，使神经网络模型适应这种扰动，从而提升模型的鲁棒性。

Goodfellow等人[50]发现使用基于快速梯度符号法（Fast gradient sign method，FGSM）的对抗性优化函数（损失函数）进行对抗训练是一种有效的正则化方法：

$$\widetilde{J}(\theta, \boldsymbol{x}, y) = \alpha J(\theta, \boldsymbol{x}, y) + (1-\alpha) J(\theta, \boldsymbol{x} + \xi \text{sign}(\nabla_{\boldsymbol{x}}(J(\theta, \boldsymbol{x}, y)))) \tag{4.1}$$

其中，α一般为0.5，ξ是FGSM方法设置的对抗扰动大小参数，$\boldsymbol{x}$表示输入数据，y是$\boldsymbol{x}$ 对应的真实标签，θ为神经网络模型的参数。从式子中可以看出，该目标函数的主要思想将添加了FGSM设计的对抗扰动的数据集与原始数据集融合到一起，然后进行共同优化。为了更加准确的评价对抗训练的模型的鲁棒性，还需要考虑到该模型对新生成的对抗样本（根据新模型生成）的防御能力。对于已经完成对抗训练的模型，如果我们仍然使用之前参与到对抗训练中的扰动，那么评价的结果是不具有参考性的，因为之前扰动已经在训练集中提前暴露给模型了。实验表明这种对抗训练方法对于重新生成的对抗样本也有一定的防御能力。由于模型的最终输出结果由模型输入和参数权重共同决定，从优化角度来看，公式(4.3)中对抗样本的训练对模型有强正则化作用，使得模型的参数权重减小，那么输入扰动对最后输出结果的影响会变得更小，一定程度上增加了模型对于扰动的鲁棒性。

然而，上述的对抗训练方法在实际应用场景中会存在一个问题：只能对特定的对抗攻击方法有效。目前除了FGSM这种方法以外，还存在很多种对抗攻击方法，我们只能选择出有限种的攻击方法加入到对抗训练当中，如果是一种未知的攻击方法，没有参与到模型对抗训练的过程中，那么这种训练方法很难有防御效果。因此这种训练方式不具有在不同对抗攻击方法中有泛化能力。

除此以外，对于通用对抗扰动，由于在整个数据集只添加了一个扰动，如果我们将添加了通用对抗扰动之后的对抗样本与原始数据合在一起训练，容易让检测模块只学到了一种固定的范式，这样就会导致分类器只能对特定的通用对抗扰动拥有鲁棒性。为了提高模型防御通用对抗扰动的泛化能力，我们使用了投影梯度下降的方法。

4.4 投影梯度下降

基于公式(4.3)的对抗训练方法需要满足损失函数是线性的或者至少是局部线性的假设。如果不是（局部）线性的，那梯度提升的方向就不一定是最优方向。为了解决这个线性假设问题，Madry等人[69] 提出了投影梯度下降（Projected gradient

descent，PGD）的方法来防御基于一阶梯度的对抗攻击，从鲁棒优化的角度研究了神经网络对于对抗样本的鲁棒性。PGD 方法既是一种防御对抗攻击的通用范式，也可以作为一种对抗攻击方法。

首先，对于一个标准的分类任务，我们的目标是找出模型参数θ来最小化风险：

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim D}[l(\theta, \boldsymbol{x}, y)] \tag{4.2}$$

其中，D为$\boldsymbol{x}$的数据分布，$l(\theta, \boldsymbol{x}, y)$是模型的损失函数，一般使用交叉熵（Cross-entropy）。然后使用梯度下降法对损失函数进行最小化，这种优化模型参数的方式又被称为经验风险最小化（Empirical risk minimization，ERM）。为了可靠的训练出一个对对抗扰动鲁棒的模型，需要改变ERM 的训练方式，我们希望最终的模型不再只局限于对特定的攻击方式有防御效果。从ERM这个角度来看，一个正常的神经网络模型是为了让风险损失最小化，那么对抗攻击则是让风险损失最大化，这样才容易造成模型最后的分类结果发生巨大的改变。

假定我们把对抗攻击看做是一种使经验损失最大化的方法，如果我们训练的模型能使这个损失最小，那么该模型的鲁棒性会很强）那么这种思想用数学形式可以表示为求解下面一个极大极小问题:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim D}[\max l(\theta, \boldsymbol{x}+\boldsymbol{v}, y)] \tag{4.3}$$

其中，$\boldsymbol{v}$是添加的对抗扰动。从上式可以看出，训练一个对对抗扰动具有鲁棒性的模型可以转化为求解一个最大最小（鞍点）问题，将公式(4.4) 中的数据样本$\boldsymbol{x}$换成了添加扰动之后的对抗样本。

因此，上述优化鞍点问题的求解可以分为两个部分：内部最大化和外部最小化问题。首先，对于外部最小化问题而言，这是一个典型的ERM形式，与一般的神经网络模型训练相同，我们可以使用随机梯度下降法进行优化。其次，文献[69]表明根据Danskin理论：对于连续可微的函数，内部问题的最大值处的梯度对应于该鞍点问题的下降方向。那么，对于内部最大化问题，我们期望找出一个可行的扰动ξ能使得内部损失函数$l(\theta, \boldsymbol{x}+\xi, y)$ 最大，即找出一种十分有效的对抗攻击方法可以使内部问题最大化。在上文中我们提到，FGSM是一种非常简单且有效的对抗攻击方法，生成对抗样本$\boldsymbol{x}_{adv}$的方式如下所示：

$$\boldsymbol{x}_{adv} = \boldsymbol{x} + \xi \cdot \mathrm{sign}(\nabla_{\boldsymbol{x}} l(\theta, \boldsymbol{x}, y)) \tag{4.4}$$

那么，结合公式(4.4)和公式(4.4)来看，FGSM可以看做是一种基于损失函数一阶梯度的单步（one-step）方法来求解内部最大化问题。除了以外，还有一种更强力的攻击方法，多步FGSM（multi-step），生成对抗样本的方式如下所示:

$$\boldsymbol{x}^{t+1} = \mathrm{Proj}_{\boldsymbol{x}+\mathcal{S}}(\boldsymbol{x}^t + \alpha \cdot \mathrm{sign}(\nabla_{\boldsymbol{x}^t} l(\theta, \boldsymbol{x}^t, y))) \tag{4.5}$$

其中，α为每步对抗攻击的步长参数。文献[70]表明多步FGSM这种攻击方法本质上是在损失函数l上进行投影梯度下降（PGD）。与常用的随机梯度下降（Stochastic gradient descent，SGD）法不同，PGD在每次梯度下降的迭代过程中，对数据进行了一次投影变化，将其约束在可行域中，例如使用PGD进行第k次梯度下降时，参数更新方式如下所示：

$$\overline{\boldsymbol{x}}_k = \mathrm{Proj}_{\boldsymbol{x}+\mathcal{S}}(\boldsymbol{x}_k - s_k \nabla f(\boldsymbol{x}_k)), s_k > 0 \tag{4.6}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k(\overline{\boldsymbol{x}}_k - \boldsymbol{x}_k), \alpha_k \in (0, 1] \tag{4.7}$$

其中，s_k是一个步长参数，α_k是梯度下降中的学习率，我们要求$\alpha_k \in (0, 1]$来保证数据不会移动到可行域外面去。当$\alpha_k = 1$时，上面的算法就变成如下的迭代：

$$\boldsymbol{x}_{k+1} = \mathrm{Proj}_{\boldsymbol{x}+\mathcal{S}}(\boldsymbol{x}_k - s_k \nabla f(\boldsymbol{x}_k)) \tag{4.8}$$

可以看出，多步FGSM和PGD两种表示形式在数学形式上是等价的。实验表明，PGD是一个可以找出上述内部问题极大值的非常不错的方法[69]。如果通过优化公式(4.4)来训练神经网络模型参数，使得PGD 设计出的对抗扰动造成的损失风险最小，那么这个神经网络会是一个对对抗样本具有鲁棒性的模型。PGD从鲁棒优化的角度出发，为对抗训练提供了一个统一的视角，它包含了之前许多关于对抗扰动鲁棒性的研究。

PGD是一种简单有效的对抗训练方法，但是会增加大量的计算代价。在传统的神经网络训练中，模型训练m组（batch）数据只计算了m次梯度；但对于k-step PGD（k为PGD迭代次数参数）而言，最外部也是采用梯度下降法进行求解，除了最外部需要进行m 次梯度以外（获取模型参数的梯度，训练模型），内部最大化问题每次还需要计算k次梯度（获取输出的梯度，寻找扰动），相比于不进行对抗训练的模型，PGD一共需要计算$m(k+1)$次梯度，使得模型整体的训练时间大大增加。

总的来说，PGD是一种防御对抗攻击的通用范式，也可以作为一种对抗攻击方法。它是一种基于FGSM的多步迭代攻击，相比于普通的FGSM仅做一次迭代，PGD是做多次迭代，每次走一小步，每次迭代都会将扰动投射到可行域（规定范围）内。由于每次只走很小的一步，所以局部线性假设基本成立的。经过多步之后就可以达到最优解了，也就是达到最强的攻击效果。文献[69]的实验证明了使用PGD算法生成的攻击样本，对比其他一阶对抗样本有最强的攻击效果。这里所说的一阶对抗样本是指依据一阶梯度的对抗样本，如果模型对PGD产生的样本鲁棒，那基本上就对所有的一阶对抗样本都鲁棒。

DF-UAP和TLM-UAP都属于一阶对抗扰动。DF-UAP是基于DeepFool设计通用对抗扰动，由于DeepFool是一种一阶对抗攻击方法（参考公式()），所以DF-UAP属于

一阶对抗扰动；TLM-UAP的设计是基于公式(3.1)，我们使用的是传统的梯度下降法对目标函数进行优化，因此TLM-UAP也是一种一阶对抗扰动。

针对EEG数据CNN分类器的PGD训练方法的伪代码如算法4所示。

Algorithm 4: 针对EEG数据CNN分类器的PGD训练方法伪代码。

Input: $\boldsymbol{X}_{train} = \{\boldsymbol{x}_{train,i}\}_{i=1}^{n}$, n个EEG训练数据;
θ, CNN分类器模型参数;
T, PGD攻击迭代步数;
α, PGD攻击迭代步长;
s, PGD扰动在ℓ_p范数下最大值（在范围$\mathcal{S}$内）;
M, 最大训练次数;
Output: k, CNN分类器。
随机初始化模型参数θ;
for $m = 1, ..., M$ **do**
  for *每一个小批量数据* $\boldsymbol{D} \in \boldsymbol{X}_{train}$ **do**
    for $\boldsymbol{x} \in \boldsymbol{D}$ **do**
      在数据点$\boldsymbol{x}$上添加范围$\mathcal{S}$内的随机扰动;
      for $t = 1, ..., T$ **do**
        使用公式(4.4)得到PGD对抗样本$\boldsymbol{x}^t$;
      end
      $\boldsymbol{x} = \boldsymbol{x}^t$;
      使用优化器梯度下降更新公式(4.4)中模型参数θ;
    end
  end
end
根据模型参数θ得到CNN分类器k;
Return k。

4.5 实验部分

4.5.1 实验设置

为了研究针对TLM-UAP的防御机制，我们采用了与第二章和第三章完全相同的实验设置，分别在Within-subject和Cross-subject两个不同的脑机接口应用场景上进行了实验。

在TLM-UAP的检测实验中，我们使用与目标模型相同的模型作为检测模型，对

数据集进行了扩充，原始正常EEG样本作为一类，添加了TLM-UAP的EEG样本作为第二类，其中，TLM-UAP在没有约束函数C，扰动大小参数$\xi = 0.2$条件下生成。Within-subject实验中，我们针对每个用户都训练了检测模型，其中，每个用户随机划分80% 的数据作为训练集，20%的数据作为测试集，同时，我们从训练集中随机选择出25%的数据作为验证集来实验早停操作。Cross-subject实验中，我们使用留一法训练检测模型，假设数据集中有N个用户的数据，使用$N-1$个用户作为训练集，剩下一个用户的数据作为测试集。同样的，我们随机打乱这$N-1$个用户的数据，取75%作为训练集，25%作为验证集来实现早停。

在模型鲁棒性研究中，我们使用标准对抗训练和PGD两种方式对三种CNN的模型进行了训练。为了评价模型鲁棒性的泛化能力，所有实验都使用了已经生成好的TLM-UAP（旧TLM-UAP）和新生成的TLM-UAP（新TLM-UAP）两种攻击扰动。其中，在标准对抗训练实验中，旧TLM-UAP被加入到对抗训练中。PGD训练中，使用算法4训练模型，参数设置为迭代步数$T = 5$，攻击迭代步长$\alpha = 0.01$，扰动在ℓ_p范数下最大值s设定为0.05。值得一提的是，两种训练方式都从训练集中取25%作为验证集来实现早停。

4.5.2 实验结果

首先，我们进行了TLM-UAP的检测实验。由于Within-subject和Cross-subject实验在每个数据集上都会生成N个检测模型，我们计算了每个数据集上检测模型识别测试集中正常样本和对抗样本的分类准确率的均值，实验结果如表4.1 所示。从表中可以看出，在两种实验设置中，大多数情况下检测正确率在90%以上，添加了TLM-UAP的对抗样本比较容易被分类器识别出来。该实验表明如果我们提前已知添加通用扰动的对抗样本数据，简单的使用目标模型就可以实现TLM-UAP的检测，后续只需要拒绝将该对抗样本输入到脑机接口系统中就能够起到防御作用。

为了研究脑机接口分类器对于TLM-UAP的鲁棒性，我们先对模型进行了标准对抗训练，实验结果如表4.2所示。从表中可以看出，经过标准对抗训练后，模型被TLM-UAP攻击之后的正确率对比训练前有了很大的提升，表明标准对抗训练过后的模型对之前加入到模型训练的对抗样本（添加旧TLM-UAP的样本）拥有鲁棒性；但是根据新模型（标准对抗训练后）重新设计出的TLM-UAP仍然能够使目标模型的正确率严重下降，经过标准对抗训练后的模型对于新的TLM-UAP 并没有任何防御能力。实验结果表明如果只进行标准对抗训练，新模型对于TLM-的防御能力并不具有泛化性。

然后，我们研究了PGD训练对TLM-UAP的防御性能。我们对模型进行正常训练和PGD训练两种方式，然后使用新旧TLM-UAP攻击正常训练模型和PGD训练模型，其中，新TLM-UAP在正常训练和PGD训练后的模型上生成，在Within-subject实验场景中，RCA和BCA结果分别如表4.3和表4.4所示，其中，PGD迭代步数参数$T = 5$，

表 4.1 三种CNN分类器在三个BCI数据集上对TLM-UAP的检测正确率。其中攻击目标模型的TLM-UAP大小约束参数$\xi = 0.2$。

实验	数据集	目标模型（检测模型）	检测正确率
Within -Subject	P300	EEGNet	0.9957
		DeepCNN	0.9816
		ShallowCNN	0.8654
	ERN	EEGNet	0.9821
		DeepCNN	0.7178
		ShallowCNN	0.7160
	MI	EEGNet	0.9713
		DeepCNN	0.9378
		ShallowCNN	0.9990
Cross -Subject	P300	EEGNet	0.9923
		DeepCNN	0.9361
		ShallowCNN	0.9345
	ERN	EEGNet	0.9988
		DeepCNN	0.9993
		ShallowCNN	0.9923
	MI	EEGNet	0.9993
		DeepCNN	0.9905
		ShallowCNN	1.0000

TLM-UAP的扰动大小参数设置为$\xi = 0.2$。从表中可以看出：

(1) TLM-UAP攻击前，经过PGD训练后的模型的分类精度RCA和BCA没有明显下降，表明PGD训练对模型在正常样本的分类能力上影响不大。

(2) 旧TLM-UAP攻击后，正常训练模型的RCA和BCA指标明显下降，P300和ERN数据集上BCA结果都接近于0.5，MI数据集上BCA接近于0.25，模型将所有测试样本基本上分到了一个类别，表明TLM-UAP攻击成功。然而，PGD训练的模型的RCA和BCA对比正常训练模型的结果有明显提高，表明经过PGD训练的模型对旧TLM-UAP有一定的防御能力。

(3) 新TLM-UAP攻击后，正常训练模型的RCA和BCA仍然显著下降，模型被TLM-UAP攻击成功。PGD训练的模型的RCA和BCA仍然高于正常训练模型，，表明PGD训练提高了模型的鲁棒性，并对TLM-UAP的防御效果有一定的泛化能力。

(4) 新TLM-UAP攻击PGD训练模型的结果比旧TLM-UAP攻击后的结果要低，实验结果表明，重新优化生成的新TLM-UAP仍然会对PGD训练的模型有危害，攻击与

表 4.2 Cross-subject实验场景，标准对抗训练前后，模型在正常样本和对抗样本上的RCA。其中，TLM-UAP 的扰动大小参数设置为$\xi = 0.2$。

数据集	目标模型	标准对抗训练前		标准对抗训练后	
		正常样本	对抗样本	对抗样本 旧TLM-UAP	对抗样本 新TLM-UAP
P300	EEGNet	0.6832	0.1697	0.6231	0.1782
	DeepCNN	0.6940	0.1764	0.6125	0.2131
	ShallowCNN	0.6679	0.1894	0.6275	0.2119
ERN	EEGNet	0.6669	0.2923	0.6093	0.3449
	DeepCNN	0.6520	0.3349	0.5822	0.4215
	ShallowCNN	0.6776	0.2925	0.6199	0.2925
MI	EEGNet	0.4437	0.2500	0.4118	0.2500
	DeepCNN	0.4660	0.2542	0.4409	0.3337
	ShallowCNN	0.4738	0.2500	0.4738	0.2502

防御是一个博弈的过程，模型对于UAP的鲁棒性还存在提升空间。

表 4.3 Within-subject实验场景，PGD训练前后，模型在正常样本和对抗样本上的RCA结果。

数据集	目标模型	TLM-UAP攻击前		旧TLM-UAP攻击		新TLM-UAP攻击	
		正常训练	PGD训练	正常训练	PGD训练	正常训练	PGD训练
P300	EEGNet	0.7899	0.8151	0.1689	0.5742	0.1689	0.3278
	DeepCNN	0.8384	0.8567	0.1747	0.8285	0.1713	0.6588
	ShallowCNN	0.8004	0.8447	0.3406	0.8161	0.2050	0.6031
ERN	EEGNet	0.6884	0.7013	0.3401	0.6985	0.3272	0.5055
	DeepCNN	0.7169	0.7188	0.5046	0.7187	0.3888	0.6176
	ShallowCNN	0.6939	0.7463	0.5037	0.7269	0.4136	0.6544
MI	EEGNet	0.5019	0.4195	0.2375	0.4176	0.2433	0.3487
	DeepCNN	0.5450	0.4282	0.2644	0.4148	0.2692	0.3688
	ShallowCNN	0.7634	0.4866	0.2768	0.4741	0.2749	0.3755

值得一提的是，上述的实验结果表明，加入对抗训练后，看RCA结果指标来评价模型的分类能力并不准确，模型可能只是将测试样本都分到了多数类，造成分类性能高的假象。因此，在下面的实验分析中，我们只列出了BCA结果。

此外，我们做了额外的实验研究了模型容量与鲁棒性的关系。我们增加了三种CNN网络（EEGNet，DeepCNN和ShallowCNN）卷积层的卷积核数，将模型参数都提高到原来的四倍，然后使用PGD方法训练模型，与原始模型的对比实验结果如

表 4.4 Within-subject实验场景，PGD训练前后，模型在正常样本和对抗样本上的BCA结果。

数据集	目标模型	TLM-UAP攻击前		旧TLM-UAP攻击		新TLM-UAP攻击	
		正常训练	PGD训练	正常训练	PGD训练	正常训练	PGD训练
P300	EEGNet	0.7907	0.7530	0.5011	0.6754	0.5011	0.5740
	DeepCNN	0.8035	0.6274	0.5027	0.6934	0.5027	0.6511
	ShallowCNN	0.7660	0.6442	0.5641	0.6749	0.5641	0.6541
ERN	EEGNet	0.6336	0.6582	0.5000	0.6680	0.5000	0.5753
	DeepCNN	0.7073	0.6411	0.5916	0.6581	0.5916	0.6192
	ShallowCNN	0.6484	0.6391	0.5255	0.6301	0.5255	0.5803
MI	EEGNet	0.5013	0.4202	0.2500	0.4187	0.2500	0.3436
	DeepCNN	0.5437	0.4310	0.2581	0.4159	0.2681	0.3586
	ShallowCNN	0.7601	0.4832	0.2753	0.4718	0.2726	0.3714

表4.5所示。实验结果表明，对于针对EEG数据的CNN分类器，在大多数情况下增大模型的容量能够提高模型对TLM-UAP的防御能力。但是在大部分实验中大模型容量的BCA结果没有明显提高，这是因为在脑机接口系统中，针对EEG数据的CNN网络结构通常十分简单，网络层数和模型参数里都很少（例如在EEGNet中只有1809个权重参数），小模型对于EEG数据的拟合能力已经足够，继续增大模型容量反而使模型过于冗余，导致过拟合，以至于模型正常分类能力减弱。

表 4.5 模型容量对TLM-UAP防御能力的影响，BCA结果。

数据集	目标模型	PGD训练			
		旧TLM-UAP		新TLM-UAP	
		原始模型	大模型	原始模型	大模型
P300	EEGNet	0.5962	0.6918	0.6179	0.6553
	DeepCNN	0.5127	0.5322	0.5136	0.5584
	ShallowCNN	0.5103	0.5671	0.5084	0.5916
ERN	EEGNet	0.6146	0.6387	0.5848	0.5988
	DeepCNN	0.5915	0.6314	0.5977	0.6078
	ShallowCNN	0.5294	0.6034	0.5393	0.6104
MI	EEGNet	0.4187	0.4149	0.3436	0.3504
	DeepCNN	0.4159	0.3471	0.3586	0.3195
	ShallowCNN	0.4718	0.4758	0.3714	0.3919

最后，我们研究了PGD训练的超参数迭代步数T对模型防御TLM-UAP的影响，实验结果如表4.6所示。从表中可以看出，对于EEG数据，迭代步数T较小时，才能

确保PGD训练不会严重影响到模型对于正常样本的分类能力。否则，如果迭代步数T过大，PGD训练后的模型会失去正常的分类能力，将所有正常样本都分到了同一类。

表 4.6 PGD训练迭代步数T对TLM-UAP防御能力的影响，BCA结果。

数据集	目标模型	正常训练	PGD训练			
			T=3	T=5	T=10	T=30
P300	EEGNet	0.7907	0.7530	0.5011	0.6754	0.5011
	DeepCNN	0.8035	0.6274	0.5027	0.6934	0.5027
	ShallowCNN	0.7660	0.6442	0.5641	0.6749	0.5641
ERN	EEGNet	0.6336	0.6582	0.5000	0.6680	0.5000
	DeepCNN	0.7073	0.6411	0.5916	0.6581	0.5916
	ShallowCNN	0.6484	0.6391	0.5255	0.6301	0.5255
MI	EEGNet	0.5013	0.4202	0.2500	0.4187	0.2500
	DeepCNN	0.5437	0.4310	0.2581	0.4159	0.2681
	ShallowCNN	0.7601	0.4832	0.2753	0.4718	0.2726

4.6 本章小结

本章介绍了针对我们提出的TLM-UAP的防御方法。我们从通用对抗扰动的检测和模型鲁棒性两个层面出发，分别研究了防御TLM-UAP的方法性能。首先，在提前已知对抗样本的场景中，我们搭建了TLM-UAP的检测模块，通过使用目标模型直接训练出检测模型，实验结果表明该模型的对对抗样本的识别率非常高，在大多数情况下检测正确率在90%以上。其次，对于未知对抗样本的场景中，我们使用对抗训练的方式来增强脑机接口CNN分类器的模型鲁棒性。我们先对模型进行了标准对抗训练，实验结果表明这种方式只能对特定已知的TLM-UAP有防御效果，对新生成的TLM-UAP并没有鲁棒性。然后，我们使用了投影梯度下降（PGD）方法训练CNN模型，实验表明PGD训练使得模型对于TLM-UAP攻击的鲁棒性增强，并对已知和新生成的TLM-UAP都有防御效果，防御能力具有泛化性。最后，我们分别对模型容量和PGD训练的迭代步数与模型的防御能力的关系进行了研究。我们希望针对TLM-UAP的防御机制研究能对设计出更安全的脑机接口系统提供帮助。

5 总结与展望

5.1 总结

通用对抗扰动（UAP）是一种十分微小且对于所有数据完全相同的扰动。目前，有多个专门针对脑机接口EEG数据的CNN分类器被提出，然而，本文的研究表明这些CNN分类器非常容易被UAP攻击。UAP可以通过提前离线计算得到通用扰动模板，当把它添加到正常的EEG数据样本上后，其强大的攻击能力能够显著降低CNN分类器的性能。针对基于EEG数据的脑机接口系统，我们提出了基于DeepFool的通用对抗扰动（DF-UAP）和基于总损失最小化的通用对抗扰动（TLM-UAP）两种通用扰动设计算法，实验结果表明这两种UAP能够显著的降低脑机接口CNN分类器的分类精度，并且在不同的模型之间具有迁移性。为了设计出更安全的脑机接口系统，在不同应用场景我们提出了两种针对TLM-UAP的防御机制。实验结果表明，在已知对抗样本的场景中，在目标模型之前嵌入UAP检测模块能够成功识别添加了扰动的对抗样本；在未知对抗样本的场景中，标准对抗训练只能对特定已知的TLM-UAP有防御能力，然而通过PGD训练能够提高CNN模型的鲁棒性，对新生成的TLM-UAP仍然有防御效果。

本文的主要内容包括以下几个部分：

(1) 介绍了DF-UAP，首次将UAP的思想引入到脑机接口系统中。在白盒和灰盒的非攻击场景中都验证了DF-UAP攻击的有效性。

(2) 介绍了我们提出的TLM-UAP，设计算法更为简洁、可扩展性更强，并首次将UAP从非目标攻击推广到目标攻击。实验结果表明，在非目标攻击场景中TLM-UAP对比DF-UAP有更强的攻击性和通用性，并在目标攻击场景取得了接近100%的目标率。

(3) 首次在脑机接口系统中进行了TLM-UAP防御机制的研究。实验结果表明搭建UAP检测模块和PGD训练这两种方式能够分别在不同应用场景对TLM-UAP进行防御。

我们的研究揭示了脑机接口系统中存在的一个关键的安全问题，这项工作将为设计出更安全的脑机接口系统提供帮助。

5.2 展望

当前的UAP攻击和防御还存在一些不足，我们准备进一步研究如下问题：

(1) 目前的DF-UAP和TLM-UAP设计算法只能针对CNN分类器，在脑机接口中依然存在一些使用传统机器学习算法搭建的系统，为了进一步提高UAP的攻击范围，

我们需要设计出一种针对传统机器学习算法的UAP攻击算法。

(2) 我们对UAP的防御只进行了初步的研究，UAP攻击可能会成为非法分子攻击脑机接口系统的一种手段，我们需要搭建出对于对抗攻击更鲁棒的模型，设计出更安全的脑机接口系统。

(3) 目前，对抗攻击是一种主要集中在在测试阶段应用的攻击手段，在一些封闭式系统中，无法对原始数据进行篡改会限制对抗攻击的成功应用。因此，如何在训练阶段就对模型进行攻击，使得模型的性能降低，这将成为我们下一步的研究方向。

致　　谢

至此，毕业论文的相关工作即将完成。时光荏苒，不知不觉中三年的研究生生涯即将结束，我的学生时代也到了尾声。进入HUST BCI & ML实验室接近四年，在这期间，我成长了许多。

首先，我要感谢我的导师伍冬睿教授。不论是在科研还是在做人方面，伍老师都是一名尽职尽责的好老师，给我们树立了一个好的榜样。在我大四初进实验室时，开始接触到机器学习这个新领域，内心是充满未知与迷惘的。伍老师在跟我的交流中，耐心的帮我解答了心中每一个疑惑，并给我指明了好的研究方向。研究生正式加入实验室后，伍老师治学严谨的科研态度，在实验室营造了一种良好的精神氛围。在伍老师的指导下，我思考问题的方式也逐渐变得更深更广，科研能力也得到了非常大的提高。此外，伍老师对于我的所有选择都给予了充分的理解，并支持我的每一个决定。俗话说，严师出高徒，在跟伍老师的相处中，我学习到了许多做人做事的道理，对待每一位合作者都要尊重，办事效率要快，真心真诚的对待身边的每一个人。千言万语汇成一句话，衷心的感谢伍老师您对我的栽培与付出，我毕生难忘。

然后，我要感谢恩明楼302实验室的各位小伙伴。在我对科研和生活有疑问的时候，你们和我一起讨论，给我提供了无数的帮助与支持。每个人身上都有着独特的闪光点，值得人去学习和交流。十分荣幸在我的研究生生涯中有这样一群最可爱的人的陪伴，也希望实验室的大家在未来的日子越来越好，年少有为，实现每个人的理想。然后，我还要感谢我的室友，生活上的朝夕相处，我们共同成长，相互鼓励，一起度过失落和快乐的时光，这是我生命中一段难忘且珍贵的回忆，你们是我一辈子的好朋友、好兄弟。

最后，感谢我的父母。你们是我最坚实的后盾，为我默默的付出了一切，你们是我最亲的家人。因为有你们，我不再害怕困难和险阻；因为有你们，才能让我在追逐梦想的航路上，顺利起航；因为有你们，我会更加坚定的去努力，奋斗！

参考文献

[1] Sutton S, Braren M, Zubin J, et al. Evoked-Potential Correlates of Stimulus Uncertainty. Science, 1965, 150(3700):1187–1188.

[2] Farwell L, Donchin E. Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology, 1988, 70(6):510–523.

[3] Wu D. Online and Offline Domain Adaptation for Reducing BCI Calibration Effort. IEEE Trans. on Human-Machine Systems, 2017, 47(4):550–563.

[4] Wu D, Lawhern V J, Hairston W D, et al. Switching EEG Headsets Made Easy: Reducing Offline Calibration Effort Using Active Weighted Adaptation Regularization. IEEE Trans. on Neural Systems and Rehabilitation Engineering, 2016, 24(11):1125–1137.

[5] Pfurtscheller G, Neuper C. Motor imagery and direct brain-computer communication. Proc. IEEE, 2001, 89(7):1123–1134.

[6] Zhu D, Bieger J, Garcia Molina G, et al. A survey of stimulation methods used in SSVEP-based BCIs. Computational Intelligence and Neuroscience, 2010, page 702357.

[7] Schirrmeister R T, Springenberg J T, Fiederer L D J, et al. Deep learning with convolutional neural networks for EEG decoding and visualization. Human Brain Mapping, 2017, 38(11):5391–5420.

[8] Bashivan P, Rish I, Yeasin M, et al. Learning Representations from EEG with Deep Recurrent-Convolutional Neural Networks. in: Proc. Int'l Conf. on Learning Representations, San Juan, Puerto Rico, May, 2016.

[9] Tabar Y R, Halici U. A novel deep learning approach for classification of EEG motor imagery signals. Journal of Neural Engineering, 2017, 14(1):016003.

[10] Tayeb Z, Fedjaev J, Ghaboosi N, et al. Validating deep neural networks for online decoding of motor imagery movements from EEG signals. Sensors, 2019, 19(1):210.

[11] Vidal J J. Toward direct brain-computer communication. Annual review of Biophysics and Bioengineering, 1973, 2(1):157–180.

[12] Fetz E E. Operant conditioning of cortical unit activity. Science, 1969, 163(3870):955–958.

[13] Elbert T, Rockstroh B, Lutzenberger W, et al. Biofeedback of slow cortical potentials. I. Electroencephalography and Clinical Neurophysiology, 1980, 48(3):293–301.

[14] Farwell L A, Donchin E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology, 1988, 70(6):510–523.

[15] Kuhlman W N. EEG feedback training: enhancement of somatosensory cortical activity. Electroencephalography and Clinical Neurophysiology, 1978, 45(2):290–294.

[16] Wolpaw J R, McFarland D J, Neat G W, et al. An EEG-based brain-computer interface for cursor control. Electroencephalography and Clinical Neurophysiology, 1991, 78(3):252–259.

[17] Wolpaw J R, McFarland D J. Multichannel EEG-based brain-computer communication. Electroencephalography and Clinical Neurophysiology, 1994, 90(6):444–449.

[18] Wolpaw J R, McFarland D J. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. Proc. of the National Academy of Sciences, 2004, 101(51):17849–17854.

[19] Hochberg L R, Serruya M D, Friehs G M, et al. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. Nature, 2006, 442(7099):164–171.

[20] Krusienski D J, Shih J J. Control of a visual keyboard using an electrocorticographic brain-computer interface. Neurorehabilitation and Neural Repair, 2011, 25(4):323–331.

[21] Leuthardt E C, Gaona C, Sharma M, et al. Using the electrocorticographic speech network to control a brain-computer interface in humans. Journal of Neural Engineering, 2011, 8(3):036004.

[22] Suthana N, Haneef Z, Stern J, et al. Memory enhancement and deep-brain stimulation of the entorhinal area. New England Journal of Medicine, 2012, 366(6):502–510.

[23] Wang W, Collinger J L, Degenhart A D, et al. An electrocorticographic brain interface in an individual with tetraplegia. PloS one, 2013, 8(2).

[24] Grau C, Ginhoux R, Riera A, et al. Conscious brain-to-brain communication in humans using non-invasive technologies. PloS one, 2014, 9(8).

[25] King C E, Wang P T, McCrimmon C M, et al. The feasibility of a brain-computer interface functional electrical stimulation system for the restoration of overground walking after paraplegia. Journal of Neuroengineering and Rehabilitation, 2015, 12(1):80.

[26] Herff C, Schultz T. Automatic speech recognition from neural signals: a focused review. Frontiers in Neuroscience, 2016, 10:429.

[27] Pandarinath C, Nuyujukian P, Blabe C H, et al. High performance communication by people with paralysis using an intracortical brain-computer interface. Elife, 2017, 6:e18554.

[28] Perdikis S, Tonin L, Saeedi S, et al. The Cybathlon BCI race: Successful longitudinal mutual learning with two tetraplegic users. PLoS biology, 2018, 16(5):e2003787.

[29] Heelan C, Lee J, OShea R, et al. Decoding speech from spike-based neural population recordings in secondary auditory cortex of non-human primates. Communications biology, 2019, 2(1):1–12.

[30] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proc. IEEE, 1998, 86(11):2278–2324.

[31] LeCun Y, Bengio Y, Hinton G. Deep Learning. Nature, 2015, 521(7553):436–444.

[32] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation, 1997, 9(8):1735–1780.

[33] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. in: Proc. Advances in Neural Information Processing Systems, Lake Tahoe, NV, December, 2012, 1097-1105.

[34] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. CoRR, 2014, abs/1409.1556.

[35] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Boston, MA, June, 2015, 1–9.

[36] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. in: Proc. Int'l Conf. on Machine Learning, Lille, France, July, 2015, 448-456.

[37] Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the inception architecture for computer vision. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, June, 2016, 2818-2826.

[38] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning. in: Proc. 31th AAAI Conf. on Artificial Intelligence (AAAI), San Francisco, CA, February, 2017.

[39] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, June, 2016, 770-778.

[40] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, June, 2014, IEEE, 580–587.

[41] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, June, 2016, 779–788.

[42] Redmon J, Farhadi A. YOLO9000: better, faster, stronger. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, June, 2017, 7263–7271.

[43] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2017, 39(12):2481–2495.

[44] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. in: Proc. Int'l Conf. on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, October, 2015, 234–241.

[45] Graves A, Mohamed A r, Hinton G. Speech recognition with deep recurrent neural networks. in: Proc. 38th Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Vancouver, Canada, May, 2013, 6645–6649.

[46] Graves A, Jaitly N. Towards end-to-end speech recognition with recurrent neural networks. in: Proc. 31th Int'l Conf. on Machine Learning (ICML), Beijing, China, June, 2014, 1764–1772.

[47] Kalchbrenner N, Danihelka I, Graves A. Grid long short-term memory. CoRR, 2015, abs/1507.01526.

[48] Lawhern V J, Solon A J, Waytowich N R, et al. EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces. Journal of Neural Engineering, 2018, 15(5):056013.

[49] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing properties of neural networks. in: Proc. Int'l Conf. on Learning Representations, Banff, Canada, April, 2014.

[50] Goodfellow I J, Shlens J, Szegedy C. Explaining and Harnessing Adversarial Examples. in: Proc. Int'l Conf. on Learning Representations, San Diego, CA, May, 2015.

[51] Kurakin A, Goodfellow I J, Bengio S. Adversarial examples in the physical world. in: Proc. Int'l Conf. on Learning Representations, Toulon, France, April, 2017.

[52] Brown T B, Mané D, Roy A, et al. Adversarial Patch. CoRR, 2017, abs/1712.09665.

[53] Athalye A, Engstrom L, Ilyas A, et al. Synthesizing Robust Adversarial Examples. in: Proc. 35th Int'l Conf. on Machine Learning, Stockholm, Sweden, July, 2018, 284-293.

[54] Carlini N, Wagner D A. Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. CoRR, 2018, abs/1801.01944.

[55] Moosavi-Dezfooli S M, Fawzi A, Fawzi O, et al. Universal adversarial perturbations. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, HI, July, 2017, 1765-1773.

[56] Moosavi-Dezfooli S M, Fawzi A, Frossard P. Deepfool: A simple and accurate method to fool deep neural networks. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, NV, June, 2016, 2574–2582.

[57] Neekhara P, Hussain S, Pandey P, et al. Universal adversarial perturbations for speech recognition systems. CoRR, 2019, abs/1905.03828.

[58] Behjati M, Moosavi-Dezfooli S M, Baghshah M S, et al. Universal Adversarial Attacks on Text Classifiers. in: Proc. IEEE Int'l Conf. on Acoustics, Speech and Signal Processing, Brighton, United Kingdom, May, 2019, 7345–7349.

[59] Mopuri K R, Ganeshan A, Radhakrishnan V B. Generalizable data-free objective for crafting universal adversarial perturbations. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2019, 41(10):2452–2465.

[60] Carlini N, Wagner D. Towards evaluating the robustness of neural networks. in: Proc. IEEE Symposium on Security and Privacy, San Jose, CA, May, 2017, IEEE, 39–57.

[61] Papernot N, McDaniel P, Goodfellow I, et al. Practical black-box attacks against machine learning. in: Proc. ACM Asia Conf. on Computer and Communications Security, Abu Dhabi, UAE, April, 2017, ACM, 506-519.

[62] Su J, Vargas D V, Sakurai K. One pixel attack for fooling deep neural networks. CoRR, 2017, abs/1710.08864.

[63] Zhang X, Wu D. On the Vulnerability of CNN Classifiers in EEG-Based BCIs. IEEE Trans. on Neural Systems and Rehabilitation Engineering, 2019, 27(5):814–825.

[64] Li J, Cheng K, Wang S, et al. Feature Selection: A Data Perspective. CoRR, 2016, abs/1601.07996.

[65] Hoffmann U, Vesin J M, Ebrahimi T, et al. An efficient P300-based brain-computer interface for disabled subjects. Journal of Neuroscience Methods, 2008, 167(1):115–125.

[66] Margaux P, Emmanuel M, Sebastien D, et al. Objective and Subjective Evaluation of Online Error Correction during P300-Based Spelling. Advances in Human-Computer Interaction, 2012, 2012(578295):13.

[67] Tangermann M, Muller K R, Aertsen A, et al. Review of the BCI Competition IV. Frontiers in Neuroscience, 2012, 6:55.

[68] Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions. in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, HI, July, 2017, 1800–1807.

[69] Madry A, Makelov A, Schmidt L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks. in: Proc. Int'l. Conf. on Learning Representations, Vancouver, Canada, May, 2018.

[70] Kurakin A, Goodfellow I J, Bengio S. Adversarial Machine Learning at Scale. CoRR, 2016, abs/1611.01236.

附录 1　攻读学位期间发表论文目录

[1] Zihan Liu, Xiao Zhang and Dongrui Wu. ”Universal Adversarial Perturbations for CNN Classifiers in EEG-Based BCIs”, IEEE Trans. on Human-Machine Systems, 2020, submitted.

[2] Zihan Liu, Bo Huang, Yuqi Cui, et al. ”Multi-Task Deep Learning with Dynamic Programming for Embryo Early Development Stage Classification from Time-Lapse Videos”, IEEE Access, 2019, 7: 122153-122163.

[3] Zihan Liu and Dongrui Wu. ”Unsupervised Ensemble Learning for Class Imbalance Problems”. Chinese Automation Congress (CAC), Xi’an, China, Nov. 2018, pp. 3593-3600.