Adversarial Artifact Detection in EEG-Based Brain-Computer Interfaces

Xiaoqing Chen, Dongrui Wu
School of Artificial Intelligence and Automation
Huazhong University of Science and Technology, Wuhan, China
Email: m202273198@hust.edu.cn, drwu@hust.edu.cn

***Abstract*—Machine learning has achieved great success in electroencephalogram (EEG) based brain-computer interfaces (BCIs). Most existing BCI research focused on improving its accuracy, but few had considered its security. Recent studies, however, have shown that EEG-based BCIs are vulnerable to adversarial attacks, where small perturbations added to the input can cause misclassification. Detection of adversarial examples is crucial to both the understanding of this phenomenon and the defense. This paper, for the first time, explores adversarial detection in EEG-based BCIs. Experiments on two EEG datasets using three convolutional neural networks were performed to verify the performances of multiple detection approaches. We showed that both white-box and black-box attacks can be detected, and the former are easier to detect.**

***Index Terms*—Brain-computer interface, electroencephalogram, adversarial example, adversarial detection, security**

I. Introduction

A brain-computer interface (BCI), which has been extensively studied in neuroscience, neural engineering and clinical rehabilitation, builds a communication pathway between the human brain and a computer [1]. Electroencephalogram (EEG), which records the brain's electrical activities from the scalp, has become the most widely used input signal in BCIs due to its low cost and convenience [2]. An EEG-based BCI system usually consists of four parts, namely signal acquisition, signal preprocessing, machine learning, and control action, as shown in Fig. 1. The machine learning block includes feature extraction and classification/regression if traditional machine learning algorithms are adopted.

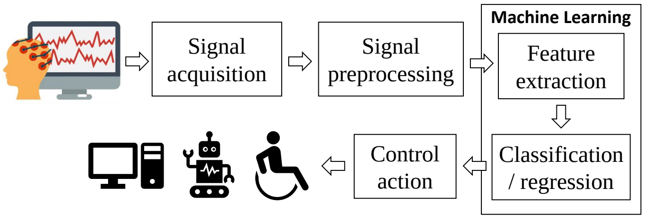


Fig. 1. The flowchart of a BCI system. Manual feature extraction is necessary if traditional machine learning algorithms are employed.

Machine learning has achieved great success in many applications [3]–[5], including EEG-based BCIs. Despite their outstanding performance and robustness to random noise, machine learning models, especially deep learning models, are vulnerable to adversarial attacks, where carefully crafted human-imperceptible perturbations are added to benign samples to cause mis-recognitions [6]–[8]. The existence of adversarial examples raised wide attention and serious security concern about using machine learning models in safety/privacy-critical applications, such as malware detection [9], speech recognition [10], autonomous driving [11], etc.

For BCIs, most studies so far focused on increasing the accuracy and efficiency of machine learning algorithms, but few considered their security. However, as first revealed by Zhang and Wu [12], adversarial examples generated by unsupervised fast gradient sign method (FGSM) [7] can significantly degrade the performance of deep learning classifiers in EEG-based BCIs [13]. Meng *et al.* [14] further exposed the vulnerability of machine learning algorithms in regression tasks of EEG-based BCIs. Zhang *et al.* [15] showed that adversarial examples can fool BCI spellers to output any wrong character the attacker wants. Liu *et al.* [16] designed a total loss minimization approach to craft universal adversarial perturbations for EEG-based BCIs, making adversarial attacks easier to implement. Bian *et al.* [17] used simple square wave signals to generate adversarial examples, which can mislead steady-state visual evoked potential based BCIs. Jiang *et al.* [18] used active learning for efficient poisoning attacks to EEG-based BCIs.

The consequences of adversarial attacks to BCIs could range from merely user frustration to severe injury, raising a critical safety concern and an urgent need for adversarial defense. For example, adversarial attacks can cause malfunctions in exoskeletons or wheelchairs controlled by EEG-based BCIs [19], and may drive the user into danger on purpose. In BCI-based driver drowsiness estimation [20], adversarial attacks may hijack the output of the BCI system and increase the possibility of traffic accidents. More importantly, in military settings, adversarial attacks in BCIs may result in false commands, such as friendly fire [21]. Therefore, it is critical to develop adversarial defense approaches for BCIs.

In the literature, multiple adversarial defense approaches have been proposed for other applications, e.g., computer vision and natural language processing, which can be divided into proactive defenses and reactive defenses [22]. Proactive defenses, such as adversarial training [23], [24], aim at obtaining a robust classifier. Reactive defenses [25]–[27] attempt to identify adversarial examples to reject them or restore them to normal ones.

Though various approaches have been proposed to tackle adversarial attacks, challenges still exist [28], [29]. In adversarial training approaches (proactive defense), the inclusion of adversarial examples into training can not only incur extra training overheads but also compromise the model's performance on benign examples, and classifiers trained to accommodate a certain attack may later be defeated by new attack strategies. Aimed at exploring the intrinsic properties of adversarial examples, adversarial detection approaches (reactive defense) based on designed statistics have shown satisfactory robustness to known optimization-based attacks and generalization ability toward unseen attacks. However, they are usually task-specific, e.g., for image classification only [25]–[27], [30]. A given detection approach performing well for one task may perform poorly for another. To our knowledge, adversarial detection in EEG-based BCIs has not been studied yet.

This paper implements and compares several state-of-the-art adversarial detection approaches for EEG-based BCIs. Experiments on two EEG datasets and three convolutional neural network (CNN) models showed that by extracting features from the output of neural networks, we can effectively distinguish between white-box adversarial examples and normal ones.

The remainder of this paper is organized as follows: Section II introduces several adversarial attacks used in our experiments. Section III describes our adversarial detection strategies. Section IV details our performance evaluation settings. Section V presents the detection results. Section VI draws conclusions and points out some future research directions.

II. Adversarial Attacks

Let $C_{\boldsymbol{\theta}}(X) : \mathcal{X} \to \mathcal{Y}$ be an EEG classifier, where $\boldsymbol{\theta}$ is model parameters, $X \in \mathcal{X} \subset \mathbb{R}^{C \times T}$ is an EEG epoch where C stands for the number of EEG channels and T the number of time domain samples, and $\mathcal{Y} = \{1, 2, ..., k\}$, with k being the number of classes. Let X^{adv} be an adversarial example generated from X, and y the ground-truth label for X. In adversarial attacks, X^{adv} needs to satisfy:

$$C_{\boldsymbol{\theta}}(X^{adv}) \neq y, \tag{1}$$

$$D(X^{adv}, X) < \epsilon, \tag{2}$$

where D is a given distance metric, and ϵ is a predefined perturbation threshold. A common choice of D is ℓ_p norm, and ℓ_∞ is used in this paper.

To satisfy the above constraints, X^{adv} can be re-expressed as [24]:

$$X^{adv} = X + \underset{\|\boldsymbol{\delta}\| < \epsilon}{\arg\max}\, \mathcal{L}(C_{\boldsymbol{\theta}}(X + \boldsymbol{\delta}), y), \tag{3}$$

where $\mathcal{L}$ is the classifier's loss function, $\| \cdot \|$ stands for ℓ_p norm, and $\boldsymbol{\delta}$ is the calculated adversarial perturbation.

According to how much the attacker knows about the target model, adversarial attacks can be categorized into white-box attacks and black-box attacks. White-box attacks assume the adversary has full access to the model architecture and parameters, whereas in black-box attacks, the adversary can only acquire the model's predicted classes or probabilities. Most white-box attacks are designed based on optimization or gradient strategies, and black-box attacks are implemented based on the transferability between models or by querying the model outputs. Black-box attacks are more challenging to implement than white-box attacks due to less knowledge acquired, but are more practical.

To fully explore the characteristics of different attacks, this paper implements two gradient-based white box attacks, i.e., FGSM [7] and projected gradient descent (PGD) [24], one optimization-based white-box attack, i.e., Carlini-Wagner (CW) attack [31], and one transferability-based black-box attack, to EEG-based BCIs.

A. Fast Gradient Sign Method (FGSM)

FGSM [7], a simple yet powerful attack approach, crafts the adversarial example via one-step gradient calculation:

$$X^{adv} = X + \epsilon \cdot \mathrm{sign}(\nabla_X \mathcal{L}(C_{\boldsymbol{\theta}}(X), y)). \tag{4}$$

Along the direction of the gradient, the loss of the model output with respect to the ground-truth label is increased, forcing the model to misclassify. Increasing $\boldsymbol{\epsilon}$ usually improves the attack success rate.

B. Projected Gradient Descent (PGD)

PGD [24] applies FGSM multiple times with a smaller step size, and bounds the magnitude of the generated adversarial perturbation after each iteration.

More specifically, PGD starts from a random point near the benign example X:

$$X_0^{adv} = X + \boldsymbol{\xi}, \tag{5}$$

where $\boldsymbol{\xi} \in \mathcal{U}(-\epsilon, \epsilon)$ is random noise, and then iterates:

$$X_i^{adv} = \mathrm{Proj}_{X,\epsilon} \left(X_{i-1} + \alpha \cdot \mathrm{sign}(\nabla_{X_{i-1}} \mathcal{L}(C_{\boldsymbol{\theta}}(X_{i-1}), y)) \right), \tag{6}$$

where $\alpha \leq \epsilon$ is the attack step size, and $i = 1, 2, ..., n_{iter}$, with n_{iter} being the number of iterations. $\mathrm{Proj}_{X,\epsilon}$ clips the value of the input to keep X_i^{adv} in the ϵ neighborhood of the benign example X under ℓ_∞ norm.

C. Carlini-Wagner (CW) Attack

CW attack was proposed by Carlini and Wagner [31], which is now one of the strongest white-box attacks. Carlini and Wagner formulated adversarial attack as the following optimization problem:

$$\min_{\boldsymbol{\delta}} \|\boldsymbol{\delta}\|_2 + c \cdot h(X + \boldsymbol{\delta}), \tag{7}$$

where c is a trade-off parameter found by binary search during the optimization process, and $h(\cdot)$ is designed in such a way that $h(X + \boldsymbol{\delta}) \leq 0$ if and only if $C_{\boldsymbol{\theta}}(X + \boldsymbol{\delta}) \neq y$. Carlini and Wagner designed $h(X')$ as a hinge loss:

$$h(X') = \max \left(Z(X')_y - \max_{i \neq y} Z(X')_i, -\kappa \right), \tag{8}$$

where $Z(X')$ is the output of the classifier's penultimate layer, and κ is a hyperparameter called confidence. An adversarial example with a higher κ usually has larger perturbation size and better transferability.

For a given example X, CW attack tries to find a perturbation $\boldsymbol{\delta}$ that is small in size ($\|\cdot\|$ in the loss function) but can mislead the classifier [$h(\cdot)$ in the loss function].

After optimization, we clip the generated adversarial perturbation so that the adversarial examples are kept in the ϵ neighborhood of the benign example.

D. Transferability-based Black-box Attack

In black-box attacks, the adversary can only obtain the model output for an input. Black-box attacks are generally more challenging than white-box attacks because of less knowledge acquired, but they are more practical.

Transferability-based black-box attacks leverage the transferability of adversarial examples between models to attack the target model [32], i.e., for a certain task, adversarial examples generated using one model can confuse other models with high probability, regardless of the differences of model architectures and parameters.

As in Zhang and Wu's work [12], first, a substitute model is trained utilizing generated data labeled with the target model's outputs; then, adversarial examples crafted using the substitute model via white-box attack strategies are used to attack the target model. We used FGSM in black-box attacks, as in [12].

III. Adversarial Detection

Although BCIs are vulnerable to adversarial attacks, adversarial detection in BCIs has not been explored yet.

A. Bayesian Uncertainty

Adversarial detection based on kernel density and Bayesian uncertainty (BU) has been shown as the most robust against optimization-based white-box attacks among ten different adversarial defense approaches [28], including adversarial retraining and input transformation.

A deep neural network trained with dropout is an approximation of the deep Gaussian process [33], a type of Bayesian model, whose uncertain output can be used to estimate low-confidence regions of the input space where adversarial examples may lie. Given a set of functions $\mathcal{G}$ that can map the input space to the output space, the prediction of each function $g \in \mathcal{G}$ is computed, and in a Gaussian process, the variance of the output values can be used as an indicator of the model's uncertainty [25].

For a test example X and its stochastic prediction scores $\{\mathbf{s}_1, ..., \mathbf{s}_T\}$ generated by a neural network with dropout enabled, the BU of the model can be computed as [25]:

$$\mathrm{BU}(X) = \frac{1}{T}\sum_{i=1}^{T} {\mathbf{s}_i}^{\top}\mathbf{s}_i - \left(\frac{1}{T}\sum_{i=1}^{T}\mathbf{s}_i\right)^{\top}\left(\frac{1}{T}\sum_{i=1}^{T}\mathbf{s}_i\right). \quad (9)$$

Feinman *et al.* [25] claimed that the BU for an adversarial example X^{adv} falling near (but not on) the benign data manifold is high. Research has also found that models can be very confident about some adversarial examples [7], which Feinman *et al.* [25] assumed to lie far away from the benign data manifold and can be detected via distance-based metrics.

B. Local Intrinsic Dimensionality

To better understand the properties of adversarial examples, Ma *et al.* [26] proposed to use local intrinsic dimensionality (LID) to characterize adversarial regions. LID evaluates the space-filling ability of the region surrounding a reference example via the distance distribution of its neighbouring examples, which can reflect the intrinsic data dimensionality [34]. They argued that the LID of an adversarial example should be far higher than that of a benign example.

Given an example $X \in \mathcal{X}$, its LID is calculated as [26]:

$$\mathrm{LID}(X) = -\left(\frac{1}{k}\sum_{i=1}^{k}\log\frac{r_i(X)}{r_k(X)}\right)^{-1}, \quad (10)$$

where $r_i(X)$ denotes the distance between X and its ith nearest neighbor among a batch of samples drawn from $\mathcal{X}$, and k is the number of neighbors taken into account. To perform more efficient calculations, Ma *et al.* [26] calculated LID from a randomly selected batch of examples from the dataset. While a larger batch of data can often lead to a more precise estimation of LID, they showed that discrimination between adversarial and benign examples can be achieved using a minibatch size as small as 100 and k as small as 20.

C. Mahalanobis Distance-based Confidence Score

Lee *et al.* [27] proposed a Mahalanobis distance-based metric for the detection of adversarial examples. Given pre-trained features of a softmax neural classifier $f(X)$, they computed the empirical class mean $\widehat{\boldsymbol{\mu}}_c$ and covariance $\widehat{\boldsymbol{\sigma}}$ of training samples $\{(X_1, y_1), ..., (X_N, y_N)\}$ as:

$$\widehat{\boldsymbol{\mu}}_c = \frac{1}{N_c}\sum_{i:y_i=c} f(X_i), \quad (11)$$

$$\widehat{\boldsymbol{\sigma}} = \frac{1}{N}\sum_{c}\sum_{i:y_i=c}(f(X_i) - \widehat{\boldsymbol{\mu}}_c)(f(X_i) - \widehat{\boldsymbol{\mu}}_c)^T, \quad (12)$$

where N_c is the number of samples belonging to Class c. Using the above induced class mean $\widehat{\boldsymbol{\mu}}_c$ and covariance $\widehat{\boldsymbol{\sigma}}$, they defined the Mahalanobis distance-based confidence score $\mathrm{MD}_{\max}(X)$:

$$\mathrm{MD}_{\max}(X) = \max_{c} -(f(X) - \widehat{\boldsymbol{\mu}}_c)^T \widehat{\boldsymbol{\sigma}}^{-1}(f(X) - \widehat{\boldsymbol{\mu}}_c). \quad (13)$$

Lee *et al.* [27] further proposed an input calibration technique to better distinguish adversarial examples from benign ones. For each test sample X, they calculated the calibrated $\widehat{X}$ by adding small noise to X:

$$\begin{aligned}\widehat{X} &= X + \varepsilon \cdot \mathrm{sign}(\nabla_X \mathrm{MD}_{\max}(X)) && (14)\\ &= X - \varepsilon \cdot \mathrm{sign}((f(X) - \widehat{\boldsymbol{\mu}}_c)^T \widehat{\boldsymbol{\sigma}}^{-1}(f(X) - \widehat{\boldsymbol{\mu}}_c)), && (15)\end{aligned}$$

where c is the closest class to X under the Mahalanobis distance, which can be identified during the calculation of $\mathrm{MD}_{\max}(X)$. After calibration, $\widehat{X}$ is closer to the class mean vector $\widehat{\boldsymbol{\mu}}_c$ than X based on the Mahalanobis distance.

IV. Experimental Setup

A. Datasets

We conducted experiments using the following two datasets:

1) Feedback error-related negativity (ERN) [35]: The ERN dataset was released in a Kaggle challenge[1] at the 2015 IEEE Neural Engineering Conference. This dataset was collected from 26 subjects for two-class classification (good-feedback and bad-feedback) tasks. It consists of a training set of 16 subjects and a test set of 10 subjects. We only used the training set in our experiments. We downsampled the 56-channel EEG data to 200 Hz and filtered them using a [1,40] Hz band-pass filter. EEG epoches between [0,1.3]s were extracted and z-score standardized for later tasks, and each subject had 340 EEG epochs.
2) Motor imagery (MI) [36]: The MI dataset is Dataset 2A in BCI Competition IV[2]. The dataset was collected from 9 subjects for four-class classification tasks (left hand, right hand, feet and tongue). The 22-channel EEG signals were sampled at 128 Hz and were filtered by our [4,40] Hz band-pass filter. We extracted the data in [0,2]s after each imagination prompt and standardized them using an exponential moving average window with a decay factor of 0.999. Each subject had 144 EEG epochs for every class.

B. Evaluated Models

The following three CNN models were used in our experiments:

1) EEGNet [37]: EEGNet is a compact CNN model custom-made for EEG classification tasks, which contains two convolutional blocks and one classification block. Depthwise and separable convolutions are used in EEGNet instead of traditional ones, which reduces the number of model parameters.
2) DeepCNN [38]: DeepCNN is larger in size than EEGNet. It contains four convolutional blocks and a softmax layer for classification. The first convolutional block is specially designed for EEG inputs and the other three are standard ones.
3) ShallowCNN [38]: Inspired by filter bank common spatial patterns, ShallowCNN is a shallow version of DeepCNN. It has one convolutional block with a larger kernel, a different activation function and a different pooling approach compared with DeepCNN.

C. Evaluation Settings

We conducted leave-one-subject-out cross-validation on each dataset. For a dataset with N subjects, one subject was used as the test set, two subjects were used to train two substitute models respectively if transferability-based black-box attack was performed, or otherwise discarded, and the remaining subjects were used to train the target model. In our transferability-based black-box attack, the first substitute model was used to imitate the substitute model trained by the adversary, and the second to craft adversarial examples for training our adversarial detectors.

All adversarial detectors were trained on the training set with equal numbers of normal and adversarial examples. We extracted features from the last layer of neural networks, which was found the most effective in [26]. In white-box attack detection, we crafted adversarial examples directly on the target model. In black-box attack detection, we trained our adversarial detectors using adversarial examples generated on the second substitute model. Benign examples which can be correctly classified by the target model and whose corresponding adversarial examples can successfully fool the target model were used in the white-box attack experiments and the training of our black-box adversarial example detectors. In the testing phase of black-box adversarial example detection, we used all benign examples that can be correctly classified by the target model, regardless of whether their corresponding adversarial examples were effective or not. Because black-box attacks are generally weaker than white-box attacks, the number of successful adversarial examples was smaller.

We crafted FGSM, PGD, CW and black-box adversarial examples under ℓ_∞ norm constraint of 0.1. We set the perturbation amplitude $\epsilon = 0.1$ in FGSM and PGD, the perturbation step size $\alpha = 0.01$ in PGD, and its number of iterations to 30. In CW attack, we clipped the perturbation value to [-0.1,0.1].

D. Parameter Tuning

The number of neighbors k for LID and the noise magnitude ε for the Mahalanobis distance-based detector were chosen using nested cross-validation within the training dataset, based on the AUC values of the detection ROC curve. For LID, the number of nearest neighbors was tuned using grid search in [50, 90] with mini-batch size 100, as in [26]. For the Mahalanobis method, we tuned ε using an exhaustive grid search in the log-space of [$1E^{-5}$,$1E^{-2}$].

E. Performance Measures

We used both raw classification accuracy (RCA) and balanced classification accuracy (BCA) to evaluate the classification performance of the models. The RCA is the ratio of the number of correctly classified examples to the number of total examples. The BCA is the average of the RCAs of different classes. BCA is necessary because the ERN dataset has significant class imbalance, which is the case of real-world EEG classifications; using RCA alone can be misleading here.

For adversarial detectors, we used the AUC score to evaluate their performance. The AUC score is the area under the receive operating characteristic curve and ranges from 0 to 1. In a binary classification task, using the AUC score to measure the performance of the classifier avoids the error caused by the selection of classification threshold. The closer the AUC value of a classifier is to 1, the better the performance of this classifier.

[1] https://www.kaggle.com/competitions/inria-bci-challenge

[2] https://www.bbci.de/competition/iv/

V. Experimental Results

A. Attack Performance

Knowing the performances and characteristics of various attacks helps understand adversarial examples. Before adversarial detection, we computed the BCAs and RCAs of neural network classifiers on various adversarial examples, and the ℓ_2 norms of adversarial perturbations imposed by various adversarial attacks.

The results of white-box attacks and the corresponding ℓ_2 norms of the adversarial perturbations are shown in Table I. Adversarial examples created by FGSM, PGD and CW attacks were almost all successful in deceiving the classifiers. ℓ_2 norms of the adversarial perturbation imposed by PGD were slightly smaller than those by FGSM, and those by CW attack were significantly smaller.

The rank of ℓ_2 norms (magnitudes of the perturbations) is intuitive. FGSM is a simple one-step approach. PGD uses a smaller step size to iteratively perform the attack. Compared with FGSM, PGD is more sophisticated and therefore better able to find smaller adversarial perturbations. CW performs multiple optimizations to find smaller adversarial perturbations with guaranteed attack success, which largely reduces the magnitude of the imposed adversarial perturbations.

Examples of a P300 EEG epoch before and after adversarial attacks are shown in Fig. 2. All three adversarial perturbations are tiny and difficult to be noticed by human eyes.

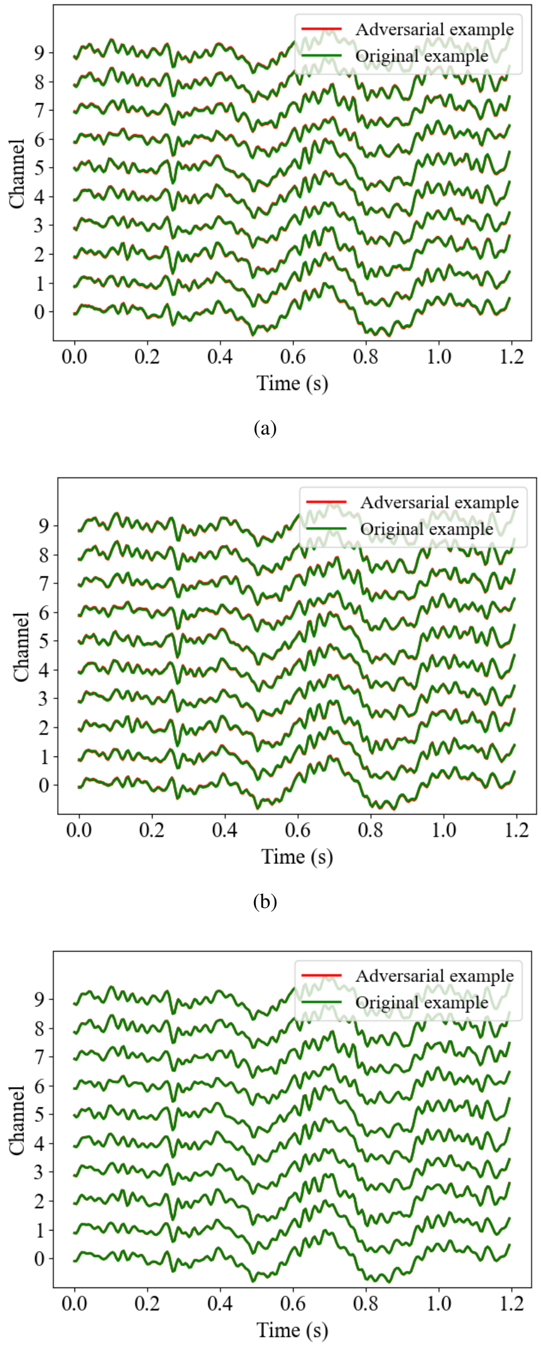
We used EEGNet, DeepCNN and ShallowCNN as the substitute model to craft black-box adversarial examples, respectively. The results of FGSM black-box attacks and the corresponding ℓ_2 norms of the adversarial perturbations are shown in Table II. Black-box attacks can also greatly reduce the classification accuracy of the neural networks. In general, the attacks were more effective when the substitute model and the target model had the same structure. The ℓ_2 norms in Table II are similar to those of FGSM attacks in Table I.

B. Detection of White-box Attacks

Table III shows the discrimination performance (AUC score) of three adversarial detectors (BU, LID, and Mahalanobis distance-based detector) on two EEG datasets and three EEG classifiers. We compared the adversarial detection AUC scores under two gradient-based adversarial attacks (FGSM and PGD) and one optimization-based attack (CW).

Optimization-based white-box attack (CW) was more difficult to detect than gradient-based white-box attacks (FGSM and PGD). Almost all classifiers had significantly lower performance on CW attacks, and the best AUC scores of the eight evaluated adversarial detectors on CW attacks were also lower. The performance differences of these classifiers on gradient-based and optimization-based attacks may reflect, to some extent, the different characteristics of these adversarial examples.

All detectors can distinguish adversarial examples from benign ones to a certain extent, indicating that there are differences between adversarial examples and benign examples. Meanwhile, the performances of the same adversarial detector under different attacks (gradient-based adversarial attack and optimization-based adversarial attack) were different, indicating that there may exist some distribution differences among different adversarial examples. Fig. 3 shows t-SNE visualizations of distributions of a batch of benign and corresponding adversarial MI EEG epochs in the output space of EEGNet. There are clear differences between the distributions of adversarial and benign examples.



(a)

(b)

(c)

Fig. 2. Examples of a P300 EEG epoch before and after adversarial attack. (a) FGSM; (b) PGD; and (c) CW.

TABLE I

RCAs/BCAs OF DIFFERENT CNN CLASSIFIERS UNDER FGSM, PGD AND CW ATTACKS, AND ℓ_2 NORMS OF THEIR CORRESPONDING ADVERSARIAL PERTURBATIONS.

Dataset	Model	RCAs/BCAs				ℓ_2 Norm		
		Baseline	FGSM	PGD	CW	FGSM	PGD	CW
ERN	EEGNet	.6585/.6472	.0001/.0005	.0001/.0005	.0001/.0005	12.07	11.87	1.16
	DeepCNN	.6647/.6359	.0013/.0019	.0001/.0005	.0001/.0005	11.88	11.02	1.31
	ShallowCNN	.6818/.6312	.0001/.0003	.0001/.0003	.0000/.0000	12.00	10.55	1.37
MI	EEGNet	.4515/.4515	.0000/.0000	.0000/.0000	.0000/.0000	7.50	6.75	0.39
	DeepCNN	.4418/.4418	.0001/.0001	.0000/.0000	.0000/.0000	7.45	7.07	0.92
	ShallowCNN	.4511/.4511	.0000/.0000	.0000/.0000	.0041/.0041	7.42	6.79	0.77

TABLE II

RCAs/BCAs OF DIFFERENT CNN CLASSIFIERS UNDER FGSM BLACK-BOX ATTACK, AND ℓ_2 NORMS OF THEIR CORRESPONDING ADVERSARIAL PERTURBATIONS.

Dataset	Target Model	RCAs/BCAs				ℓ_2 norm		
		Baseline	EEGNet	DeepCNN	ShallowCNN	EEGNet	DeepCNN	ShallowCNN
ERN	EEGNet	.6585/.6472	.1352/.1416	.1730/.1794	.5814/.5789	12.07	11.88	12.00
	DeepCNN	.6647/.6359	.2754/.2686	.1767/.1711	.5995/.5701	12.07	11.88	12.00
	ShallowCNN	.6818/.6312	.5479/.5016	.5426/.4897	.3458/.2954	12.07	11.88	12.00
MI	EEGNet	.4515/.4515	.0372/.0372	.2587/.2587	.1707/.1707	7.50	7.45	7.42
	DeepCNN	.4418/.4418	.3568/.3568	.1836/.1836	.1214/.1214	7.50	7.45	7.42
	ShallowCNN	.4511/.4511	.3558/.3558	.2495/.2495	.0774/.0774	7.50	7.45	7.42

TABLE III

AUC SCORES (%) FOR DIFFERENT ADVERSARIAL DETECTION APPROACHES UNDER WHITE-BOX ATTACKS.

Attack	Dataset	Model	BU	LID	$\text{MD}_{\max}$
FGSM	ERN	EEGNet	91.75	**96.49**	93.62
		DeepCNN	**92.93**	89.49	71.08
		ShallowCNN	**99.72**	92.62	67.05
	MI	EEGNet	67.49	**92.02**	92.00
		DeepCNN	72.58	**77.30**	63.73
		ShallowCNN	61.02	**81.90**	70.89
PGD	ERN	EEGNet	94.03	**96.92**	94.38
		DeepCNN	**97.74**	89.65	72.37
		ShallowCNN	**99.81**	92.64	71.01
	MI	EEGNet	65.80	95.22	**96.88**
		DeepCNN	**88.08**	55.53	74.78
		ShallowCNN	**91.78**	87.97	87.73
CW	ERN	EEGNet	70.40	82.17	**85.11**
		DeepCNN	77.48	90.11	**90.21**
		ShallowCNN	74.85	90.96	**91.99**
	MI	EEGNet	55.76	**71.73**	69.36
		DeepCNN	63.08	79.80	**81.82**
		ShallowCNN	58.22	77.92	**80.08**

TABLE IV

AUC SCORES (%) FOR DIFFERENT ADVERSARIAL DETECTION APPROACHES UNDER BLACK-BOX ATTACKS.

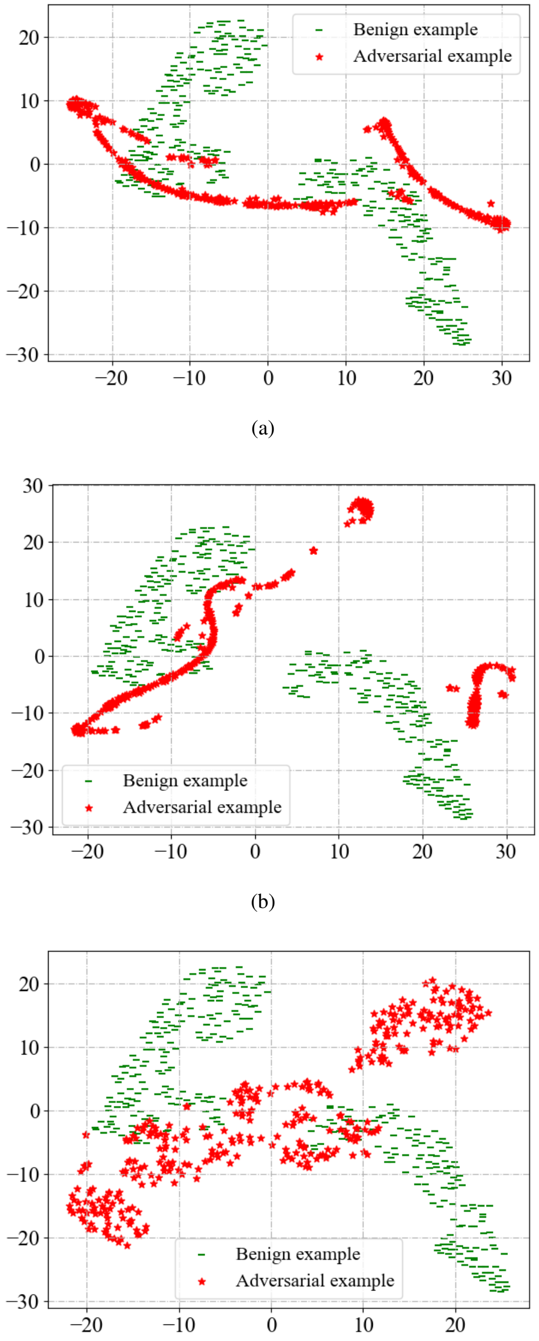
Dataset	Target Model	Substitute	BU	LID	$\text{MD}_{\max}$
ERN	EEGNet	EEGNet	61.10	**77.08**	50.55
		DeepCNN	59.78	**76.11**	51.53
		ShallowCNN	51.58	**59.05**	51.08
	DeepCNN	EEGNet	48.27	**76.01**	49.90
		DeepCNN	53.91	**78.68**	51.40
		ShallowCNN	48.65	**57.26**	51.19
	ShallowCNN	EEGNet	56.86	**63.22**	52.71
		DeepCNN	55.39	**63.14**	51.45
		ShallowCNN	56.54	**71.23**	48.48
MI	EEGNet	EEGNet	71.47	**86.59**	80.72
		DeepCNN	65.02	**69.45**	53.81
		ShallowCNN	56.30	**71.00**	53.34
	DeepCNN	EEGNet	54.18	**61.75**	54.67
		DeepCNN	56.30	**67.16**	53.33
		ShallowCNN	56.36	**70.83**	54.78
	ShallowCNN	EEGNet	46.02	**64.93**	54.08
		DeepCNN	47.40	**70.12**	54.68
		ShallowCNN	48.04	**74.64**	55.64

C. Detection of Black-box Attacks

Table IV shows the AUC scores of various detectors under black-box attacks. Compared with white-box attacks, the detection of black-box attacks was more challenging. It should be noted that the detection performance degradation of black-box attacks may not be attributed to the detector itself alone, but may also be because that the adversarial examples generated by black-box attacks were themselves less aggressive and thus their adversarial artifacts were less obvious.

LID consistently achieved the best performance, may be because:

1) Unlike the MD feature, whose calculations rely on estimated statistics from the training set, the calculation of the LID of the examples in the test set does not depend on the training set. Since the training and test sets came from different subjects, the LID may be computed more accurately.
2) LID is an estimate of the data dimensionality and is not sensitive to the relative position of the class distribution. Although the distribution of adversarial examples generated under black-box attacks may deviate less from the benign example manifold than that of the adversarial examples generated under white-box attacks, the intrinsic dimensionality of black-box adversarial examples is still higher than that of the benign examples.



(a)

(b)

(c)

Fig. 3. t-SNE visualization of benign and corresponding adversarial MI EEG epochs in the output space of EEGNet. (a) FGSM; (b) PGD; and, (c) CW.

TABLE V
GENERALIZATION OF ADVERSARIAL DETECTION FROM FGSM ATTACK TO UNKNOWN ATTACKS. THE LR CLASSIFIER WAS TRAINED ON THE FEATURES EXTRACTED AFTER APPLYING FGSM ATTACK, AND THEN EVALUATED ON PGD AND CW.

Test	Dataset	Model	BU	LID	$\text{MD}_{\max}$
PGD	ERN	EEGNet	94.28	**96.91**	94.25
		DeepCNN	**98.40**	89.76	72.65
		ShallowCNN	**99.88**	92.64	70.92
	MI	EEGNet	65.45	94.93	**96.85**
		DeepCNN	**91.97**	78.25	73.82
		ShallowCNN	75.48	**87.97**	87.56
CW	ERN	EEGNet	30.77	79.25	**83.52**
		DeepCNN	22.51	**90.36**	89.34
		ShallowCNN	25.22	90.92	**91.07**
	MI	EEGNet	55.51	64.58	**69.20**
		DeepCNN	36.79	70.56	**81.11**
		ShallowCNN	45.50	70.86	**79.76**

TABLE VI
GENERALIZATION OF ADVERSARIAL DETECTION FROM WHITE-BOX FGSM ATTACK TO BLACK-BOX ATTACKS. THE LR CLASSIFIER WAS TRAINED ON THE FEATURES EXTRACTED AFTER APPLYING WHITE-BOX FGSM ATTACK, AND THEN EVALUATED ON BLACK-BOX ATTACKS.

Dataset	Target Model	Substitute	BU	LID	$\text{MD}_{\max}$
ERN	EEGNet	EEGNet	39.26	**75.48**	52.19
		DeepCNN	39.25	**75.52**	51.61
		ShallowCNN	47.33	**61.91**	52.42
	DeepCNN	EEGNet	41.61	**76.07**	50.77
		DeepCNN	48.23	**78.64**	50.55
		ShallowCNN	45.29	**57.13**	52.24
	ShallowCNN	EEGNet	41.45	**62.79**	55.51
		DeepCNN	42.66	**62.77**	54.40
		ShallowCNN	41.72	**70.95**	52.82
MI	EEGNet	EEGNet	71.43	**86.92**	77.29
		DeepCNN	65.26	**68.78**	53.51
		ShallowCNN	56.41	**70.67**	53.61
	DeepCNN	EEGNet	45.65	**59.45**	56.52
		DeepCNN	44.99	**65.57**	55.23
		ShallowCNN	44.17	**69.45**	56.70
	ShallowCNN	EEGNet	51.63	**62.40**	54.62
		DeepCNN	51.44	**67.26**	56.21
		ShallowCNN	49.99	**73.57**	54.53

D. Generalization to Other Attacks

To evaluate how well the adversarial detection approaches perform under unknown attacks, we trained adversarial detectors on FGSM adversarial examples, and then evaluated them on PGD and CW. The results are shown in Table V. Adversarial detectors trained on FGSM still retained good detection performances on PGD, but the classification performances of many detectors on CW were greatly reduced. The performance differences of various detectors on PGD and CW may indicate that adversarial examples crafted by FGSM and PGD, two gradient-based attacks, have very similar properties, whereas adversarial examples crafted by FGSM and CW, one gradient-based attack and one optimization-based attack, are very different.

We also tried to detect FGSM adversarial examples generated under our black-box attack scenario by directly using the adversarial detectors trained using adversarial examples crafted by white-box FGSM attacks. The results are shown in Table VI. Compared with the performances in Table IV, the AUC scores of many adversarial detectors were significantly lower (below 50%), indicating that the distribution of adversarial examples generated in white-box and black-box attacks were quite different, even though both were based on FGSM.

E. Discussions

Our detection results under white-box attacks indicated that, when proactive adversarial defense approaches are adopted, there is no need to require the model to fully defend against white-box attacks with strong attack strength. They can be easily recognized by adversarial detection approaches. Furthermore, requiring the model to defend against adversarial examples in adversarial training degrades its performance on benign examples [24].

Adversarial examples generated under black-box attacks are harder to detect than those generated under white-box attacks, but their attack strength is also weaker. Black-box attacks may be better coped with by proactive defense approaches.

For unknown attacks, almost all adversarial detectors have good generalization performance under the same type of attacks (from FGSM to PGD). When the type of adversarial attacks changes, the effectiveness of the adversarial detector may be significantly reduced. This suggests that adversarial examples generated by the same type of adversarial attacks have high similarity, whereas those by different types of adversarial attacks are quite different.

VI. Conclusions and Future Research

This paper extended several adversarial detection approaches from computer vision to EEG-based BCIs. We compared their effectiveness under four attacks, including two gradient-based white-box attacks (FGSM and PGD), an optimization-based white-box attack (CW), and a transferability-based black-box attack, on two EEG datasets using three CNN classifiers. We showed that both white-box and black-box attacks can be detected, and the former are easier to detect.

Our future research will:

1) Explore more metrics for adversarial detection, which describe adversarial examples from more perspectives and may help better understand them.
2) Design feature combinations for adversarial detection. Different features provide different perspectives in adversarial detection. Some may be complementarily, whose combination could improve the adversarial detection performance.
3) Experiment adversarial detections on more adversarial attacks. The properties of adversarial examples generated by different adversarial attacks are usually different. Testing a wider variety of adversarial examples helps better understand adversarial attacks and the vulnerability of the classifiers.
4) Combine proactive adversarial defense approaches with reactive ones. Reactive approaches are better at defending against large perturbations, whereas proactive approaches are better at defending against small ones. There integration may provide more comprehensive defense.

Acknowledgement

This research was supported by the IEEE Computational Intelligence Society Graduate Student Research Grant.

References

[1] M. Ienca, P. Haselager, and E. J. Emanuel, “Brain leaks and consumer neurotechnology,” *Nature Biotechnology*, vol. 36, no. 9, pp. 805–810, 2018.

[2] L. F. Nicolas-Alonso and J. Gomez-Gil, “Brain computer interfaces, a review,” *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, Jun. 2016, pp. 770–778.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics*, Minneapolis, Minnesota, Jun. 2018, pp. 4171–4186.

[5] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” in *Proc. of the British Machine Vision Conf.*, Swansea, UK, Sep. 2015, pp. 1–12.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” in *Proc. Int’l Conf. on Learning Representations*, Banff, Canada, Apr. 2014, pp. 1–10.

[7] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in *Proc. Int’l Conf. on Learning Representations*, SanDiego, CA, May 2015, pp. 1–11.

[8] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, “Adversarial patch,” in *Proc. Int’l Conf. on Neural Information Processing Systems*, LongBeach, CA, Dec. 2017.

[9] K. Grosse, N. Papernot, P. Manoharan, M. Backes, and P. McDaniel, “Adversarial perturbations against deep neural networks for malware classification,” *arXiv preprint arXiv:1606.04435*, 2016.

[10] N. Carlini and D. Wagner, “Audio adversarial examples: Targeted attacks on speech-to-text,” in *Proc. IEEE Security and Privacy Workshops (SPW)*, San Francisco, CA, May 2018, pp. 1–7.

[11] A. Bar, J. Lohdefink, N. Kapoor, S. J. Varghese, F. Huger, P. Schlicht, and T. Fingscheidt, “The vulnerability of semantic segmentation networks to adversarial attacks in autonomous driving: Enhancing extensive environment sensing,” *IEEE Signal Processing Magazine*, vol. 38, no. 1, pp. 42–52, 2020.

[12] X. Zhang and D. Wu, “On the vulnerability of CNN classifiers in EEG-based BCIs,” *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 5, pp. 814–825, 2019.

[13] D. Wu, W. Fang, Y. Zhang, L. Yang, X. Xu, H. Luo, and X. Yu, “Adversarial attacks and defenses in physiological computing: A systematic review,” *National Science Open*, 2021, in press.

[14] L. Meng, C.-T. Lin, T.-P. Jung, and D. Wu, “White-box target attack for EEG-based BCI regression problems,” in *Proc. Int’l Conf. on Neural Information Processing*, Sydney, Australia, Dec. 2019, pp. 476–488.

[15] X. Zhang, D. Wu, L. Ding, H. Luo, C.-T. Lin, T.-P. Jung, and R. Chavarriaga, “Tiny noise, big mistakes: Adversarial perturbations induce errors in brain–computer interface spellers,” *National Science Review*, vol. 8, no. 4, p. nwaa233, 2020.

[16] Z. Liu, L. Meng, X. Zhang, W. Fang, and D. Wu, “Universal adversarial perturbations for CNN classifiers in EEG-based BCIs,” *Journal of Neural Engineering*, vol. 18, no. 4, p. 0460a4, 2021.

[17] R. Bian, L. Meng, and D. Wu, “SSVEP-based brain-computer interfaces are vulnerable to square wave attacks,” *Science China Information Sciences*, vol. 65, no. 4, pp. 1–13, 2022.

[18] X. Jiang, L. Meng, S. Li, and D. Wu, “Active poisoning: Efficient backdoor attacks to transfer learning based BCIs,” *Science China Information Sciences*, 2022, in press.

[19] Y. Li, J. Pan, J. Long, T. Yu, F. Wang, Z. Yu, and W. Wu, “Multimodal BCIs: Target detection, multidimensional control, and awareness evaluation in patients with disorder of consciousness,” *Proc. of the IEEE*, vol. 104, no. 2, pp. 332–352, 2015.

[20] D. Wu, V. J. Lawhern, S. Gordon, B. J. Lance, and C.-T. Lin, “Driver drowsiness estimation from EEG signals using online weighted adaptation regularization for regression (OwARR),” *IEEE Trans. on Fuzzy Systems*, vol. 25, no. 6, pp. 1522–1535, 2016.

[21] A. Binnendijk, T. Marler, and E. M. Bartels, *Brain-computer-interfaces: U.S. military applications and implications, an initial assessment*. Santa Monica, CA: RAND Corporation, 2020.

[22] A. Aldahdooh, W. Hamidouche, S. A. Fezza, and O. Déforges, “Adversarial example detection for DNN models: A review and experimental comparison,” *Artificial Intelligence Review*, vol. 55, no. 6, pp. 4403–4462, 2022.

[23] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, “Theoretically principled trade-off between robustness and accuracy,”

in *Proc. Int'l Conf. on Machine Learning*, Long Beach, CA, Jun. 2019, pp. 7472–7482.

[24] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int'l Conf. on Learning Representations*, Vancouver, Canada, Apr. 2018, pp. 1–28.

[25] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," *arXiv preprint arXiv:1703.00410*, 2017.

[26] X. Ma, B. Li, Y. Wang, S. M. Erfani, S. Wijewickrema, G. Schoenebeck, D. Song, M. E. Houle, and J. Bailey, "Characterizing adversarial subspaces using local intrinsic dimensionality," *arXiv preprint arXiv:1801.02613*, 2018.

[27] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Proc. Advances in Neural Information Processing Systems*, Montreal, Canada, Dec. 2018.

[28] N. Carlini and D. Wagner, "Adversarial examples are not easily detected: Bypassing ten detection methods," in *Proc. of the 10th ACM Workshop on Artificial Intelligence and Security*, Dallas, TX, Nov. 2017, pp. 3–14.

[29] Z. Deng, X. Yang, S. Xu, H. Su, and J. Zhu, "Libre: A practical bayesian approach to adversarial detection," in *Proc. Int'l Conf. on Computer Vision and Pattern Recognition*, Nashville, TN, Jun. 2021, pp. 972–982.

[30] C. Zhang, Z. Ye, Y. Wang, and Z. Yang, "Detecting adversarial perturbations with saliency," in *Proc. 3rd IEEE Int'l Conf. on Signal and Image Processing (ICSIP)*, Shenzhen, China, Jul. 2018, pp. 271–275.

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

[32] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against deep learning systems using adversarial examples," in *Proc. ACM Asia Conf. on Computer and Communications Security*, Abu Dhabi, United Arab Emirates, Apr. 2017, p. 3.

[33] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int'l Conf. on Machine Learning*, New York City, NY, Jun. 2016, pp. 1050–1059.

[34] M. E. Houle, "Local intrinsic dimensionality I: An extreme-value-theoretic foundation for similarity applications," in *Proc. Int'l Conf. on Similarity Search and Applications*, Munich, Germany, Oct. 2017, pp. 64–79.

[35] P. Margaux, M. Emmanuel, D. Sébastien, B. Olivier, and M. Jérémie, "Objective and subjective evaluation of online error correction during P300-based spelling," *Advances in Human-Computer Interaction*, vol. 2012, no. 6, p. 4, 2012.

[36] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. Mueller-Putz *et al.*, "Review of the BCI competition IV," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012.

[37] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, 2018.

[38] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.