Patient-Specific Seizure Prediction via Adder Network and Supervised Contrastive Learning

Yuchang Zhao, Chang Li, *Member, IEEE*, Xiang Liu, Ruobing Qian, Rencheng Song, *Member, IEEE*, and Xun Chen, *Senior Member, IEEE*

***Abstract*— Deep learning (DL) methods have been widely used in the field of seizure prediction from electroencephalogram (EEG) in recent years. However, DL methods usually have numerous multiplication operations resulting in high computational complexity. In addtion, most of the current approaches in this field focus on designing models with special architectures to learn representations, ignoring the use of intrinsic patterns in the data. In this study, we propose a simple and effective end-to-end adder network and supervised contrastive learning (AddNet-SCL). The method uses addition instead of the massive multiplication in the convolution process to reduce the computational cost. Besides, contrastive learning is employed to effectively use label information, points of the same class are clustered together in the projection space, and points of different class are pushed apart at the same time. Moreover, the proposed model is trained by combining the supervised contrastive loss from the projection layer and the cross-entropy loss from the classification layer. Since the adder networks uses the ℓ_1-norm distance as the similarity measure between the input feature and the filters, the gradient function of the network changes, an adaptive learning rate strategy is employed to ensure the convergence of AddNet-SCL. Experimental results show that the proposed method achieves 94.9% sensitivity, an area under curve (AUC) of 94.2%, and a false positive rate of (FPR) 0.077/h on 19 patients in the CHB-MIT database and 89.1% sensitivity, an AUC of 83.1%, and an FPR of 0.120/h in the Kaggle database. Competitive results show that this method has broad prospects in clinical practice.**

***Index Terms*— Deep learning, seizure prediction, electroencephalogram (EEG), adder network, contrastive learning, adaptive learning rate.**

Manuscript received January 1, 2022; revised May 5, 2022; accepted May 31, 2022. Date of publication June 3, 2022; date of current version June 10, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFF1200600; in part by the National Natural Science Foundation of China under Grant 61922075, Grant 41901350, and Grant 32150017; in part by the Provincial Natural Science Foundation of Anhui under Grant 2008085QF285; in part by the Anhui Key Project of Research and Development Plan Grant 202104d07020005; and in part by the Fundamental Research Funds for the Central Universities under Grant JZ2021HGTB0078 and Grant PA2021KCPY0051. *(Corresponding author: Chang Li.)*

Yuchang Zhao, Chang Li, and Rencheng Song are with the Department of Biomedical Engineering, Hefei University of Technology, Hefei 230009, China, and also with the Anhui Province Key Laboratory of Measuring Theory and Precision Instrument, School of Instrument Science and Optoelectronics Engineering, Hefei University of Technology, Hefei 230009, Anhui, China (e-mail: yuchangzhao@mail.hfut.edu.cn; changli@hfut.edu.cn; rcsong@hfut.edu.cn).

Xiang Liu and Ruobing Qian are with the Epilepsy Centre, Department of Neurosurgery, The First Affiliated Hospital of USTC, Division of Life Sciences and Medicine, University of Science and Technology of China, Hefei, Anhui 230001, China (e-mail: ahslyyliuxiang@163.com; qianruobing@fsyy.ustc.edu.cn).

Xun Chen is with the Epilepsy Centre, Department of Neurosurgery, The First Affiliated Hospital of USTC, Division of Life Sciences and Medicine, University of Science and Technology of China, Hefei, Anhui 230001, China, and also with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230001, China (e-mail: xunchen@ustc.edu.cn).

Digital Object Identifier 10.1109/TNSRE.2022.3180155

I. Introduction

EPILEPSY is one of the most common non-communicable diseases of the nervous system. It is caused by discharges of brain neurons. There are about 70 million people with epilepsy in the world, and about one-third of them cannot be controlled by drugs since they are resistant to anti-epileptic drugs [1], [2]. Generally speaking, intervention and protection before the arrival of epilepsy activities can greatly reduce the suffering of patients and have a positive effect on treatment.

Electroencephalogram (EEG) is a device that records the electrical activity of neurons in the brain cortex and contains various information related to brain electrical functions [3]. Studies have shown that there are EEG signals of preictal activity before the onset of epilepsy, which are the key information for predicting seizure [4], [5]. In traditional machine learning methods, a series of manually extracted linear and nonlinear features were used to predict seizures, such as absolute and relative spectral band power [6], Kolmogorov entropy [7]. Then, the classifier distinguished between preictal and interictal states by these features, SVM [8], [9] and k-nearest neighbor classifier [10] were widely used in seizure prediction. However, massive expert experience and professional knowledge are required to extract features manually. The performance of classification depends on the discrimination of the designed features and the learning ability of the classifier. These algorithms usually generalize poorly to new patients and data [11].

With the advent of Graphics Processing Units (GPUs) and growth of computing power, the speed of multiplication has been greatly improved. Deep learning (DL) methods with more floating number multiplications could be implemented, which are widely used in the field of seizure prediction and make important progress [5], [12]–[16]. However, these methods with billions of calculations can bring high energy consumption and need to be deployed on high-end GPUs. Recently, several energy-efficient methods have been used for seizure prediction. Zhao *et al.* [17] binarized all parameters except the first layer, thereby reducing the calculation and storage. Four models were proposed in [14], the best of which has only 18345 trainable parameters. Quantization and pruning were employed to compress the network in [18]. Although operations such as binarization and pruning significantly reduce the computational cost, they tend to reduce the original performance. Moreover, the binary network is more difficult to train and usually requires a smaller learning rate and slower convergence rate [19].

In addition, seizure prediction distinguishes preictal and interictal signals, which are different from the data during ictal. Usually, these two types of data contain some hard samples, which cannot be distinguished well by the cross-entropy loss. Most seizure prediction techniques used the cross-entropy loss as supervised classification loss function. The model is trained by projecting the samples into the feature space, then the features go through the classifier to output the probability distribution, and finally the network parameters are updated by reducing the difference between the outputs and the one-hot encoded labels. Many works have explored the drawbacks of this loss, for example, the cross-entropy loss may lead to poor margins between different classes [20], lack of robustness to noisy labels [21], and cause generalization performance degradation. Several hard positive and negative samples may be generated when defining preictal and interictal, resulting in poor classification margins.

In this work, we propose a simple and effective end-to-end adder network and supervised contrastive learning (AddNet-SCL), On the one hand, the adder network (AddNet) uses cheap addition instead of multiplication to reduce computational complexity. Since addition and subtraction can be converted to each other through complements, the ℓ_1 distance with only addition and subtraction is used instead of the cosine similarity of multiplication in traditional convolution, which is a hardware-friendly similarity measure. Importantly, it can speed up the network without losing accuracy. In order to ensure better training and convergence of the network, a back-propagation scheme suitable for addition convolution is also used. On the other hand, supervised constrastive learning is used to make full use of the label information and the intrinsic pattern of the data, clustering samples of the same class together while pushing them away from samples of different classes. The contrastive loss provides an intrinsic mechanism for hard positive and negative mining, which can better learn the potential information of difficult samples, and better optimize the classification margin. Then, the representations in the projection space are further separated by the cross-entropy loss. By using a hybrid function of contrastive loss and the cross-entropy loss to train the network, better seizure prediction performance can be obtained than using the cross-entropy or contrastive loss alone. Finally, the input of the network is the original EEG signal without any feature preprocessing. Automatically extracting features from data is more beneficial to DL methods and more in line with data-driven principles.

Experimental was conducted on two public databases the Boston Children's Hospital (CHB)-MIT [22] and The American Epilepsy Society Seizure Prediction Challenge (Kaggle) [23]. The experimental results show that the proposed method can not only reduce the computational cost better, but also obtain competitive performance. Our main contributions are as follows:

1) We propose a new framework AddNet-SCL for seizure prediction, using addition instead of multiplication to reduce computational cost, while using a new back propagation scheme and adaptive learning rate to ensure the convergence of the network. To the best of our knowledge, this is first time to use adder network for seizure prediction.
2) We use a loss function that is a mixture of supervised contrastive loss and the cross-entropy loss. The supervised contrastive loss is used in the projection space to separate the samples, and the cross-entropy loss is used on the classification layer to further map the representation to the corresponding labels. This study is the first to report supervised contrastive learning used for seizure prediction.
3) The experimental results of the proposed method are significantly better than the baseline model, and the energy consumption and latency are much lower than the baseline model. The method has no feature preprocessing and special structure design, achieves 94.2% sensitivity, an area under curve (AUC) of 94.9%, and a false positive rate of (FPR) 0.077/h on 19 patients in the CHB-MIT database and 89.1% sensitivity, an AUC of 0.831, and an FPR of 0.120/h in the Kaggle database.

The rest of paper is composed as follows. In Section II, we present the databases used in this paper and the details of the proposed method. In Section III, we show the experimental results of this method. In Section IV, we discuss the experimental results and compare them with related work. Finally, summary of this work in Section V.

II. Databases and Methods

A. Databases

The databases used in this work include CHB-MIT [22] and Kaggle [23]. The CHB-MIT database contains the scalp EEG (sEEG) data of 23 children, with a total of 844 hours continuous EEG recordings and 182 seizure events, each seizure event is annotated by clinical experts. These EEG signals are collected using 22 electrodes with 256 Hz sampling rate. Kaggle database includes intracranial EEG (iEEG) data from five dogs and two patients, with a total of 627.7 hours interictal records and 48 seizure events. Dogs 1-4 iEEG data were collected though 16 implanted electrodes at a sampling rate of 400 Hz, and the number of electrodes for Dog 5 were 15. The iEEG records of the two patients were collected from 15 depth electrodes (Patient 1) and 24 subdural electrodes (Patient 2) at 5000 Hz sampling rates, respectively.

B. Preprocessing

In seizure prediction task, data preprocessing could have a large effect on the results. Most previous works performed

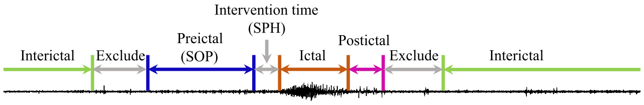


Fig. 1. Interictal, preictal, ictal and postictal states of seizures (from the file *chb*01_03.edf).

TABLE I
DATA DURATION (IN HOUR) OF CHB-MIT DATABASE

Patient	Interictal-Ictal Distance	Preictal Length		No. of Seizures
	240 Minutes	30 Minutes	60 Minutes	
pt1	14.4	3.5	6.1	7
pt2	26.0	1.5	3.0	3
pt3	27.4	3.0	5.6	6
pt5	14.4	2.5	5.0	5
pt6	24.8	5.0	9.5	10
pt8	1.0	2.5	4.7	5
pt9	48.6	2.0	4.0	4
pt10	24.2	3.4	5.8	7
pt11	32.0	0.9	1.1	2
pt13	15.1	3.0	4.7	7
pt14	4.7	3.4	5.6	8
pt16	5.6	4.2	5.6	9
pt17	11.8	1.5	2.5	3
pt18	26.4	2.5	4.0	5
pt19	26.0	1.0	1.6	2
pt20	19.1	3.9	5.3	8
pt21	23.4	2.0	3.6	4
pt22	17.1	1.3	2.3	3
pt23	14.2	3.5	5.5	7
Total	375.9	50.6	85.5	105

feature preprocessing of EEG signals. In order to make the network automatically learn the optimal features, we used the raw data as input. First, we defined interictal period, preictal period, postictal period, seizure prediction horizon (SPH) and seizure occurrence period (SOP). The interictal period was defined as between at least 4 hours before seizure onset and 4 hours after seizure end. The preictal period was generally 15 to 60 minutes before seizure onset [5], [12]–[15]. In this paper, we chose 30 minutes and 60 minutes for the experiment. The 10 minutes after seizure end was defined as the postictal period [5]. SOP is the period when seizure is expected to occur, which is equal to preictal period. The period from the alarm to the start of SOP is SPH [24], as shown in Fig. 1. When an alarm is triggered, there will be a period of time for doctors to intervene in the clinic. Therefore, SPH is also called intervention time [25]. In the CHB-MIT database we set SPH to one minute [5], [16], and in the Kaggle database, SPH is set to five minutes by the organizer. For cases with more than one seizure event, if the preictal period is less than 15 minutes, we combine it with the leading seizure into one seizure. For patients with more than ten seizures, seizure prediction becomes less critical, so we exclude these patients from the data. With these definitions and limitations, 105 seizure events in 19 patients in CHB-MIT database and five dogs in Kaggle database were used to evaluate. The EEG signal duration information of the selected patients is shown in Table I and Table II.

Through the above method, we obtain the interictal and preictal records. Samples are obtained by using moving window analysis on the continuous interictal and preictal records. Notably, seizure prediction task also has data imbalance challenges. For most patients' data, the interictal data is far more than the preictal data, and the unbalanced data could affect final performance. Therefore, in the process of moving window analysis, a four seconds window was employed to perform non-overlapping sampling in the interictal period and half-overlapping sampling in the preictal period during training phase. In the testing phase, non-overlapping sampling was used for both interictal and preictal data, and the continuity of the samples in the temporal dimension is maintained. Finally, in the training phase, if the interictal data in the training set is still more than the preictal data, we randomly discard part of the interictal data so that the ratio of interictal to preictal is equal [5], [12], [13].

TABLE II
DATA DURATION (IN HOUR) OF KAGGLE DATABASE

Participant	Interictal hours	No. of Seizures
Dog1	80	4
Dog2	83.3	7
Dog3	240	12
Dog4	134	14
Dog5	75	5
Total	612.3	42

C. AddNet-SCL

In this paper, a novel network structure AddNet-SCL was used for seizure prediction. To reflect the innovative advantages of additive convolution and supervised contrastive learning, we used CNN as the backbone and added residual connections to the network (ResCNN). We replaced the traditional convolution in the network with additive convolution to get the AddNet, and added supervised contrastive loss in the projection space. The specific structure is shown in Fig. 2.

First, the temporal dimension of the original EEG data is much larger than the channel dimension, and the data did not undergone any denoising and feature preprocessing. Therefore, we used one-dimensional convolution to extract features in the temporal dimension of the data, remove noise information, and reduce the dimensionality of the data. Then, additive convolution was used to perform feature learning on the input features, and combined residual connections to make the network learning representation more stable. The first additive convolution layer contains convolution kernel sizes of 11 × 1 and 3 × 3 with a stride of 1, and then skip connections though 1 × 1 convolution, finally go through a 4×1 maximum pooling layer. The second additive convolution layer consists of convolution size of 5 × 5 with a stride of 2, and a 3 × 3 convolution with a stride of 1, finally a skip connection with 1 × 1 convolution with a stride of 2. The structure of the third additive convolution layer is the same as the second layer. A 64-dimensional feature vector is obtained through adaptive average pooling. Finally, the obtained vector is updated and classified by contrastive loss and the cross-entropy loss. Next, we describe the implementation process of AddNet and supervised contrastive loss in detail.

D. AddNet

It is well known that the complexity, energy consumption, and proportion of the computational unit for multiplication

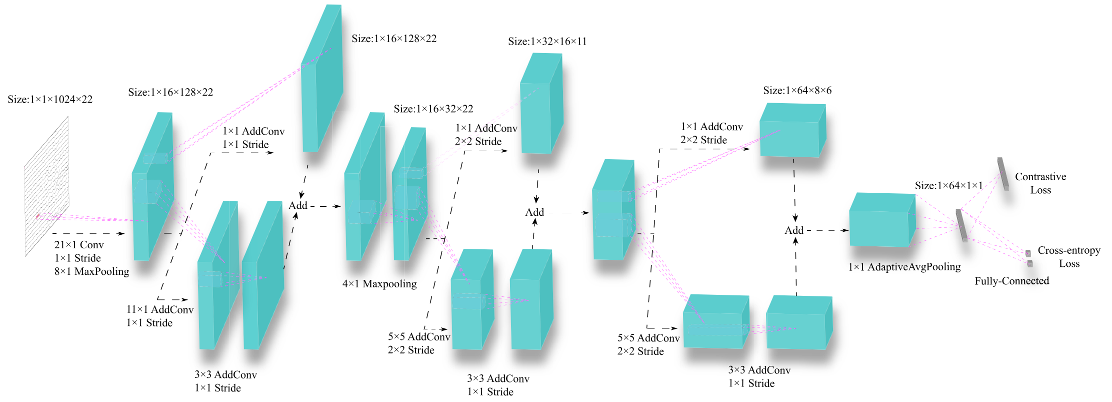
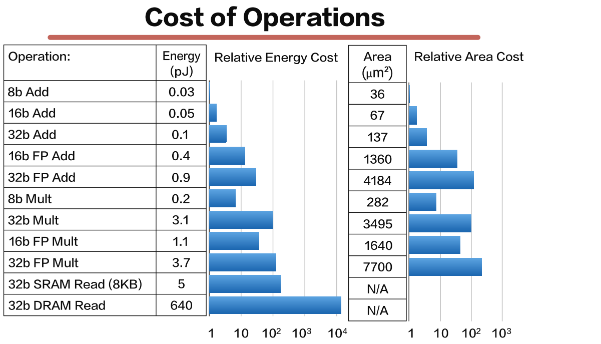


Fig. 2. The structure graph of the proposed AddNet-SCL.



Operation:	Energy (pJ)	Area (μm²)
8b Add	0.03	36
16b Add	0.05	67
32b Add	0.1	137
16b FP Add	0.4	1360
32b FP Add	0.9	4184
8b Mult	0.2	282
32b Mult	3.1	3495
16b FP Mult	1.1	1640
32b FP Mult	3.7	7700
32b SRAM Read (8KB)	5	N/A
32b DRAM Read	640	N/A

Fig. 3. Energy consumption of addition and multiplication, and the area of the calculation cell occupied. The data in the figure comes from [35].

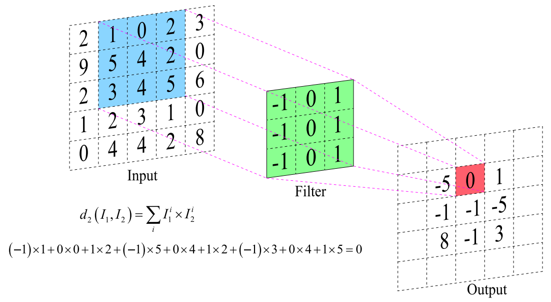

2 1 0 2 3
9 5 4 2 0
2 3 4 5 6
1 2 3 1 0
0 4 4 2 8
Input
-1 0 1
-1 0 1
-1 0 1
Filter
-5 0 1
-1 -1 -5
8 -1 3
Output
$d_2(I_1,I_2)=\sum_i I_1^i \times I_2^i$
$(-1)\times1+0\times0+1\times2+(-1)\times5+0\times4+1\times2+(-1)\times3+0\times4+1\times5=0$

Fig. 4. Typical convolution calculation method.

of floating number are much higher than these of addition, as shown in Fig. 3. To speed up the network and reduce the amount of calculation, researchers have proposed various work based on different principles. The pruning method reduces network complexity by removing redundant weights [26], [27]. Different from the theory of pruning, many works are devoted to designing new convolution modules or operations to replace typical convolution. A series of lightweight network have been proposed, such as SqueezeNet [28], MobileNet [29], ShuffleNet [30], Xception [31], GhostNet [32]. The knowledge distillation scheme [33] obtains a lighter netwrok by transferring useful information from complex teacher networks to a lightweight student network. However, these lightweight models or technologies suffer from substantial multiplication, and still consume massive computational resources. BinaryConnect [34] forces the network weight to be binary, which can make many multiply-accumulate operations become accumulations, thereby reducing computational complexity. However, binarized networks are often hard to train and difficult to preserve accuracy. In this work, we applied the principle of additive convolution to seizure prediction, and used this method to reduce the computational complexity. The detailed introduction is as follows.

Typical convolution measures similarity by multiplying and accumulating between filters and input features, as shown in Fig. 4. Given that $F \in \mathbb{R}^{m\times n\times c_{in}\times c_{out}}$ is a filter in one layer of the network, where kernel size is $m \times n$, c_{in} and c_{out} represent the number of input channels and output channels, respectively. $I \in \mathbb{R}^{H\times W\times c_{in}}$ is the input feature, where H and W denote the height and width of the input feature, respectively. The output feature O is obtained by computing the similarity between the filter and input feature.

$$O(a,b,c)=\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\sum_{k=1}^{c_{in}} S(I(a+i,b+j,k),F(i,j,k,c)), \tag{1}$$

where $S(\cdot,\cdot)$ is the algorithm of similarity measurement, and c represents one of the output channels. When $S(x, y) = x \times y$, Eq. 1 is typical convolution operation. If the kernel size is 1×1, then Eq. 1 can represent the calculation of fully-connected layer. There are other metrics that can express the distance between two vectors, but most of them involve complex multiplications. The ℓ_1 distance calculates the sum of the absolute value of the difference between the two vectors. Therefor, using ℓ_1 distance to measure the similarity, Eq. 1 can be expressed as:

$$O(a,b,c) = -\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\sum_{k=1}^{c_{in}} |I(a+i,b+j,k)-F(i,j,k,c)|, \tag{2}$$

Eq. 2 only contains addition and subtraction, as shown in Fig. 5, and subtraction can easily be simplified to addition. It is worth noting that the output of the addition filter of Eq. 2 is always negative. The output of typical convolution corresponds

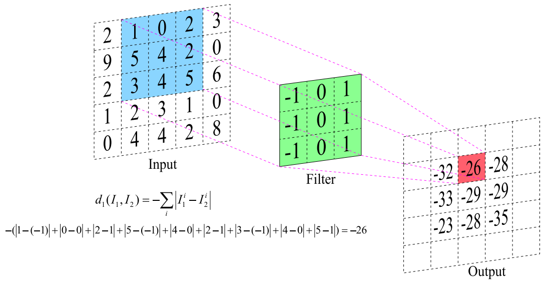


Fig. 5. Additive convolution calculation method.

TABLE III
The ℓ2-Norm of the Gradient of Each Convolutional Layer After the First Iteration of Different Models

Model	layer1	layer2	layer3
AddNet	0.0046	0.0010	0.0006
ResCNN	0.4506	1.0651	1.5955

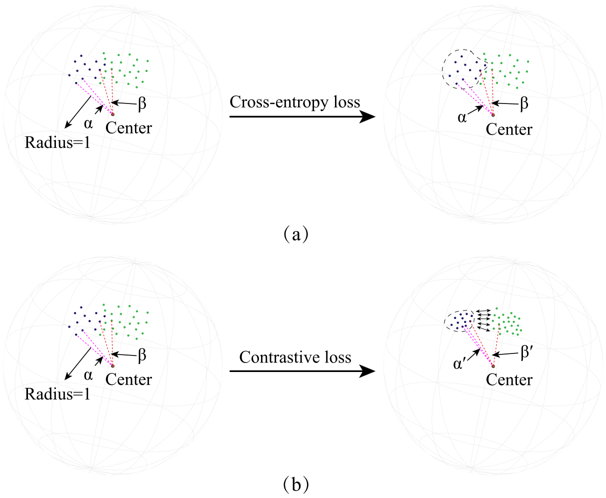
to the weighted sum of the values in the input feature map, and the result can be positive or negative. Therefore, to better use the traditional convolution activation function, batch normalization is used to normalize the output of the addition filter to an appropriate range. Although batch normalization contains multiplications, it can be ignored compared to typical convolution. Assuming that one of the convolution layer has a filter $F \in \mathbb{R}^{m \times n \times c_{in} \times c_{out}}$, an input $I \in \mathbb{R}^{H \times W \times c_{in}}$, and an output $O \in \mathbb{R}^{H' \times W' \times c_{out}}$, the typical convolution and batch normalization multiplication calculation amount are $\mathcal{O}_{conv-mul}(mnc_{in}c_{out}HW)$ and $\mathcal{O}_{BN-mul}(c_{out}H'W')$, respectively. In practice, given $c_{in} = 64$ and kernel size of 5×5, it can be calculated that the complexity of typical convolution is $\frac{mnc_{in}c_{out}HW}{c_{out}H'W'} \approx 1600$ times the complexity of batch normalization. The number of addition calculations for typical convolution and additive convolution is $\mathcal{O}_{conv-add}(mnc_{in}c_{out}HW)$ and $\mathcal{O}_{addconv-add}(2mnc_{in}c_{out}HW)$, respectively, while the number of multiplications for additive convolution is 0. Additive convolution reduces computational cost by converting all multiplications in typical convolutions into additions.

According to the calculation of the partial derivative of the input of the loss, it is clear that the gradient of the filters in AddNet is much smaller than that of ResCNN, which makes the parameter update very slow. We report the gradient of AddNet and ResCNN in the first iteration in Table III. A straightforward way to solve this problem is to use a larger learning rate. Since the large difference in gradient between each layer of the network, we employed a more effective adaptive learning rate. The specific calculation method is as follows:

$$\Delta F_l = \eta \times \theta_l \times \Delta L(F_l), \tag{3}$$

where η is the global learning rate, θ_l is the local learning rate of the l-th layer, and $\Delta L(F_l)$ is the gradient of the filter at l-th layer. The calculation method of θ_l is as follows:

$$\theta_l = \frac{\lambda\sqrt{z}}{||\Delta L(F_l)||_2}, \tag{4}$$



(a)

(b)

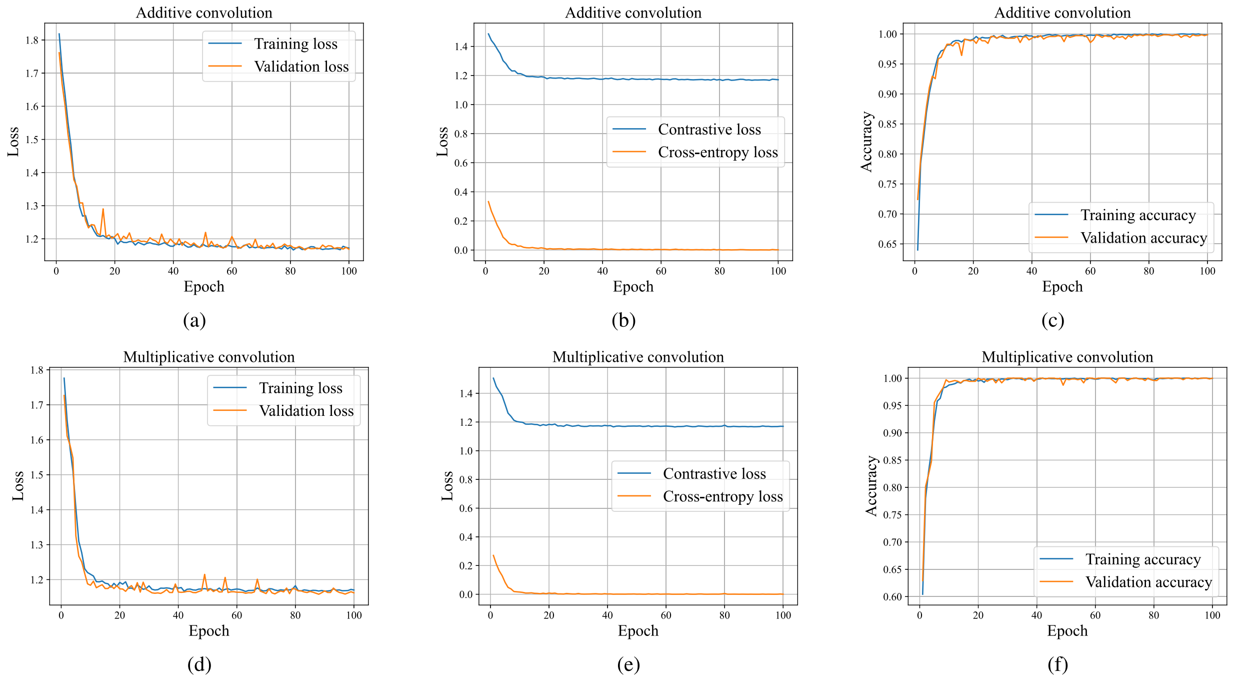
Fig. 6. The effect of the cross-entropy loss and contrastive loss on classification. Assuming that the hypersphere is a sphere with a radius of 1. (a) is the effect of the cross-entropy loss on the classification boundary, (b) is the effect of contrastive loss on the classification boundary.

where λ is the hyperparameter that controls the local learning rate and is set to $\frac{1}{5}$, and z is the number of elements of the filter F_l.

E. Supervised Contrastive Learning

In the previous DL-based seizure prediction methods, most of them used the cross-entropy loss as the loss function. Cross-entropy measures the Kullback-Leibler (KL) divergence between two distributions (label distribution and empirial regression distribution). However, the cross-entropy loss has some shortcomings, such as lack of robustness to noisy labels, and poor classification margins between samples of different classes. Recently, Khosla *et al.* [36] inspired by self-supervised contrastive loss and metric learning, proposed supervised contrastive loss, which completely removes the reference distribution, clusters the embeddings of samples from the same class, and pushes away samples that are different from their own. It achieves better performance than the cross-entropy in the classification task of ImageNet. Nasiri *et al.* [37] used supervised contrastive loss and the cross-entropy loss in environmental sound classification tasks to achieve the state-of-the-art performance. In seizure prediction, since the ictal, interictal, and preictal period are artificially defined, there may be hard samples, and use of the cross-entropy loss may be affected by these samples. In this work, we combine the cross-entropy loss and supervised contrastive loss to further distinguish between interictal and preictal.

Fig. 6 shows simple schematic of the cross-entropy loss and the supervised contrastive loss. The embedding feature vectors in the projection space was normalized so that they all fall on a hypersphere with a radius of 1. As shown in Fig. 6 (a), using the cross-entropy loss to directly classify may produce a poor margin, and even classify some samples incorrectly. In Fig. 6 (b), by using contrastive loss to increase the cosine similarity between samples of the same class, while reducing the cosine similarity between different classes (to make the



(a) (b) (c)

(d) (e) (f)

Fig. 7. Loss convergence and accuracy graph. (a), (b), (c) are the loss curve and accuracy curve of additive convolution in the training phase. (d), (e), (f) are the loss curve and accuracy curve of multiplicative convolution in the training phase.

angle α between the same class smaller, and at the same time to make the angle β between different classes larger). In this way, samples are separated from the projection space, and then the cross-entropy loss classification can be used to obtain a better classification effect.

Given that the input samples $X = \{x_1, x_2, \ldots, x_N\}$, the corresponding labels are $Y = \{y_1, y_2, \ldots, y_N\}$, N represents batch size. The supervised contrastive loss is calculated by the following formula:

$$\mathcal{L}_i^{sup} = -\frac{1}{N_{y_i}}\log\frac{\sum_{j=1}^{N}\mathbb{1}_{[y_j=y_i,j\neq i]}\exp(\boldsymbol{z_i}\cdot\boldsymbol{z_j}/\tau)}{\sum_{k=1}^{N}\mathbb{1}_{[k\neq i]}\exp(\boldsymbol{z_i}\cdot\boldsymbol{z_k}/\tau)}, \tag{5}$$

where $\boldsymbol{z_i}, (i \in \{1, 2, \ldots, N\})$ is the embedding vector of input x_i in the projection space, N_{y_i} denotes those samples that have the same label as y_i in a batch. $\mathbb{1}_{[y_j=y_i,j\neq i]} \in \{0, 1\}$, if $y_j = y_i$ and $j \neq i$, the value is 1, otherwise it is 0. $\boldsymbol{z_i}\cdot\boldsymbol{z_j}$ represents the inner product of $\boldsymbol{z_i}$ and $\boldsymbol{z_j}$. Since $\boldsymbol{z_i}, (i \in \{1, 2, \ldots, N\})$ is normalized, it is also equivalent to calculating the cosine similarity of these vectors. τ is the temperature and set to 0.08, which controls the smoothness of training and the effect of hard samples.

F. Training and Loss

In this study, patient-specific method was used to conduct separate model training for each subject. Convergence was achieved by optimizing the hybrid loss function that combined the supervised contrastive loss and the cross-entropy loss. The hybrid loss function is as follows:

$$\mathcal{L} = \alpha\mathcal{L}^{sup} + (1-\alpha)\mathcal{L}^{cross}, \tag{6}$$

where α is hyperparameter and we set $\alpha = 0.5$, $\mathcal{L}^{cross}$ represents the cross-entropy loss, the calculation formula is as follows:

$$\mathcal{L}^{cross} = -\frac{1}{N}\sum_i [y_i\log(p_i) + (1-y_i)\log(1-p_i)], \tag{7}$$

where y_i is the label and p_i is the probability of the preictal sample.

Although the balance of preictal and interictal data in training stage was maintained, the length of preictal and interictal records for each seizure event is different, which could also lead to overfitting. The early stopping was employed to further reduce the overfitting of the network. When the loss in the validation set no longer decreases in 10 consecutive iterations, the training stops early.

The AddNet-SCL model was implemented by Python 3.7 and based on the Pytorch 1.8 framework. The training process used Adam optimizer [38] to optimize the hybrid loss through an adaptive learning rate scheme to achieve convergence. The batch size is set to 32, and the initial learning rate is 0.003 during training. The graphs of the loss convergence and accuracy were shown in Fig. 7, which includes the loss and accuracy of the training set and the validation set. Fig. 7 (a), (b), (c) show the loss convergence and accuracy curves of additive convolution on training and validation set during training; Fig. 7 (d), (e), (f) show the loss convergence and accuracy curves of multiplicative convolution on training and validation set during training. From the graphs, we can see that the additive convolution can also fit the data well using the ℓ_1-norm distance as the similarity measure.

G. Postprocessing

To obtain a continuous warning device, an event-based post-processing method was used to convert AddNet-SCL

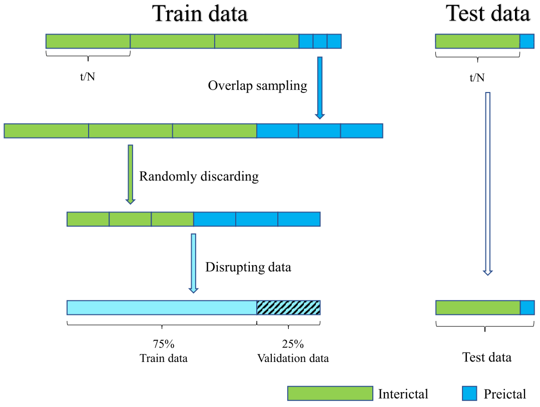


Fig. 8. Strategies for experimental training and testing.

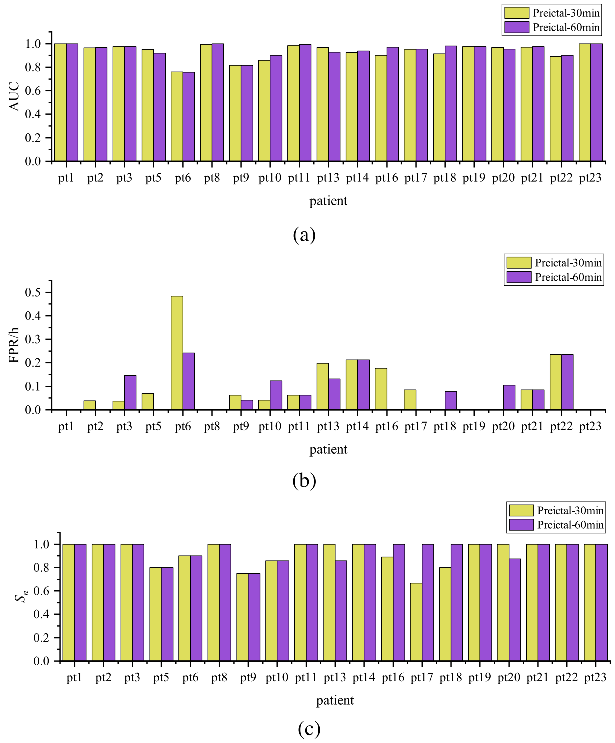
into a practical seizure predictor through a persistent warning scheme [39]. Some works used k of n [12], Kalman filtering [13] and moving average filtering scheme [5], [16], [40]. In this work, we refer to the previous works, such as [5], [16], [40], adopt similar postprocessing methods and moving average filtering to predict seizure event, and the length of the moving average filter was set to 1 minute. Specifically, we input continuous test data from a seizure event to obtain the conditional probability p_i of each sample. The probability of the i-th sample represents the conditional probability that the predictor judges that it belongs to preictal period. Then, a moving average filter was used to smooth p_i in the time dimension and removed outliers to obtain a more reliable probability $p_s(i)$. When $p_s(i)$ exceeds the threshold w, it counts as one alarm, where w is trade-off threshold parameter for sensitivity and FPR, and we experimentally set $w = 0.5$. However, continuous alarms in a short period of time could increase the false alarm rate. Therefore, as in the literature based on event prediction, we defined a 30 minutes refractory period, which indicated that other alarms will be ignored for a period of time after the alarm is triggered [5].

III. Results

In this section, some details about the experimental setup were first added, and then the results were shown compared with those of baseline method. Finally, the effect of experimental conditions on performance was described.

A. Experimental Settings

To better distinguish the data of each seizure event, we choose LOOCV strategy, which tests the data of each seizure event separately to ensure the reliability of the results. The specific plan is shown in Fig 8. Given a patient's data contains N seizure events and t hours interictal period, all t hour data is divided into N parts, each with approximately t/N hours data. Then, N tests are performed based on N seizure events. In each test, one of the N events is selected as the test data, in which the preictal data and the corresponding interictal data maintain temporal continuity, respectively. The remaining $N-1$ pairs of data are used as training data. Since there may be an overfitting problem in training stage, a part of the training data needs to be used as validation data to monitor whether the model is overfitting. In this work, 25% of the preictal and interictal samples in the training set were selected as the validation set, respectively. Finally, the average of N results were calculated to obtain the prediction performance.



(a)

(b)

(c)

Fig. 9. Performance comparison of different lengths of preictal period. (a) is the comparison of AUC, (b) is the comparison of FPR, (c) is the comparison of sensitivity.

In the selection of evaluation metrics, we used four commonly used event-based metrics in seizure prediction, namely sensitivity (S_n), AUC, FPR, and p-value. Sensitivity typically indicates the probability of correctly predicting seizures in a given time period [39]. In a macro event-based sense, it is defined as the proportion of correctly predicted seizures to the total number of seizures. AUC is a metric for evaluating the classification performance. Assuming that each class has the same prior probability, the random classification method can obtain an AUC of 0.5, while a perfect classifier can reach 1.0. FPR represents the number of false alarms per hour. According to feedback from clinicians, when the sensitivity is higher than 75%, FPR may be the single most important measurement metric [41]. p-value is to judge whether the prediction system is statistically superior to the random predictor. Suppose the algorithm identifies n of N seizures for a single patient, the one-sided p-value is used to evaluate the significance of an improvement over chance.

B. Overall Performance

Several end-to-end technologies are used as baseline models, and experiments were conducted under the same conditions to evaluate the performance of AddNet-SCL. Then,

TABLE IV
SEIZURE PREDICTION PERFORMANCE OF EACH MODEL ON CHB-MIT DATABASE

Patient	1D+CNN [42]				DCNN+Bi-LSTM [14]				ResCNN				SCL-AddNets			
	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value
1	0.995	100.0	0.000	<0.001	0.975	100.0	0.000	<0.001	0.999	100.0	0.000	<0.001	0.999	100.0	0.000	<0.001
2	0.650	66.7	0.077	**0.117**	0.669	66.7	0.269	0.006	0.792	66.7	0.192	0.012	0.865	100.0	0.039	<0.001
3	0.959	100.0	0.037	<0.001	0.932	100.0	0.073	<0.001	0.975	100.0	0.037	<0.001	0.976	100.0	0.037	<0.001
5	0.909	80.0	0.277	0.002	0.863	100.0	0.346	<0.001	0.949	80.0	0.346	0.001	0.951	80.0	0.069	<0.001
6	0.620	80.0	0.605	<0.001	0.669	90.0	0.605	<0.001	0.715	90.0	0.403	<0.001	0.761	90.0	0.484	<0.001
8	0.999	100.0	0.000	<0.001	0.996	100.0	0.000	**0.149**	0.999	100.0	0.000	<0.001	0.995	100.0	0.000	<0.001
9	0.694	50.0	0.041	**0.087**	0.749	75.0	0.329	0.002	0.797	75.0	0.103	<0.001	0.815	75.0	0.062	<0.001
10	0.852	85.7	0.124	<0.001	0.824	85.7	0.041	<0.001	0.857	85.7	0.124	<0.001	0.857	85.7	0.041	<0.001
11	0.972	100.0	0.063	<0.001	0.909	100.0	0.250	0.002	0.977	100.0	0.094	<0.001	0.982	100.0	0.063	<0.001
13	0.968	100.0	0.357	<0.001	0.926	100.0	0.331	<0.001	0.902	100.0	0.198	<0.001	0.967	100.0	0.198	<0.001
14	0.893	100.0	0.212	0.009	0.862	100.0	0.212	<0.001	0.889	87.5	0.212	<0.001	0.925	100.0	0.212	<0.001
16	0.915	88.9	0.167	<0.001	0.926	88.9	0.000	0.002	0.933	88.9	0.000	<0.001	0.899	88.9	0.177	<0.001
17	0.852	66.7	0.170	**0.056**	0.808	66.7	0.339	**0.065**	0.935	66.7	0.085	0.17	0.949	66.7	0.085	0.015
18	0.825	80.0	0.079	<0.001	0.829	80.0	0.039	<0.001	0.924	80.0	0.000	<0.001	0.913	80.0	0.000	<0.001
19	0.906	100.0	0.077	0.009	0.766	50.0	0.077	0.018	0.976	100.0	0.077	<0.001	0.974	100.0	0.000	<0.001
20	0.996	87.5	0.157	<0.001	0.956	87.5	0.157	<0.001	0.969	87.5	0.052	<0.001	0.968	100.0	0.000	<0.001
21	0.912	100.0	0.213	<0.001	0.870	100.0	0.384	<0.001	0.920	100.0	0.213	<0.001	0.971	100.0	0.085	<0.001
22	0.827	100.0	0.469	0.003	0.812	100.0	0.586	0.004	0.803	100.0	0.528	0.002	0.889	100.0	0.235	0.001
23	0.995	100.0	0.141	<0.001	0.999	100.0	0.000	<0.001	0.999	100.0	0.000	<0.001	0.999	100.0	0.000	<0.001
Aver	0.881	88.7	0.172	-	0.860	89.0	0.213	-	0.911	89.9	0.140	-	0.929	93.0	0.094	-

where SPH=1min, SOP=30min, interictal-ictal distance is 4 hours, p-value is at a significance value of 0.05, and significance levels of >0.05 are marked with bold.

TABLE V
SEIZURE PREDICTION PERFORMANCE OF EACH MODEL ON KAGGLE DATABASE

Patient	1D+CNN [42]				DCNN+Bi-LSTM [14]				ResCNN				SCL-AddNets			
	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value	AUC	$S_n(\%)$	FPR/h	p-value
Dog1	0.606	50.0	0.113	**0.187**	0.561	50.0	0.352	**0.057**	0.634	50.0	0.352	**0.074**	0.669	75.0	0.163	0.009
Dog2	0.921	85.7	0.120	<0.001	0.956	100.0	0.048	<0.001	0.945	100.0	0.036	<0.001	0.914	85.7	0.096	<0.001
Dog3	0.825	83.3	0.175	<0.001	0.836	91.7	0.189	<0.001	0.860	83.3	0.172	<0.001	0.849	91.7	0.146	<0.001
Dog4	0.755	85.7	0.209	<0.001	0.748	85.7	0.191	<0.001	0.756	92.9	0.191	<0.001	0.766	92.9	0.157	<0.001
Dog5	0.935	100.0	0.053	<0.001	0.924	80.0	0.054	<0.001	0.948	80.0	0.054	<0.001	0.934	100.0	0.04	<0.001
Aver	0.808	80.9	0.134	-	0.805	81.5	0.167	-	0.829	81.2	0.161	-	0.831	89.1	0.120	-

where SPH=5min, SOP=60min, p-value is at a significance value of 0.05, and significance levels of >0.05 are marked with bold.

TABLE VI
COMPUTATIONAL COMPLEXITY COMPARISON OF BASELINE MODELS

Model	Parameters ($\times 10^6$)	Flops ($\times 10^6$) (Mul-Add)	Convolution energy (mJ)	Latency on CPUs (clock cycles ($\times 10^6$))	Performance (AUC-$S_n(\%)$-FPR/h)
DeepConvNet [43]	0.13	73.0 (36.5-36.5)	0.168	219.0	0.904-89.7-0.312
EEGNet [44]	0.0025	24.0 (12.0-12.0)	0.055	72.0	0.883-87.8-0.405
1D+CNN [42]	1.07	160.6 (80.3-80.3)	0.369	481.8	0.881-88.7-0.172
DCNN+MLP [14]	0.43	164.8 (82.4-82.4)	0.388	494.4	0.861-86.9-0.208
DCNN+Bi-LSTM [14]	0.058	162.8 (81.4-81.4)	0.374	488.4	0.860-89.0-0.213
ResCNN	0.12	61.8 (30.9-30.9)	0.142	185.4	0.911-89.9-0.140
This work	0.12	61.8 (7.5-54.3)	0.077	138.6	0.929-93.0-0.094

we made extensive comparison with the several state-of-the-art methods.

1) 1D+CNN [42] first used one-dimensional convolution to perform dimensionality reduction and feature extraction from the data, and then used CNN for classification.
2) DCNN+Bi-LSTM [14] used deeper CNN to extract information from complex original EEG signals, and then employed a bi-directional long short-term memory (Bi-LSTM) as a classifier.
3) ResCNN added residual connection to CNN and have the same network structure as the proposed method, which allows for more intuitive comparison.

The performance comparison between the proposed method and the baseline models in the two databases are shown in Table IV and Table V. In the CHB-MIT database, the proposed model achieves an AUC of 0.929, a sensitivity of 93.0%, and an FPR of 0.094/h. In addition, under the 95% confidence interval, the improvement over chance of the prediction model in 19 patients is statistically significant. In the Kaggle database, the proposed method obtains 0.831 AUC, 89.1% sensitivity, and 0.120/h FPR. Moreover, we also show the parameter amount and computational complexity of each model. As shown in Table VI, the proposed model has roughly 0.12×10^6 parameters, and the use of addition instead of massive multiplication greatly reduces the computational cost. Ignoring a small number of multiplications in the normalization layer, our model has 7.57×10^6 multiplications and 54.3×10^6 additions. Compared with the previous end-to-end

TABLE VII
THE TIME COST OF MULTIPLICATIVE AND ADDITIVE CONVOLUTIONAL NETWORKS

Model	Training time cost per batch/ms	Testing time cost per batch/ms
Multiplicative convolution	28.1	3.35
Additive convolution	26.3	2.69

model, the amount of parameters and calculations have been greatly reduced. ResCNN and the proposed model have same network structure and the same amount of parameters. The difference is that additive convolution is used instead of multiplicative convolution. Through the energy consumption in Fig. 3 and the different delays in instruction tables.[1] it is calculated that the proposed method is superior to the typical convolution operation under the same structure in terms of energy consumption and speed. For example, the latency of floating point addition and multiplication in VIA Nano 3000 series is 2 CPU clock cycles and 4 CPU clock cycles respectively. AddNet-SCL with 7.57×10^6 multiplications and 54.3×10^6 additions will produce 138.6M clock cycles latency, while ResCNN with 30.9×10^6 additions and 30.9×10^6 multiplications will produce 185.4M clock cycles latency in this CPU. Although existing deep learning frameworks do not yet support additive convolution, we show the time cost of adder network and multiplicative convolution network in Table VII. From the training time cost, it can be seen that the time cost of additive convolution is not obvious compared with the typical multiplicative convolution. This is because the gradient calculation method has changed and has not been perfectly optimized in CUDA and CUDNN, and pytorch does not currently support additive convolution. There is no need for gradient propagation during testing, thus the speed of additive convolution is significantly better than that of multiplicative convolution, and the time consumption is reduced by about 20%, further optimization of additive convolution can achieve better improvement.

We compared with several lightweight models with few parameters (such as DeepConvNet [43], EEGNet [44], DCNN-BiLSTM [14]) under the same experimental conditions in Table VI. Although EEGNet has a small number of parameters, the number of floating-point operations is relatively high, and it is accompanied by a high FPR and performance degradation. DCNN+Bi-LSTM uses Bi-LSTM instead of linear fully connected layer as a classifier, which greatly reduces the number of parameter. However, DCNN still has a large computational cost, and the iterative operation of LSTM makes the classification process more complicated. DeepConvNet and the proposed method have similar parameters, and our method outperforms DeepConvNet in both computational cost and performance. From the experimental results, it can be seen that these models with few parameters are less effective in some patients and cannot distinguish between preictal and interictal period, which proves that direct reduction of network parameters is often accompanied by performance degradation. Our method outperforms both models in terms of results, which indicates that the proposed method achieves better balance in the amount of parameters, computational cost and model performance.

C. Ablation Study

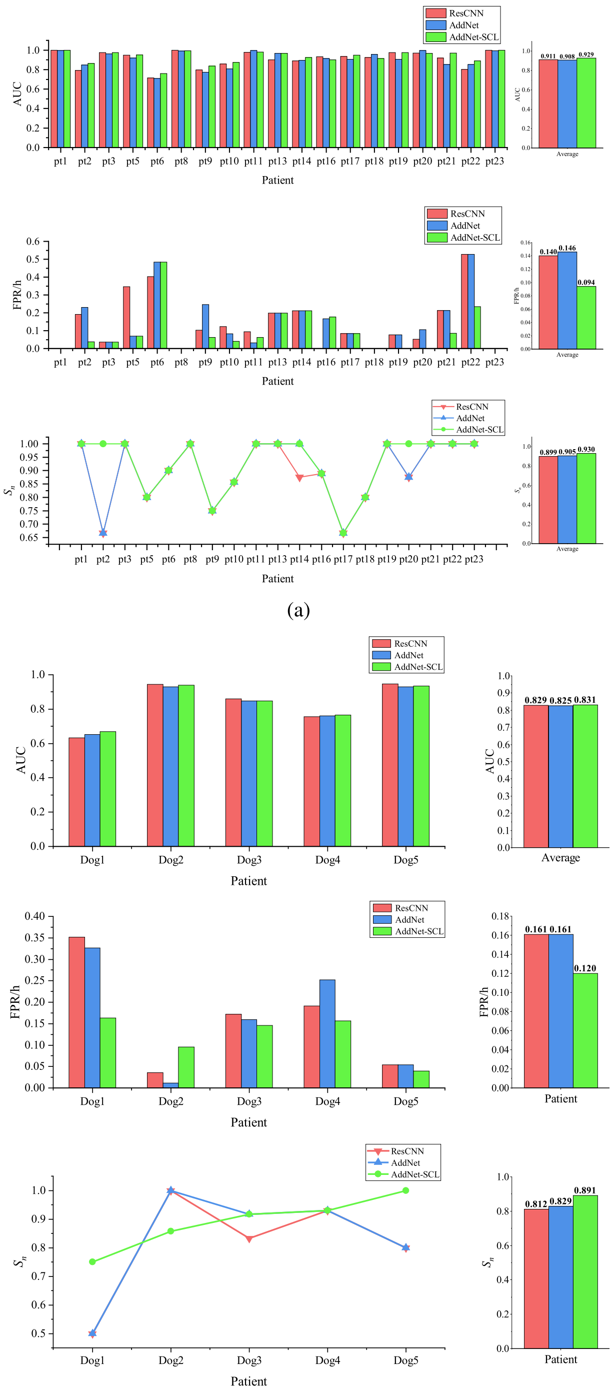
The proposed method used additive convolution and supervised contrastive loss. In order to further verify the effectiveness of additive convolution and contrastive loss, we use three models for ablation study. The first model is ResCNN, which uses the cross-entropy as the loss function. The second model changes the multiplicative convolution of the first model to additive convolution, and also uses the cross-entropy as the loss function. The third model is the proposed AddNet-SCL, which has the same structure as the second model and uses a hybrid function of supervised contrastive loss and the cross-entropy loss. The experimental results are shown in Fig. 10. In the CHB-MIT database, the average performance of ResCNN is 0.911 AUC, 0.140/h FPR, and 89.9% sensitivity; the average performance of AddNet is 0.908 AUC, 0.146/h FPR, and 90.5% sensitivity; the average performance of AddNet-SCL is 0.929 AUC, 0.094/h FPR and 93.0% sensitivity. In the Kaggle database, the average performance of ResCNN is 0.829 AUC, 0.161/h FPR, and 81.2% sensitivity; the average performance of AddNet is 0.825 AUC, 0.161/h FPR, and 82.9% sensitivity; the average performance of AddNet-SCL is 0.831 AUC, 0.120/h FPR and 89.1% sensitivity. Comparing the results of ResCNN and AddNet, it can be seen that the performance of additive convolution is similar to that of typical convolution. Compared with AddNet, the overall performance improvement of AddNet-SCL indicates the effectiveness of the contrastive loss on the task. The higher sensitivity and lower FPR suggest that samples in preictal and interictal periods can be better distinguished.

We visualized the features of the projection space through t-SNE visualization technique and compared the effect before and after using the contrastive loss. As can be seen from the comparison of Fig. 11 (a), (b), the contrastive loss can provide a better classification boundary, gather the features of the same class, and push the features of different classes away; similarly, from the comparison of Fig. 11 (c), (d), it can also be seen that the contrastive loss can make the features of the same class close and the features of different classes stay away.

D. Effects of Preictal Interval

Previous work proposed that seizures may have symptoms several hours ago, and different lengths of preictal periods may have a greater effect on the performance. A too short preictal length makes preictal training data seriously insufficient, thereby affecting performance. Too long preictal period greatly increase training time and reduces efficiency. To obtain a more adequate comparison, we conducted experiments using the commonly used preictal lengths of 30 and 60 minutes. The experimental results are shown in Fig. 9. The AUC, sensitivity, and FPR reached 0.942, 94.9%, and 0.077/h, respectively, at a preictal period length of 60 minutes. The increase in the length of the preictal period brings more preictal data, which enables

[1] Available at: https://www.agner.org/optimize/instruction_tables.pdf



(a)

(b)

Fig. 10. Performance comparison of ResCNN, AddNets, and AddNet-SCL. (a) is the result of the ablation experiment in the CHB-MIT database, (b) is the result of the ablation experiment in the Kaggle database.

the network to learn more distinguishing features, thereby achieving performance improvements in most patients.

E. Compared With Existing Methods

Table VIII shows several of state-of-the-art prediction techniques in recent years. Due to the complexity of the raw EEG data and the large amount of calculation, many works first use feature preprocessing methods convert the raw data into various feature representations, thereby simplifying network design and reducing calculations. Although these works obtain a relatively lightweight networks, they ignore the large computational cost that preprocessing would entail, and the network also contains a large number of multiplication operations. Some end-to-end efficient networks was used in [14], [18], [42] to achieved good performance. Although these methods achieved superior performance, they are not event-based prediction methods and fewer patients are tested, and the validation strategy is different, thus we cannot directly compare with them. Some works performed feature engineering on the raw data, and then used CNN or MLP for classification [5], [13], [15], [45]. Buyukccakir *et al.* [45] chose to use the 10-fold cross-validation method instead of LOOCV validation method in seizure prediction. Li *et al.* [16] first used fast independent component analysis to process EEG data, and then designed a spatio-temporal hierarchical graph convolutional network with active learning scheme to obtain state-of-the-art performance for event-based seizure prediction. However, these methods involve substantial multiplication operations, and they use the cross-entropy as loss function resulting in poor classification margin. The proposed method uses addition instead of multiplication, combined with supervised contrastive loss, to reduce computational complexity while maintaining high performance.

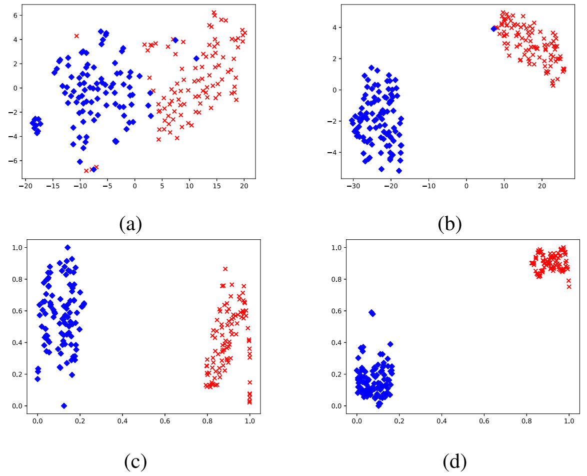


Fig. 11. *t*-SNE plot showing the effect of contrastive loss in feature leaning, where (a), (b) are a comparison group from patient 2 in the CHB-MIT database, (a) uses cross-entropy loss, (b) uses contrastive loss; (c), (d) are a comparison group from patient 11 in the CHB-MIT database, (c) uses cross-entropy loss, (d) uses contrastive loss.

IV. Discussions

In recent years, many researchers in seizure prediction tasks devoted themselves to using DL methods to learn different representations in EEG signals. However, the huge computational complexity, hardware requirements, and insufficient evaluation metrics make it not well used in clinical practice. The model in [5], [12], [13] have fewer parameters to reduce the computational cost, but they ignore the large number of operations brought by feature preprocessing. An end-to-end lightweight network was used in [18], which can greatly

TABLE VIII
PERFORMANCE OF EXISTING METHODS ON THE CHB-MIT DATABASE

Author	Features	Classfier	Validation strategy	No. of patients-seizures	Intericatal-Preictal Intervals (min)	Evaluated hours	No. of patients over chance	Average AUC-S_n(%)-FPR(/h)
Daoud *et al.* 2019 [14]	Raw data	DACE+Bi-LSTM	LOOCV	8-43	NA-60	NA	NA	NA-99.72-0.004
Ozcan *et al.* 2019 [5]	Spectral power, statistical moments, Hjorth	3D CNN	LOOCV	16-77	60-60 120-60 240-60	466.1 419.4 353.5	13/16 14/16 15/16	NA-86.8-0.292 NA-87.0-0.186 NA-85.7-0.096
Zhang *et al.* 2019 [13]	Common spatial pattern statistics	CNN	LOOCV	23-156	30-30	NA	NA	0.90-92-0.12
Xu *et al.* 2020 [42]	Raw data	CNN	NA	7-27	NA-30	NA	NA	0.988-98.8-0.074
Buyukccakir *et al.* 2020 [45]	Hilbert Vibration Decomposition	MLP	10-fold CV	10-62	NA-30	326	NA	NA-89.8-0.081
Zhao *et al.* 2021 [18]	Raw data	CNN	NA	10-NA	NA-30	NA	NA	1.000-99.8-0.005
Yang *et al.* 2021 [15]	STFT spectral images	RDANet	LOOCV	13-64	240-30	268.6	NA	0.913-89.25-NA
Li *et al.* 2021 [16]	FastICA, Spatio-temporal-spectral	STS-HGCN-AL	LOOCV	19-98	Adaptive	453.0	19/19	0.938-95.5-0.109
This work	Raw data	AddNet-SCL	LOOCV	19-105	240-30 240-60	426.5 461.4	19/19 19/19	0.929-93.0-0.094 0.942-94.9-0.077

where NA means not applicable in this work, No. of patients-seizures are the number of patients and seizures involved for evaluation, No. of patients over chance is at a significance value of 0.05.

reduce the complexity of the network by using advanced quantitative methods and pruning operations. These operations are usually accompanied by performance degradation, and CNN compressed by other methods [44] usually suffer from the same problem. Moreover, DL-based methods generally include massive floating number multiplications, which require GPUs acceleration. GPUs needs a lot of support from other hardware, which cannot meet the requirements of device implantability and wearability.

Therefore, we use another method to reduce computational complexity, which uses additive convolution instead of multiplicative convolution. Addition consumes much less energy and time than multiplication, and it also runs fast on the CPUs. In addition, AddNet-SCL can achieve performance similar to that of the traditional CNN, and has broad prospects in the clinical application of seizure predictors. Moreover, unlike most work, we did not design a special network or manually extract features, but used supervised contrastive loss to explore intrinsic patterns of the data. The classification boundary was improved by contrastive loss, and the classification performance is further improved.

Since the ℓ_1 distance was used as the similarity measure, the backpropagation gradient function changes and we showed the gradient size of different layers. An adaptive learning rate was employed in the training process to ensure the convergence of the model, it could be seen from the loss convergence and accuracy curves in Fig. 7 that model can fit the data well, and the effectiveness of the proposed method was demonstrated by comparison with the baseline model. We calculated the computational complexity of popular end-to-end networks. In contrast, our method could achieve lower energy consumption and delay under the same parameters. We explored the effect of additive convolution and contrastive loss on performance through ablation studies, and the improvement effect of the contrastive loss on the classification boundary was further illustrated by feature visualization. By comparing with existing methods, the proposed method has advantages in energy consumption while maintaining high performance.

V. CONCLUSION

In this paper, a novel end-to-end model AddNet-SCL was proposed for seizure prediction based on EEG signals. This model used cheap addition instead of multiplication in convolution to make the network more lightweight while maintaining the accuracy of the network. The model combined with supervised comtrastive learning to learn the intrinsic pattern of the data, there is no special structure design and feature extraction, it has better robustness to hard samples, and improves the poor classification margin caused by the cross-entropy loss. The proposed method obtained better performance than the baseline model in two public databases, 0.942 AUC, 94.9% sensitivity and 0.077/h FPR on 19 patients in the CHB-MIT database, and 0.831 AUC, 89.1% sensitivity and 0.120/h FPR on 5 dogs in the Kaggle database. The experimental results verify the effectiveness of the proposed method in seizure prediction. In addition, additive convolution and supervised contrastive loss are still suitable for other network structures and could be easily combined with other technologies.

REFERENCES

[1] *Promoting Mental Health: Concepts, Emerging Evidence, Practice: A Report of the World Health Organization, Department of Mental Health and Substance Abuse in Collaboration With the Victorian Health Promotion Foundation and the University of Melbourne*, W. H. Org., Geneva, Switzerland, 2005.

[2] D. J. Thurman *et al.*, "Standards for epidemiologic studies and surveillance of epilepsy," *Epilepsia*, vol. 52, pp. 2–26, Sep. 2011.

[3] X. Chen, C. Li, A. Liu, M. J. McKeown, R. Qian, and Z. J. Wang, "Toward open-world electroencephalogram decoding via deep learning: A comprehensive survey," *IEEE Signal Process. Mag.*, vol. 39, no. 2, pp. 117–134, Mar. 2022.

[4] A. Yadollahpour and M. Jalilifar, "Seizure prediction methods: A review of the current predicting techniques," *Biomed. Pharmacol. J.*, vol. 7, no. 1, pp. 153–162, 2015.

[5] A. R. Ozcan and S. Erturk, "Seizure prediction in scalp EEG using 3D convolutional neural networks with an image-based approach," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 11, pp. 2284–2293, Nov. 2019.

[6] M. Bandarabadi, C. A. Teixeira, J. Rasekhi, and A. Dourado, "Epileptic seizure prediction using relative spectral power features," *Clin. Neurophysiol.*, vol. 126, no. 2, pp. 237–248, 2015.

[7] K. Natarajan, R. Acharya U, F. Alias, T. Tiboleng, and S. K. Puthusserypady, "Nonlinear analysis of EEG signals at different mental states," *Biomed. Eng. OnLine*, vol. 3, no. 1, pp. 1–11, Dec. 2004.

[8] H.-T. Shiao *et al.*, "SVM-based system for prediction of epileptic seizures from iEEG signal," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 5, pp. 1011–1022, May 2016.

[9] A. Bhattacharyya, M. Sharma, R. B. Pachori, P. Sircar, and U. R. Acharya, "A novel approach for automated detection of focal EEG signals using empirical wavelet transform," *Neural Comput. Appl.*, vol. 29, no. 8, pp. 47–57, 2018.

[10] S. Lahmiri and A. Shmuel, "Accurate classification of seizure and seizure-free intervals of intracranial EEG signals from epileptic patients," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 3, pp. 791–796, Mar. 2018.

[11] L. Kuhlmann *et al.*, "Epilepsyecosystem.Org: Crowd-Sourcing reproducible seizure prediction with long-term human intracranial EEG," Brain, vol. 141, no. 9, pp. 2619–2630, Aug. 2018.

[12] N. D. Truong *et al.*, "Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram," *Neural Netw.*, vol. 105, pp. 104–111, Sep. 2018.

[13] Y. Zhang, Y. Guo, P. Yang, W. Chen, and B. Lo, "Epilepsy seizure prediction on EEG using common spatial pattern and convolutional neural network," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 2, pp. 465–474, Feb. 2020.

[14] H. Daoud and M. A. Bayoumi, "Efficient epileptic seizure prediction based on deep learning," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 5, pp. 804–813, Oct. 2019.

[15] X. Yang, J. Zhao, Q. Sun, J. Lu, and X. Ma, "An effective dual self-attention residual network for seizure prediction," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1604–1613, 2021.

[16] Y. Li, Y. Liu, Y.-Z. Guo, X.-F. Liao, B. Hu, and T. Yu, "Spatio-temporal-spectral hierarchical graph convolutional network with semisupervised active learning for patient-specific seizure prediction," *IEEE Trans. Cybern.*, early access, May 25, 2021, doi: 10.1109/TCYB.2021.3071860.

[17] S. Zhao, J. Yang, Y. Xu, and M. Sawan, "Binary single-dimensional convolutional neural network for seizure prediction," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Oct. 2020, pp. 1–5.

[18] S. Zhao, J. Yang, and M. Sawan, "Energy-efficient neural network for epileptic seizure prediction," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 1, pp. 401–411, Jan. 2022.

[19] H. Chen *et al.*, "AdderNet: Do we really need multiplications in deep learning?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1468–1477.

[20] G. F. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large margin deep networks for classification," 2018, *arXiv:1803.05598*.

[21] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. 32nd Conf. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 1–11.

[22] A. H. Shoeb, "Application of machine learning to epileptic seizure onset detection and treatment," Ph.D. dissertation, Dept. Harvard Univ.-MIT Division Health Sci. Technol., Cambridge, MA, USA, , Massachusetts Inst. Technol., 2009.

[23] B. H. Brinkmann *et al.*, "Crowdsourcing reproducible seizure forecasting in human and canine epilepsy," *Brain*, vol. 139, no. 6, pp. 1713–1722, 2016.

[24] T. Maiwald, M. Winterhalder, R. Aschenbrenner-Scheibe, H. U. Voss, A. Schulze-Bonhage, and J. Timmer, "Comparison of three nonlinear seizure prediction methods by means of the seizure prediction characteristic," *Phys. D, Nonlinear Phenomena*, vol. 194, nos. 3–4, pp. 357–368, 2004.

[25] E. B. Assi, D. K. Nguyen, S. Rihana, and M. Sawan, "Towards accurate prediction of epileptic seizures: A review," *Biomed. Signal Process. Control*, vol. 34, pp. 144–157, Apr. 2017.

[26] Y. Wang *et al.*, "E2-train: Training State-of-the-art CNNs with over 80% energy savings," 2019, *arXiv:1910.13349*.

[27] T.-K. Hu, T. Chen, H. Wang, and Z. Wang, "Triple wins: Boosting accuracy, robustness and efficiency together by enabling input-adaptive inference," 2020, *arXiv:2002.10025*.

[28] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50X fewer parameters and < 0.5 MB model size," 2016, *arXiv:1602.07360*.

[29] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[30] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[31] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.

[32] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 1580–1589.

[33] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[34] M. Courbariaux, Y. Bengio, and J.-P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3123–3131.

[35] M. Horowitz, "1.1 Computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 10–14.

[36] P. Khosla *et al.*, "Supervised contrastive learning," 2020, *arXiv:2004.11362*.

[37] A. Nasiri and J. Hu, "SoundCLR: Contrastive learning of representations for improved environmental sound classification," 2021, *arXiv:2103.01929*.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[39] D. E. Snyder, J. Echauz, D. B Grimes, and B. Litt, "The statistics of a practical seizure warning system," *J. Neural Eng.*, vol. 5, no. 4, p. 392, 2008.

[40] A. Bhattacharyya and R. B. Pachori, "A multivariate approach for patient-specific EEG seizure detection using empirical wavelet transform," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 9, pp. 2003–2015, Sep. 2017.

[41] V. Shah, M. Golmohammadi, I. Obeid, and J. Picone, "Objective evaluation metrics for automatic classification of eeg events," *Biomed. Signal Process., Innov. Appl.*, vol. 1, no. 1, pp. 1–21, 2021.

[42] Y. Xu, J. Yang, S. Zhao, H. Wu, and M. Sawan, "An end-to-end deep learning approach for epileptic seizure prediction," in *Proc. 2nd IEEE Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, Aug. 2020, pp. 266–270.

[43] R. T. Schirrmeister *et al.*, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Hum. Brain Mapping*, vol. 38, pp. 5391–5420, Nov. 2017.

[44] V. Lawhern, A. Solon, N. Waytowich, S. M. Gordon, C. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 15, no. 5, Jul. 2018, Art. no. 056013.

[45] B. Büyükçakır, F. Elmaz, and A. Y. Mutlu, "Hilbert vibration decomposition-based epileptic seizure prediction with neural network," *Comput. Biol. Med.*, vol. 119, Apr. 2020, Art. no. 103665.