Article

A Novel Simplified Convolutional Neural Network Classification Algorithm of Motor Imagery EEG Signals Based on Deep Learning

Feng Li [1,2,†], **Fan He** [1,2,†], **Fei Wang** [3,*], **Dengyong Zhang** [1,2], **Yi Xia** [1,2] and **Xiaoyu Li** [1,2]

1   School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China; Lif@csust.edu.cn (F.L.); hf0208@stu.csust.edu.cn (F.H.); zhdy@csust.edu.cn (D.Z.); csustxy@gmail.com (Y.X.); csustecolee@163.com (X.L.)
2   Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha 410114, China
3   School of Software, South China Normal University, Guangzhou 510631, China
*   Correspondence: cce_wf@csust.edu.cn
†   These authors contributed equally to this work.

check for updates

Abstract: Left and right hand motor imagery electroencephalogram (MI-EEG) signals are widely used in brain-computer interface (BCI) systems to identify a participant intent in controlling external devices. However, due to a series of reasons, including low signal-to-noise ratios, there are great challenges for efficient motor imagery classification. The recognition of left and right hand MI-EEG signals is vital for the application of BCI systems. Recently, the method of deep learning has been successfully applied in pattern recognition and other fields. However, there are few effective deep learning algorithms applied to BCI systems, particularly for MI based BCI. In this paper, we propose an algorithm that combines continuous wavelet transform (CWT) and a simplified convolutional neural network (SCNN) to improve the recognition rate of MI-EEG signals. Using the CWT, the MI-EEG signals are mapped to time-frequency image signals. Then the image signals are input into the SCNN to extract the features and classify them. Tested by the BCI Competition IV Dataset 2b, the experimental results show that the average classification accuracy of the nine subjects is 83.2%, and the mean kappa value is 0.651, which is 11.9% higher than that of the champion in the BCI Competition IV. Compared with other algorithms, the proposed CWT-SCNN algorithm has a better classification performance and a shorter training time. Therefore, this algorithm could enhance the classification performance of MI based BCI and be applied in real-time BCI systems for use by disabled people.

Keywords: motor imagery (MI); electroencephalogram (EEG); simplified convolutional neural network (SCNN); continuous wavelet transform (CWT)

1. Introduction

A brain-computer interface (BCI) is a direct communication and control system that is established between the human brain and an electronic device [1,2]. BCI systems have important application value in many fields, especially in the field of medical treatment [3]. Various electroencephalogram (EEG) signals have been used in BCI systems, such as P300 potentials [4,5], steady state visual evoked potentials (SSVEP) [6,7], and motor imagery (MI) [8,9]. Among these EEG signals, the MI signal is one of the most common signals, as it can be generated spontaneously without any stimulation. However, the recognition of MI-EEG is often difficult for several reasons. First, the high-dimensional MI-EEG signal is too weak and its signal-to-noise ratio is low [10]. Secondly, the MI-EEG signal is a nonlinear

and non-stationary signal, which means that its parameters, such as mean and variance change along with time [11]. Further, MI signals are time-varying signals that depend on time variables [12]. In general, MI-EEG signals are highly complex and unstable signals, resulting in challenges for MI-EEG feature extraction and classification.

Feature extraction plays a crucial role in the recognition of MI-EEG signals. However, feature extraction is often used in conjunction with preprocessing methods. The selection of preprocessing methods has an important impact on the extraction from the original MI-EEG to the effective features. Traditional methods usually use energy features, and employ preprocessing methods, such as frequency or temporal filtering, to map the raw MI-EEG data into energy signals [13–15]. Duan et al. [16] used a spatial filter to map the MI-EEG data to an energy signal containing the most obvious features. Dose et al. [17] extracted the time domain energy and spatial location features directly from the raw EEG. Sturm et al. [18] applied layer-wise relevance propagation (LRP) and deep neural networks (DNNs) to convert the MI-EEG into frequency energy characteristics. Zhang R and Zong et al. [19] used a one-versus-rest filter to analyze the MI-EEG signal, and then extract the spatial and temporal features.

Ming-ai Li and Zhang et al. [20] used wavelet packet transform (WPT) to decompose and reconstruct the MI-EEG to obtain mu rhythm and beta rhythm energy feature information. Recently, a time-frequency analysis method mapped MI-EEG signals to time-frequency image signals. Zhichuan Tang and Li et al. [21] mapped the MI-EEG signals to temporal-frequency image signals using fast Fourier transform (FFT). Tabar and Halici. [22] employed short time Fourier transform (STFT) to perform temporal-frequency analysis on MI-EEG. Although FFT and STFT have been used to map MI-EEG to time-frequency images, FFT cannot fully capture the details of the signal and the window of STFT cannot change with the frequency. Furthermore, FFT and STFT were difficult to balance both global and local features when dealing with nonlinear unsteady MI-EEG signals [23]. In this study, we use the continuous wavelet transform (CWT), which can solve these problems by decomposing the signal into different segments and provide a window that changes with the frequency, with a high time resolution.

Deep learning has a strong ability to handle complex and nonlinear high-dimensional data, and it allows machines to learn the characteristics or classify the input data [24]. Deep learning has been successfully applied to pattern recognition, especially natural language processing, computer vision, and speech recognition [25–28]. Due to the ability of excellent self-learning characteristics [29–31], deep learning has gradually been applied to the identification of EEG data, such as P300 [32,33], SSVEP [34], and MI [35]. Tayeb et al. [36] used three-channel MI-EEG as the input of the STFT, and the proposed convolutional neural network (pCNN) was trained and tested with the output data of the STFT.

Li and Zhu et al. [37] used optimal wavelet packet transform (OWPT) to construct MI-EEG feature vectors, which were used to train long short-term memory (LSTM) based on a recurrent neural network (RNN). The algorithm performs well on dataset III of the BCI Competition 2003; however, its structure is overly complex. Liu et al. [32] used a new CNN structure to classify P300 signals. The algorithm performs well on the BCI competition P300 datasets. Although the classification results of the above deep learning methods perform well, these networks are commonly complex and have massive parameters. In this paper, we propose a new neural network that not only simplifies the network structure and reduces the parameters but also improves the classification performance.

In the study, a new CWT-simplified convolutional neural network (SCNN) algorithm is proposed, based on using deep learning to identify MI-EEGs. First, the CWT is used to map the MI-EEG data into time-frequency image signals, which contain time and frequency domain features. Second, we propose a convolutional neural network without pooling layers, named SCNN. There are two convolutional layers in SCNN to extract the time and frequency domain features, and finally, we use softmax to classify the MI-EEG data. The above method is validated on the BCI Competition IV Dataset 2b. The experimental results show that the performance of our algorithm is improved, compared with other algorithms. In addition, when using the same MI-EEG signal and SCNN, compared with

common spatial pattern (CSP), FFT, and STFT, the test results show that the performance of CWT is better.

2. Method

2.1. Datasets

In this study, we selected a public dataset (BCI Competition IV Dataset 2b) to validate the effectiveness of the proposed algorithm. The dataset was collected from nine subjects at electrodes C3, Cz, and C4 with a sampling frequency of 250 Hz. Each subject participated in five sessions. We choose the data in the last two sessions (04E and 05E) with feedback to analyze, each of which contained 160 trials. The experimental procedure of one trial is illustrated in Figure 1. At the beginning of each trial, a gray face label appeared on the screen. At 2 s, the experimental device emitted a short beep to remind the subject to prepare for the experiment. From 3 s to 7.5 s, the subject started to imagine the movement direction of the gray face (left or right), which depended on the cue. If the gray face moved in the same direction as the cue, a green smiley would appear on the screen, otherwise a red sad face would appear. At 7.5 s, the cue disappeared and the screen turned blank, and then the next trial would start after a random interval between 1 to 2 s.
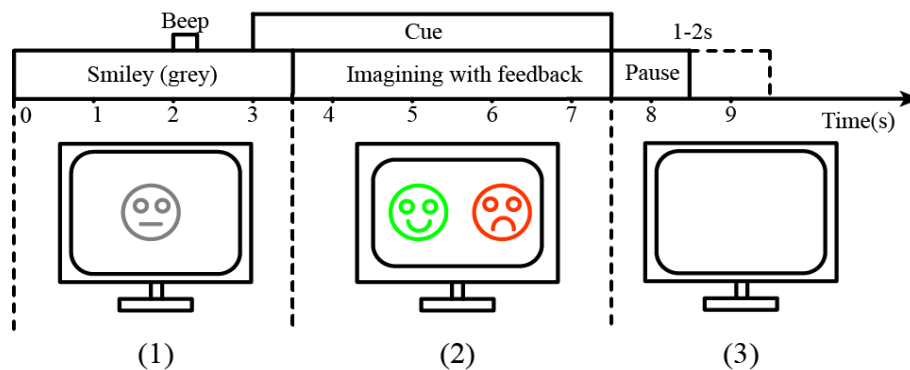

Figure 1. The experimental procedure of one trial. Moving along the time axis, a gray face label first appeared on screen. The subject was asked to control the movement of the face icons by imagining left and right hand movements, according to a cue. A red or green face label on the second screen provided the subject with feedback. While waiting to prepare for the next trial, the screen was blank for a time interval between 1 to 2 s.

2.2. Data Analysis

In this study, a CWT-SCNN algorithm was proposed to classify motor imagery EEG signals. The flowchart of signal processing in this study is presented in Figure 2. First, the raw MI-EEG is filtered with a 4–35 Hz filter (this frequency range contains important features to identify MI-EEG signals [38]). Second, the sampled signal is mapped into a time-frequency image by CWT, and time-frequency images in the range of mu and beta rhythms are extracted for training the SCNN. The details of the CWT method and the structure of theh SCNN are described later in this section. Finally, the MI-EEG data are divided into two categories after the SCNN to provide the classification results.
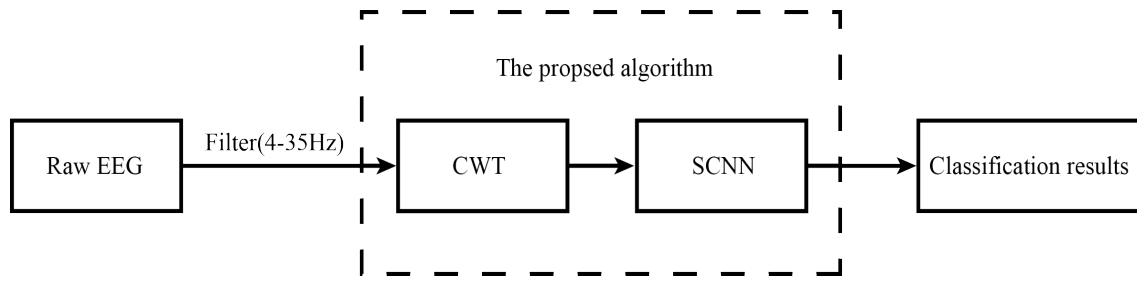
Figure 2. The flowchart of the MI-EEG signal processing in this study. The black arrows represent the flow of signals and the dashed box represents the proposed algorithm. Continuous wavelet transform (CWT) and simplified convolutional neural network (SCNN).

2.2.1. Continuous Wavelet Transform

We use the CWT method to map MI-EEG signals into two-dimensional image signals and extract the mu and beta rhythms from these image signals. The expression of the continuous wavelet transform is shown in Equation (2) [39].

$$W_s(a, \tau) = \frac{1}{\sqrt{a}} \int s(t) \phi^* \left(\frac{t - \tau}{a} \right) dt \qquad (1)$$

where $s(t)$ is the input signal, a is the scale of wavelet transform, ϕ is the wavelet basis function, and τ is the time shift. In order to better extract the local and global features of MI-EEG in time and domain, we select the Morlet wavelet as the wavelet function [40]. Its expression is as follows:

$$\phi(t) = \left(\frac{2}{\pi T^2} \right)^{\frac{1}{4}} \exp \left(-\frac{t^2}{T^2} + jw_c t \right). \qquad (2)$$

The expression of frequency is:

$$\Phi(w) = \left(\frac{T^2}{2\pi} \right)^{\frac{1}{4}} \exp \left(-\frac{(w - w_c)^2}{4T^2} \right) \qquad (3)$$

where $\phi(t)$ is the time domain expression after CWT and $\Phi(w)$ is the frequency domain expression after CWT, after the Morlet wavelet is determined as the analysis wavelet. By analyzing the MI-EEG data, T and w_c of the wavelet function are determined. A large amount of literature indicates that the energy of MI-EEG is mainly concentrated in the low frequency band below 30 Hz (the mu rhythm is 8–12 Hz and the beta rhythm is 18–26 Hz) [41]. The Morlet wavelet's center frequency f_c is 0.8125, and T is 0.04. The minimum scale a_{min} is 1 and the maximum scale a_{max} is 250.

The sampling frequency is 250 Hz, so each trial has 1000 time sample points. In order to prevent the loss of effective features, the frequency range of mu and beta rhythms is appropriately extended (mu is 4–15 Hz, beta is 19–30 Hz). Feature images with the same size as (22, 1000) the mu and beta rhythm were extracted. The length of the sample points of time is 1000, and 22 is the length of the sample points of the frequency. These two feature pictures are combined to form (44, 1000). Through analyzing the signals of the three electrodes C3, Cz, and C4 ($N_C = 3$), the time-frequency feature images obtained are (3@44×1000). To reduce the number of time sample points, we take an average of every five points on the time axis of the feature image. The feature images of the three electrodes are shown in Figure 3 ($N_C@N_{fr}×N_t$). The horizontal axis of the feature image is the time sample points, and the vertical axis is the frequency. A training sample (3@44×200) consists of feature images of the three electrodes.
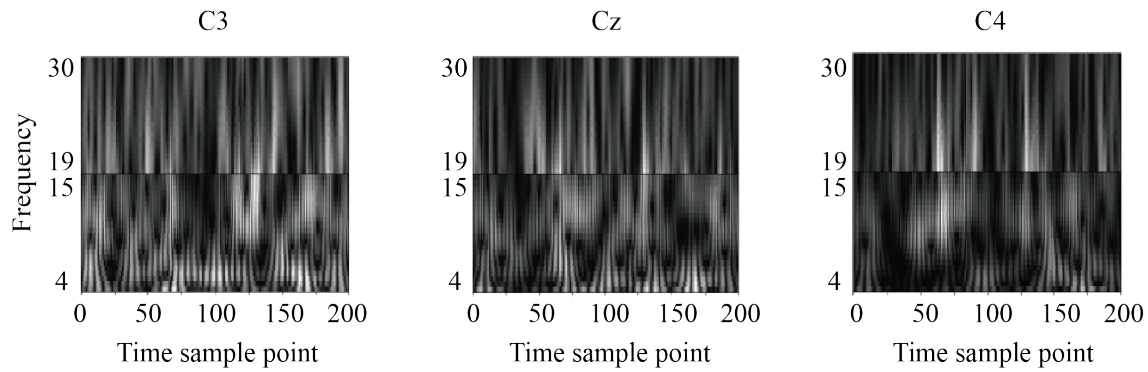
Figure 3. The feature images of the three electrodes (C3, Cz, and C4).

2.2.2. SCNN

In this study, we propose a six-layer SCNN, which includes an input layer, two convolutional layers, an flatten layer, a fully connected layer, and an output layer. The first layer is the input layer. The second layer is the C2 layer, which consists of a convolution layer, a batch normalization layer (BN), and a ectified linear units (ReLu). The convolution layer has eight filters of size (N_{fr}, 1), which move along the time axis to extract the frequency features. The third layer is the C3 layer, which also consists of a convolution layer, a batch normalization layer (BN), and a ReLu. The C3 layer has 16 filters of size (1, 10), and the filters move along the horizontal axis to extract time domain features. The fourth layer is the flatten layer, which combines the output of C3 into a vector, and the fifth layer is a fully connected layer. Lastly, the softmax layer is applied to predict the probability distribution of the output classes. The SCNN network framework is shown in Figure 4.
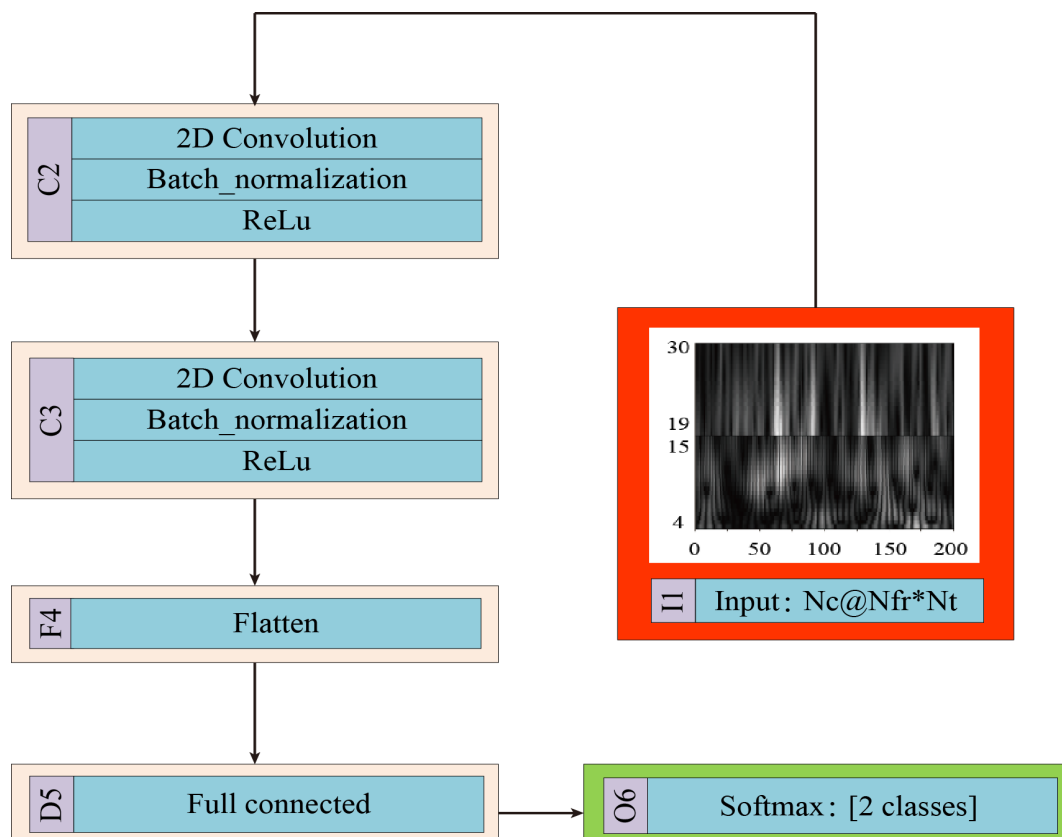

Figure 4. The six-layer SCNN framework for MI classification.

In this SCNN, a neuron is defined as $N(m, k, j)$, where m is the number of layers of the neural network, k is the number of feature maps, and j is the number of position feature maps. If the input and output of a neuron are $x_k^m(j)$ and $y_k^m(j)$, respectively, the relationship between the input and output can be expressed as Equation (4) [42].

$$y_k^m(j) = f(a) = f\left(x_k^m(j)\right) \tag{4}$$

where f is the activation function (ReLu) [43]. Its expression is as follows:

$$f(a) = \ln\left(1 + e^a\right). \tag{5}$$

I1 is the input layer, and the input image is $x_{ij}^1(j)$ ($N_C @ N_{fr} * N_t$). Then the output expression of the C2 layer is as follows:

$$y_k^2 = f(a) = f\left(\sum_{i=1}^{44} x_{ij}^1 * w_k^2 + b_k^2(j)\right) k = 1, 2, 3 \ldots 8 \tag{6}$$

where w_k^2 is the filter of $(N_{fr}, 1)$, y_k^2 is the output of the C2 layer, $b_k^2(j)$ is the deviation. Eight filters slide along the time axis of the feature image to output feature vectors of eight sizes (1, 200). These feature vectors are regularized by the BN layer before being input to the activation layer, and then input to the C3 layer. The output of the C3 layer is as follows:

$$y_k^3 = f\left(\sum_{i=1}^{10} y_k^2[(j-1) \times 10 + i] \times w_k^3 + b_k^3(j)\right) k = 1, 2, 3 \ldots 16 \tag{7}$$

where w_k^3 is the filter of (1, 10), $b_k^3(j)$ is the biases, y_k^3 is output of the C3 layer. Sixteen filters slide horizontally along the feature vector to obtain vectors of 16 sizes (1, 20). These vectors are regularized by the BN layer before being input to the activation layer, and then input to the F4 layer to be combined into a vector, y4 of (1, 320). D5 is a fully connected layer consisting of 64 neurons, and its output is a vector of (64, 1). The output of the D5 layer is as follows:

$$y^5 = f\left(\sum_{i=1}^{320} y^4 w_i^5 + b^5(j)\right). \tag{8}$$

w_i^5 and $b^5(j)$ are the weights and biases of the D5 layer, respectively. O6 is the softmax layer and consists of two neurons. The output of the O6 layer is as follows:

$$y^6 = f\left(\sum_{i=1}^{64} y^5(i) w^6(i) + b^6(j)\right) \tag{9}$$

Equations (4)–(9) are the forward propagation calculation flow of the SCNN algorithm. The SCNN corrects the weights and biases by the error back propagation algorithm. The SCNN is trained using the labeled training set and E is calculated by the difference between the predicted value and the true value. We update the weights and biases of the neural network using gradient descent [44], and the expression is as follows:

$$w^k = w^k - \eta \frac{\partial E}{\partial w^k} \tag{10}$$

$$b_k = b_k - \eta \frac{\partial E}{\partial b_k}. \tag{11}$$

In the training of the SCNN, we should not calculate the minimum value of the loss as a criterion for judging whether to stop training, but rather set the epoch equal to 150. The network layer and parameters of the SCNN are shown in Table 1.

The SCNN without a pooling layer is proposed to identify the MI-EEG signals. In the C2 layer, the frequency domain features are extracted by sliding along the time axis with a 1D filter of size (N_{fr}, 1). In the C3 layer, the time domain features are extracted by sliding along the horizontal axis with a filter of size (1, 10). There is no pooling layer in SCNN, which not only simplifies the network structure but also avoids missing certain features. Before the feature vectors are input to the ReLu layer, each feature vector is standardized. In the F4 layer, the time and frequency domain feature vectors are combined, then input to the fully connected layer, and finally softmax is used for classification.

Table 1. The output matrix size and network parameters of each layer of SCNN.

Short Name	Layer (Type)	Output Shape	Parameter
I1	Input layer	(44, 200, 3)	None
C2	Convolution layer	(1, 200, 8)	1064
	Batch normalization layer	(1, 200, 8)	4
	Activation layer	(1, 200, 8)	None
C3	Convolution layer	(1, 20, 16)	1296
	Batch normalization layer	(1, 20, 16)	64
	Activation layer	(1, 20, 16)	None
F4	Flatten layer	(none, 320)	None
D5	Fully connected layer	(none, 64)	20,544
O6	Output layer	(none, 2)	130

3. Experimental Results

In this study, 320 trials in the Competition IV Dataset 2b were used to test our algorithm. Table 2 shows the classification accuracy of each subject's MI-EEG data using the convolutional neural network and stacked autoencoder (CNN-SAE) [22], CSP [13], adaptive common spatial patterns (ACSP) [45], deep belief net (DBN), [46] and CWT-SCNN algorithms. Bold text indicates the highest classification accuracy for each subject. As seen from Table 2, the classification performance of deep learning algorithms, such as CWT-SCNN and DBN, is better than the traditional CSP and ACSP algorithms. Furthermore, four among nine subjects (S2, S5, S6, and S8) obtained the highest classification accuracy using the CWT-SCNN algorithm. In addition, the CWT-SCNN algorithm has the highest average classification accuracy and is approximately 5–8% higher than other algorithms.

Table 2. The classification results for five methods. Convolutional neural network and stacked autoencoder (CNN-SAE) [22], common spatial pattern (CSP) [13], adaptive common spatial patterns (ACSP) [45], deep belief net (DBN).

Subject	Classification Accuracy (%)				
	CNN-SAE	**CSP**	**ACSP**	**DBN**	**CWT-SCNN**
S1	**76.0**	66.6	67.5	66.6	74.7
S2	65.8	57.9	55.4	62.5	**81.3**
S3	**75.3**	61.3	62.2	60.0	68.1
S4	95.3	94.0	94.7	**96.8**	96.3
S5	83.0	80.6	76.9	82.0	**92.5**
S6	79.5	75.0	75.9	77.4	**86.9**
S7	74.5	72.5	71.3	**76.6**	73.4
S8	75.3	89.4	89.4	88.8	**91.6**
S9	73.3	85.6	81.3	**86.0**	84.4
Average	77.6	75.9	75.0	77.4	**83.2**

The kappa value is used to evaluate the classification performance of the algorithm and remove the impact of random classification [22]. The calculation expression of the kappa coefficient is as follows:

$$\text{kappa} = \frac{acc - p_e}{1 - p_e} \tag{12}$$

Since two classification problems are studied here, the random classification accuracy in Equation (10) is (P_e = 0.5). Table 3 shows the kappa values using the CNN-SAE [22], CSP [13], ACSP [45], DBN [46], and CWT-SCNN algorithms.

Table 3. The kappa values for five methods.

Subject	Kappa Value				
	CNN-SAE	**CSP**	**ACSP**	**DBN**	**CWT-SCNN**
S1	**0.488**	0.312	0.332	0.302	0.478
S2	0.289	0.192	0.163	0.218	**0.622**
S3	**0.427**	0.206	0.215	0.209	0.347
S4	0.888	0.907	0.915	**0.929**	0.923
S5	0.593	0.632	0.548	0.648	**0.847**
S6	0.495	0.521	0.546	0.567	**0.733**
S7	0.409	0.507	0.498	**0.545**	0.459
S8	0.443	0.798	0.806	0.774	**0.822**
S9	0.415	0.724	0.715	**0.731**	0.684
Average	0.547	0.533	0.526	0.547	**0.657**

As seen from Table 3, compared with traditional algorithms, such as CSP [13] and ACSP [45], random classification has a smaller impact on deep learning algorithms, such as CWT-SCNN. Four of the nine subjects achieved the highest kappa values with the proposed algorithm. Between these, three subjects had kappa values above 0.8 with the proposed algorithm. The highest kappa value of the proposed algorithm for S4 is 0.923, slightly less than that of DBN. In addition, the CWT-SCNN algorithm has the highest average classification accuracy, and is approximately 11–13% higher than other algorithms.

Tables 4 and 5 show the classification accuracies and kappa values of CSP-SCNN, FFT-SCNN, STFT-SCNN, and CWT-SCNN on the BCI Competition IV Dataset 2b. The MI-EEG signal is mapped to the time-frequency image signal using FFT, STFT, and CWT. When using tranditional CSP, a matrix that maximizes the difference between the two types of features can be obtained. Then the image signal or matrix is trained and tested by SCNN using 10×10-fold cross validation. As shown in Table 4, eight of the nine subjects obtained the highest classification accuracy using the CWT-SCNN method. Furthermore, the highest average classification accuracy is obtained with the proposed CWT-SCNN method and is at least 4% higher than the other methods. In Table 5, seven of the nine subjects obtained the highest kappa value with the CWT-SCNN method. Furthermore, the highest kappa value is obtained by using the CWT-SCNN method and is about 7–10% higher than the other methods.

Table 4. The classification results of the CWT-SCNN, CSP-SCNN, fast Fourier transform (FFT)-SCNN, and short time Fourier transform (STFT)-SCNN methods.

Subject	Mean Classification Accuracy (%)			
	CSP-SCNN	**FFT-SCNN**	**STFT-SCNN**	**CWT-SCNN**
S1	65.0	69.1	72.5	**74.7**
S2	74.4	72.8	80.0	**81.3**
S3	64.7	67.8	64.4	**68.1**
S4	95.6	94.4	**96.3**	**96.3**
S5	83.8	88.1	88.8	**92.5**
S6	72.5	72.2	71.6	**86.9**
S7	67.2	67.5	66.3	**73.4**
S8	**92.5**	90.0	91.9	91.3
S9	**84.4**	82.2	80.6	**84.4**
Average	77.8	78.2	79.2	**83.2**

In order to obtain the significance comparison results between the proposed algorithm and other algorithms, we use a non-parametric Friedman test [47,48] to evaluate the classification performance of the statistical significance of the algorithms in Tables 2–5. The alpha value is set to 0.05, and the number of samples is 9. For data in the Table 2, we establish the hypothesis H0: The median classification accuracy of each algorithm is the same for the MI-EEG data. A p value is 0.0147, less than 0.05 (significance value), so the H0 was rejected. It is revealed that there is a significant difference between the classification accuracy of the compared five algorithms. Using the same method, we can obtain that there is a significant difference ($p = 0.0134 < 0.05$) between the classification accuracy of the CWT-SCNN, CSP-SCNN, FFT-SCNN, and STFT-SCNN algorithms for data in the Table 4. Furthermore, for data in Tables 3 and 5, the impacts in the kappa values due to different algorithms are also statically significant ($p = 0.015$ and 0.0179, respectively).

Table 5. The mean kappa value results of the CWT-SCNN, CSP-SCNN, FFT-SCNN, and STFT-SCNN methods.

Subject	Mean Kappa Value			
	CSP-SCNN	**FFT-SCNN**	**STFT-SCNN**	**CWT-SCNN**
S1	0.286	0.364	0.453	**0.478**
S2	0.465	0.463	0.599	**0.622**
S3	0.296	0.341	0.299	**0.347**
S4	0.905	0.887	**0.923**	**0.923**
S5	0.666	0.754	0.769	**0.847**
S6	0.448	0.410	0.443	**0.733**
S7	0.344	0.358	0.330	**0.459**
S8	**0.848**	0.794	0.829	0.822
S9	**0.685**	0.640	0.618	0.684
Average	0.549	0.556	0.585	**0.657**

In order to compare the difference between with and without pooling layers, we add the pooling layer to the C2 and C3 layers of SCNN to form a standard CNN. Table 6 lists the output matrix and the parameters of each network layer, when training the standard CNN with a image signal of size (44, 200). Compared with the CNN, the network parameters of SCNN are reduced by half, which not only saves the calculation cost, but also shortens the training time of the network. The average training time of the CNN is 456 s, which is 45 s longer than the average training time of SCNN (each training set contains 288 trials).

Figure 5a,b shows the classification accuracies and kappa values of the BCI Competition IV Datasets 2b using the CWT-SCNN and CWT-CNN methods. The MI-EEG signal is mapped to the

time-frequency image signal using CWT. Then the image signals perform 10×10-fold cross validation on SCNN and CNN. From Figure 5a,b, compared with CWT-CNN, eight of the nine subjects obtained the highest classification accuracy and kappa value using the CWT-SCNN method. In addition, the CWT-SCNN algorithm has the highest mean classification accuracy and mean kappa value. Compared with the CWT-CNN method, the mean classification accuracy and mean kappa value improved by 2.9% and 5.3%, respectively.

Table 6. The output matrix size and network parameters of each layer of the formed standard CNN.

| Short Name	Layer (Type)	Output Shape	Parameter
I1	Input layer	(44, 200, 3)	None
C2	Convolution layer	(1, 200, 8)	1064
	Batch normalization layer	(1, 200, 8)	32
	Activation layer	(1, 200, 8)	None
P3	MaxPooling layer	(1, 100, 8)	None
C4	Convolution layer	(1, 91, 16)	1296
	Batch normalization layer	(1, 91, 16)	64
	Activation layer	(1, 91, 16)	None
P5	MaxPooling layer	(1, 46, 16)	None
F6	Flatten layer	(none, 736)	None
D7	Fully connected layer	(none, 64)	47,168
O8	Output layer	(none, 2)	130
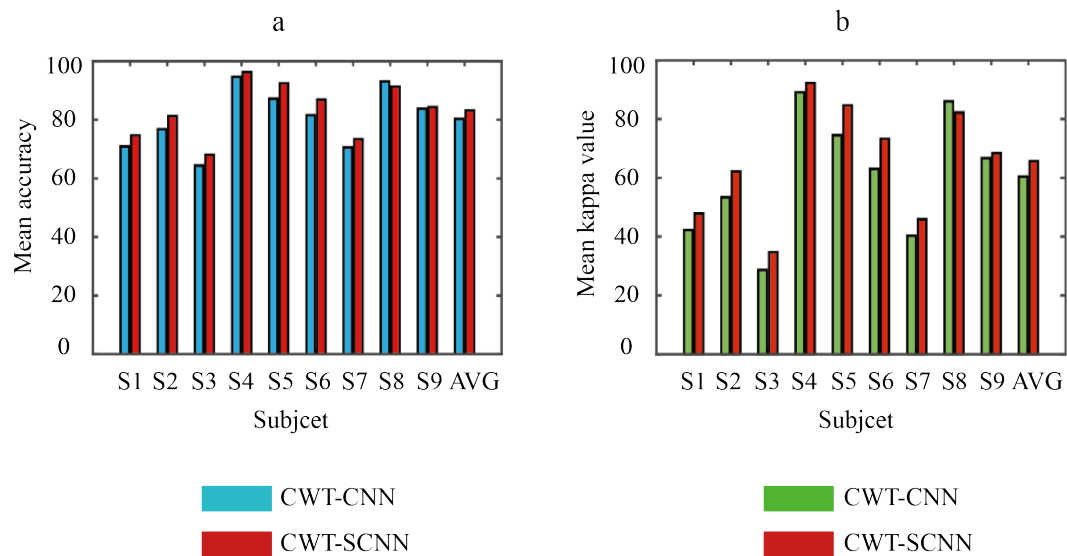
Figure 5. The mean classification accuracies and mean kappa values of the CWT-SCNN and CWT-CNN methods are shown in (**a**,**b**), respectively.

4. Discussion

In this study, the proposed CWT-SCNN algorithm is used to identify the left and right hand MI-EEG signals. After being simply filtered, the EEG signals are mapped to image signals through the CWT. Then the signals are input into the SCNN for feature extraction and classification. Tested by the BCI Competition IV dataset 2b, the average classification accuracy and average kappa value obtained by CWT-SCNN algorithm are 83.2% and 0.651, respectively. Compared with the CSP-SCNN, FFT-SCNN, STFT-SCNN, and CWT-SCNN methods, the results show that the average classification accuracy and average kappa value of the CWT-SCNN method are the best. Furthermore, the experimental results show that the CWT-SCNN method not only has a higher average classification accuracy and

average kappa value, but also has a shorter training time than that of the CWT-CNN method. In short, compared with traditional or deep learning classification methods, the CWT-SCNN method not only improves the classification accuracy and kappa value, but also shortens the training time.

In order to improve the performance of BCI systems, we proposed combining the CNN and SCNN methods to identify MI-EEG signals. As can be seen from Tables 2 and 3, compared with the traditional classification algorithms, CSP or ACSP, the CWT-SCNN method improves each subject's classification accuracy and kappa value. Compared with the deep learning algorithms, CNN-SAE or DBN, the CWT-SCNN method obtains a higher average classification accuracy and average kappa value. In general, compared with traditional or deep learning classification algorithms, the CWT-SCNN method improves not only the classification performance but also the overall performance of the system.

By comparing the classification results and kappa values of different preprocessing methods, CWT is more suitable for coordinating with SCNN to analyze MI-EEG signals than CSP, FFT, and STFT. As shown in Tables 4 and 5, compared with the CSP-SCNN, FFT-SCNN, and STFT-SCNN methods, the CWT-SCNN obtains higher classification results and a higher kappa of CWT-SCNN. The previous works showed that CSP as a linear analysis method may ignore short-term changes in the signal and fail to capture the details of the signal change [49]. Furthermore, FFT cannot capture the local features of MI-EEG signals well [23]. As the window size of STFT is fixed, it can not make the overall and local features clear. CWT can balance both global and local features by decomposing the signal and providing a time-varying window with a high temporal resolution [50]. These may indicate that the CWT method added on SCNN can enhance the classification performance for the MI-EEG signals.

The proposed SCNN framework takes advantage of the feature extraction and classification. As can be seen from Figure 5a,b, compared with the CWT-CNN method, the CWT-SCNN method has a higher classification accuracy as well as a higher kappa value. It can be known from Tables 1 and 6 that, compared with the CNN method, the SCNN method not only has a simple network structure but also has fewer network parameters. The SCNN is typically different from a traditional CNN in that it lacks a pooling layer. Generally, the pooling layer has the effect of reducing image dimensions and parameters. However, previous work has shown that high-resolution signals may lose some important information in the pooling layer [49]. In addition, in order to reduce the dimensionality of the image, the size of the convolution kernel in this method has been appropriately adjusted, similar to those described in the literature [42]. To sum up, compared with CNN method, the proposed SCNN method has a better application value.

5. Conclusions

In this paper, we propose CWT-SCNN, which is a new algorithm for identifying left and right hand motion imagery EEG signals. To obtain the time-frequency image as the feature signal and to better extract the features of the MI-EEG in the next step, the CWT method is used to map the MI-EEG signal after being simply filtered. The application of the CWT method solves the problem where the traditional and current preprocessing methods cannot balance the overall and local features. Then the signals are input into the SCNN to extract the features and classify them, which is upgraded by removing the pooling layer from the traditional CNN structure.

Compared with the CNN method, the SCNN method not only shortens the training time and reduces the parameters but also improves the classification accuracy and kappa value. The proposed SCNN method can improve the overall performance of CNN and can be regarded as a CNN upgrade. Overall, the combined CWT and SCNN method performs better than the traditional or deep learning classification methods. The experimental results show that the CWT-SCNN algorithm performs well and is worth considering for further application in BCI systems. We will continue to improve the performance of the algorithm and we expect the performance of the algorithm to reach the highest levels. Furthermore, in the future we will improve the robustness and classification accuracy of the algorithm and apply it to real-time online BCI systems.

References

1. Wolpaw, J.R.; Birbaumer, N.; McFarl, D.J.; Pfurtscheller, G.; Vaughana, T.M. Brain–computer interfaces for communication and control. *Clin. Neurophysiol.* **2002**, *113*, 767–791. [CrossRef]

2. Birbaumer, N.; Cohen, L.G. Brain–computer interfaces: communication and restoration of movement in paralysis. *J. Physiol.* **2007**, *579*, 621–636. [CrossRef] [PubMed]

3. Kerous, B.; Skola, F.; Liarokapis, F. EEG-based BCI and video games: A progress report. *Virtual Real.* **2018**, *66*, 2992–3005. [CrossRef]

4. Kshirsagar, G.B.; Londhe, N.D. Improving performance of Devanagari script input-based P300 speller using deep learning. *IEEE Trans. Biomed. Eng.* **2018**, *66*, 2992–3005. [CrossRef] [PubMed]

5. Gao, W.; Guan, J.; Gao, J.; Zhou, D. Multi-ganglion ANN based feature learning with application to P300-BCI signal classification. *Biomed. Signal Process. Control* **2015**, *18*, 127–137. [CrossRef]

6. Wang, M.; Li, R.; Zhang, R.; Li, G.; Zhang, D. A wearable SSVEP-based BCI system for quadcopter control using head-mounted device. *IEEE Access* **2018**, *6*, 26789–26798. [CrossRef]

7. Kwak, N.S.; Müller, K.R.; Lee, S.W. A convolutional neural network for steady state visual evoked potential classification under ambulatory environment. *PLoS ONE* **2017**, *12*, e0172578 [CrossRef]

8. Lu, N.; Li, T.; Ren, X.; Miao, H.Y. A deep learning scheme for motor imagery classification based on restricted boltzmann machines. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2016**, *25*, 566–576. [CrossRef]

9. Ma, T.; Li, H.; Yang, H.; Lv, X.; Li, P.; Yao, D. The extraction of motion-onset VEP BCI features based on deep learning and compressed sensing. *J. Neurosci. Methods* **2017**, *275*, 80–92. [CrossRef]

10. Müller, K.R.; Tangermann, M.; Dornhege, G.; Krauledat,M.; Curio, G.; Blankertz, B. Machine learning for real-time single-trial EEG-analysis: From brain–computer interfacing to mental state monitoring. *J. Neurosci. Methods* **2008**, *167*, 82–90. [CrossRef]

11. Acharya, R.; Faust, O.; Kannathal, N.; Chua, T.; Laxminarayan, S. Non-linear analysis of EEG signals at various sleep stages. *Comput. Methods Programs Biomed.* **2005**, *80*, 37–45. [CrossRef] [PubMed]

12. Tang, X.; Li, W.; Li, X.; Ma, Z.; Dang, X. Motor imagery EEG recognition based on conditional optimization empirical mode decomposition and multi-scale convolutional neural network. *Expert Syst. Appl.* **2020**, *149*, 113285. [CrossRef]

13. Wu, S.L.; Wu, C.W.; Pal, N.R.; Chen, C.; Chen, S.; Lin, C. Common spatial pattern and linear discriminant analysis for motor imagery classification. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain, Singapore, 16–19 April 2013.

14. Liu, Z.; Lai, Z.; Ou, W.; Zhang, K.; Zheng, R. Structured optimal graph based sparse feature extraction for semi-supervised learning. *Signal Process.* **2020**, *170*, 107456. [CrossRef]

15. Ruan, J.; Wu, X.; Zhou, B.; Guo, X.; Lv, Z. An Automatic Channel Selection Approach for ICA-Based Motor Imagery Brain Computer Interface. *J. Med. Syst.* **2018**, *42*, 253. [CrossRef] [PubMed]

16. Sakhavi, S.; Guan, C.; Yan, S. Parallel convolutional-linear neural network for motor imagery classification. In Proceedings of the 2015 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 1–4 September 2015.

17. Dose, H.; Møller, J.S.; Iversen, H.K.; Puthusserypady, S. An end-to-end deep learning approach to MI-EEG signal classification for BCIs. *Expert Syst. Appl.* **2018**, *114*, 532–542. [CrossRef]

18. Sturm, I.; Lapuschkin, S.; Samek, W.; Müller, K.R. Interpretable deep neural networks for single-trial EEG classification. *J. Neurosci. Methods* **2016**, *274*, 141–145. [CrossRef]

19. Zhang, R.; Zong, Q.; Dou, L. A novel hybrid deep learning scheme for four-class motor imagery classification. *J. Neural Eng.* **2019**, *16*, 066004. [CrossRef]

20. Li, M.A.; Zhang, M.; Sun, Y.J. A novel motor imagery EEG recognition method based on deep learning. In Proceedings of the 2016 International Forum on Management, Education and Information Technology Application, Guangzhou, China, 30–31 January 2016.

21. Tang, Z.; Li, C.; Sun, S. Single-trial EEG classification of motor imagery using deep convolutional neural networks. *Opt.-Int. J. Light Electron Opt.* **2017**, *130*, 11–18. [CrossRef]

22. Tabar, Y.R.; Halici, U. A novel deep learning approach for classification of EEG motor imagery signals. *J. Neural Eng.* **2016**, *14*, 016003. [CrossRef]

23. Adeli, H.; Zhou, Z.; Dadmehr, N. Analysis of EEG records in an epileptic patient using wavelet transform. *J. Neurosci. Methods* **2003**, *123*, 69–87. [CrossRef]

24. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2013**, *521*, 436–444. [CrossRef] [PubMed]

25. Xiang, L.; Guo, G.; Yu, J.; Sheng, V.; Yang, P. A convolutional neural network-based linguistic steganalysis for synonym substitution steganography. *Math. Biosci. Eng.* **2020**, *17*, 1041–1058. [CrossRef]

26. Strubell, E.; Ganesh, A.; McCallum, A. Energy and policy considerations for deep learning in NLP. *arXiv* **2019**, arXiv:1906.02243.

27. González-Briones, A.; Villarrubia, G.; De Paz, J.F.; Corchado, J.M. A multi-agent system for the classification of gender and age from images. *Comput. Vis. Image Underst.* **2018**, *172*, 98–106. [CrossRef]

28. Zhang, D.; Yang, G.; Li, F.; Sangaiah, A.K. Detecting seam carved images using uniform local binary patterns. *Multimedia Tools Appl.* **2018**, *18*, 1–16. [CrossRef]

29. Dai, M.; Zheng, D.; Na, R.; Wang, S.; Zhang, S. EEG classification of motor imagery using a novel deep learning framework. *Sensors* **2019**, *19*, 551. [CrossRef]

30. Tavanaei, A.; Ghodrati, M.; Kheradpisheh, S.R.; Masquelier, T.; Maida, A. Deep learning in spiking neural networks. *Neural Netw.* **2019**, *111*, 47–63. [CrossRef]

31. Saxe, A.M.; Bansal, Y.; Dapello, J.; Kolchinsky, A.; Tracey, B.D. On the information bottleneck theory of deep learning. *J. Stat. Mech. Theory Exp.* **2019**, *2019*, 124020. [CrossRef]

32. Liu, M.; Wu, W.; Gu, Z. Deep learning based on Batch Normalization for P300 signal detection. *Neurocomputing* **2018**, *275*, 288–297. [CrossRef]

33. Maddula, R.; Stivers, J.; Mousavi, M.; Ravindran, S. Deep Recurrent Convolutional Neural Networks for Classifying P300 BCI signals. In Proceedings of the GBCIC, Graz, Austria, 18–22 September 2017.

34. Aznan, N.K.N.; Bonner, S.; Connolly, J.; Moubayed, N.; Breckon, T. On the classification of SSVEP-based dry-EEG signals via convolutional neural networks. In Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics, Miyazaki, Japan, 7–10 October 2018.

35. Kumar, S.; Sharma, A.; Mamun, K.; Tsunoda, T. A deep learning approach for motor imagery EEG signal classification. In Proceedings of the 2016 3rd Asia-Pacific World Congress on Computer Science and Engineering, Nadi, Fiji, 5–6 December 2016 .

36. Tayeb, Z.; Fedjaev, J.; Ghaboosi, N.; Richter, C.; Everding, L. Validating deep neural networks for online decoding of motor imagery movements from EEG signals. *Sensors* **2019**, *19*, 210. [CrossRef]

37. Li, M.; Zhu, W.; Zhang, M.; Sun, Y.; Wang, Z. The novel recognition method with optimal wavelet packet and LSTM based recurrent neural networkIn Proceedings of the IEEE International Conference on Mechatronics and Automation, Ningbo, China, 19–21 November 2017.

38. Pfurtscheller, G.; Neuper, C.; Flotzinger, D.; Pregenzer, M. EEG-based discrimination between imagination of right and left hand movement. *Electroencephalogr. Clin. Neurophysiol.* **1997**, *103*, 642–651. [CrossRef]

39. Sethi, S.; Upadhyay, R.; Singh, H.S. Stockwell-common spatial pattern technique for motor imagery-based Brain Computer Interface design. *Comput. Electr. Eng.* **2018**, *71*, 492–504. [CrossRef]

40. Qiu, Z.; Allison, B.Z.; Jin, J.; Zhang, Y.; Wang, X.; Li, W.; Cichocki, A. Optimized motor imagery paradigm based on imagining Chinese characters writing movement. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1009–1017. [CrossRef] [PubMed]

41. Cun, Y.L.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.E.; Jackel, L.D. Handwritten Digit Recognition with a Back-Propagation Network. *Adv. Neural Inf. Process. Syst.* **1997**, *2*, 396–404.

42. Hanin, B. Universal function approximation by deep neural nets with bounded width and relu activations. *Mathematics* **2019**, *7*, 992. [CrossRef]

43. Zhang, J.; Bargal, S.A.; Lin, Z.; Brandt, J.; Shen, X.; Sclaroff, S. Top-down neural attention by excitation backprop. *Int. J. Comput. Vis.* **2018**, *126*, 1084–1102. [CrossRef]

44. Sun, S.; Zhou, J. A review of adaptive feature extraction and classification methods for EEG-based brain-computer interfaces. In Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, 6–11 July 2014.

45. An, X.; Kuang, D.; Guo, X.; Zhao, Y. A deep learning method for classification of EEG data based on motor imagery. In Proceedings of the International Conference on Intelligent Computing, Taiyuan, China, 3–6 August 2014; pp. 203–210.

46. Blankertz, B.; Tomioka, R.; Lemm, S.; Kawanabe, M.; Muller, K.R. Optimizing Spatial filters for Robust EEG Single-Trial Analysis. *IEEE Signal Process. Mag.* **2008**, *25*, 41–56. [CrossRef]

47. Li, M.; Liu, Y.; Wu, Y.; Liu, S.; Jia, J.; Zhang, L. Neurophysiological substrates of stroke patients with motor imagery-based brain-computer interface training. *Int. J. Neurosci.* **2014**, *124*, 403–415.

48. Ono, T.; Kimura, A.; Ushiba, J. Daily training with realistic visual feedback improves reproducibility of event-related desynchronisation following hand motor imagery. *Clin. Neurophysiol.* **2013**, *124* 1779–1786. [CrossRef]

49. Hsu, W.Y.; Sun, Y.N. EEG-based motor imagery analysis using weighted wavelet transform features. *J. Neurosci. Methods* **2009**, *176*, 310–318. [CrossRef]

50. Ma, L.; Stückler, J.; Wu, T.; Cremers, D. Detailed Dense Inference with Convolutional Neural Networks via Discrete Wavelet Transform. *arXiv* **2018**, arXiv:1808.01834.