Biomedical Signal Processing and Control 55 (2020) 101645

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

P300 based character recognition using convolutional neural network and support vector machine

Sourav Kundu, Samit Ari*

Department of Electronics and Communication Engineering, National Institute of Technology Rourkela, Odisha 769008, India

ARTICLE INFO

Article history:
Received 22 December 2018
Received in revised form 10 July 2019
Accepted 7 August 2019
Available online 21 August 2019

Keywords:
Brain–computer interface (BCI)
Convolutional neural network (CNN)
Ensemble of support vector machines
Fisher ratio
P300 speller

ABSTRACT

In this work, a brain–computer interface (BCI) system for character recognition has been proposed based on the P300 signal. P300 signal classification is the most challenging task in electroencephalography signal processing as it is affected by the surrounding noise and low signal-to-noise ratio (SNR). Feature extraction and feature selection are essential steps for any classification task. Most of the earlier techniques reported hand-crafted features for detection of P300 signal. However, the hand-crafted features are not efficient to represent the signal properly due to surrounding environment and subject variability. Motivated by this, convolutional neural network (CNN) has been proposed for automatic high-level feature extraction to detect P300 signal. In general, CNN model consists of convolutional and fully-connected layers followed by a softmax layer. In the developed system, two different convolutional layers are used to extract the spatial and temporal features from the dataset. Also, a 2D convolutional layer based CNN architecture has been proposed where spatio-temporal feature is extracted in a single layer. To mitigate the over-fitting problem, dropout is employed in CNN architecture, which improves the network performance. After extracting high-level features, Fisher ratio (*F-ratio*) based feature selection is proposed to find the optimal features. The optimal features are used in the ensemble of support vector machine (ESVM) classifier for P300 detection. ESVM has been adopted in this work to minimize the classifier variability. The models are tested on two widely used datasets, and the experimental results show better or comparable performance compared to the earlier reported techniques.

1. Introduction

Brain–computer interface (BCI) is a technology that works as a bridge between human brain and computer. A patient with neuromuscular disease can use BCI system for their daily communication or rehabilitation [1]. A typical BCI system consists of a stimulus presentation paradigm, a signal acquisition system, a signal processing unit, and a controller to control the external device. Different types of electroencephalography (EEG) signals like P300, steady-state visually evoked potential (SSVEP), motor imagery (MI) are used to control the BCI system [2]. A P300 speller which enables the user to spell words using brain signal is discussed in this work. A P300 signal appears in the EEG when visual stimuli occur randomly. A positive deflection appears around 300 ms after the visual stimulus. In general, P300 signal is collected in non-invasive manner. EEG signal is influenced by environmental noise or other bio-signals. Repeated stimuli have been used for a single character, and the signals are averaged out to improve the signal-to-noise ratio (SNR).

BCI competitions have been conducted to evaluate different signal processing algorithms and encourage the BCI developers [3,4]. Feature extraction, feature selection and pattern classification are the important tasks in BCI system. For efficient classification, feature extraction is an essential step as more relevant feature provides better classification performance. Several feature extraction techniques like discrete wavelet transform (DWT) [5,6], higher order spectral regression discriminant analysis (HOSRDA) models [7], principal components analysis (PCA) [8] and independent components analysis (ICA) [9] have been reported for BCI application. For efficient P300 signal detection, different types of classifier like linear discriminant analysis (LDA), artificial neural network (ANN), support vector machine (SVM) are introduced in [5,8,10–12]. A data partitioning technique has been introduced to enhance the ensemble diversity of an ensemble of SVM (ESVM) in [13].

Over the past decade, deep learning techniques achieve impressive result in image processing and natural language processing [14–16]. In recent years, deep learning technique has been used in BCI applications [17–21]. In BCI systems, the high dimension of EEG data creates an over-fitting problem and irrelevant features reduce the classification performance. Therefore, to reduce the feature dimension, different types of channel selection algorithms are

* Corresponding author.
E-mail addresses: 514ec1002@nitrkl.ac.in (S. Kundu), samit@nitrkl.ac.in (S. Ari).

https://doi.org/10.1016/j.bspc.2019.101645

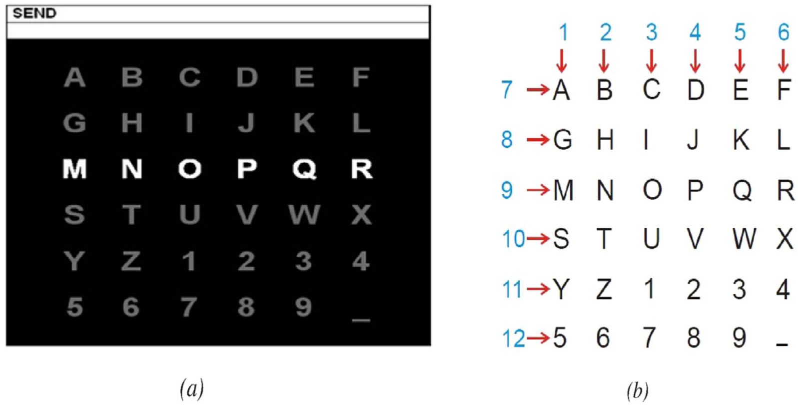

Fig. 1. The P300 speller paradigm [3]. (a) The 6 × 6 BCI paradigm for data collection, and (b) row/column information of the P300 speller paradigm.

proposed in [5,12,22]. The channel selection is a time consuming task at the time of training. It also depends on the number of channels present in the data acquisition system. The channel selection algorithm will not work well if the number of channels are less in the acquisition system. Thus, all the channels have been used in the reported literature [19]. Yoon and Kim have used raw samples and two morphological features for P300 classification [23]. PCA which projects the data linearly, is used for feature reduction in [8]. Conventional machine learning techniques use hand-crafted features which neither represent the signal properly nor address the non-linearity of the signal appropriately.

In BCI application, most of the researchers use SVM, ESVM, ensemble of weighted SVM (EWSVM) [8,10,12,13,24,25], Fisher's linear discriminant (FLD) or group-sparse Bayesian linear discriminant analysis (gsBLDA) [7,22] as a classifier. Ensemble of classifiers reduces the classifier variability and minimizes the error. Training time required for SVM is significant compared to FLD [13]. However, SVM with sufficient and well represented feature vectors can generate a good decision surface for classification and provides better performance compared to FLD. Deep learning techniques like convolutional neural network (CNN), multi-classifiers convolutional neural network (MCNN), batch normalized neural network (BN^3) and stacked autoencoder (SAE) [17–19] are used for P300 classification and the performance of these techniques are comparable to the other state-of-the-art techniques. In these deep learning methods, high-level features are extracted automatically depending on the subject and class, and classification is performed using a softmax classifier in the last layer of the network. Based on CNN and MCNN, seven different models are reported in [17], and among them, CNN-1 and MCNN-1 show better performance compared to other models.

To overcome the limitation of the hand-crafted features, deep learning based automated feature extraction technique has been proposed as it extracts subject and class dependent features [26]. In this work, CNN has been used for feature extraction as it learns a group of kernels which extract the information from the data, that are relevant to the problem. These kernels are fine-tuned by the back-propagation algorithm. The proposed CNN architecture consists of two convolutional layers (C1 and C2) and a fully-connected (FC) layer. C1 and C2 layers perform the convolution in the spatial and temporal domain, respectively. The system has been developed to extract spatio-temporal feature from the EEG signal. After the training of the CNN, features are extracted from the FC layer and, Fisher ratio (*F-ratio*) based feature selection has been applied on extracted features to find the optimal features. Then, the selected features are used for P300 detection. ESVM is used to get better classification performance. Before ensemble the score of SVM classifiers, the score of each SVM is normalized using median absolute deviation (MAD). The developed model is tested on two publicly accessible datasets as mentioned in [3,4]. The experimental results show better or comparable recognition performance compared to the previously reported methods.

The rest of the paper is structured as follows: the database is described in Section 2. In Section 3, the developed P300 speller system for the character recognition is described. The experimental results and discussions are presented in Section 4. Finally, the conclusion of the work is presented in Section 5.

2. Dataset

Dataset IIb of BCI Competition II [3] and dataset II of the BCI Competition III [4] have been used in this work. These datasets are provided by BCI Laboratory of the Wadsworth Center, NYS Department of Health. 42 training and 31 testing characters of a single subject are collected for BCI Competition II dataset. BCI Competition III contains the data of two subjects. There are 85 training and 100 testing characters for each subject. The data is collected using a P300 speller paradigm as shown in Fig. 1. Initially, this type of P300 speller paradigm for character recognition has been proposed by Farwell and Donchin [27]. Six rows and six columns of the BCI paradigm are flashed randomly at a rate of 5.7 Hz. The character matrix is intensified for 100 ms and remains blank for 75 ms. For one round, there are 12 flashing and out of which only 2 contains the desired character. These 12 flashing of one round is known as one epoch. Therefore, time required to collect each character data for one epoch is $(100 + 75) \times 12 = 2.1$ s. The sets of 12 intensifications have been repeated 15 times for each character and thus, there are $12 \times 15 = 180$ total intensifications for a single character. It means, each character data has been collected for 15 epochs. The performance of character recognition or predicted character is calculated after each epoch. There is a gap of 2.5 s after each character and during this time the speller matrix remains blank. Signals are collected from a data acquisition system with 64 channels. The signals are bandpass filtered from 0.1 to 60 Hz and digitized at 240 Hz.

3. Methods

The developed P300 speller system as shown in Fig. 2, consists of the following steps: preprocessing, feature extraction, feature selection and character recognition from the EEG signal.

3.1. Preprocessing

In P300 signal, a positive peak appears about 300 ms after the stimuli. It is assumed that a time window of 667 ms after the stimulus is sufficient to capture the necessary information about P300

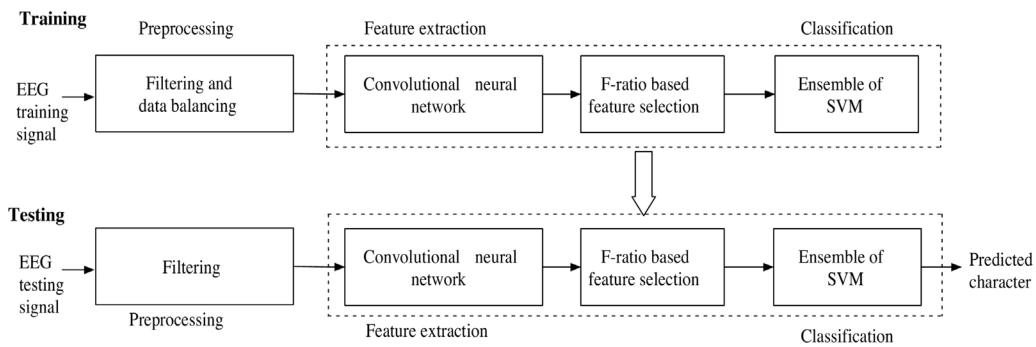

Fig. 2. Block diagram of the developed P300 speller system used for the character recognition.

signal [12]. In preprocessing stage, the EEG signal from each channel is filtered with an 8-order Chebyshev Type I bandpass filter of cut-off frequency 0.1–10 Hz. Data duration of 667 ms means 160 samples are taken from each channel. Therefore, the dimension of the single data sample is 64×160. Due to the P300 speller paradigm design, it generates two P300 and ten not-P300 signals in one round of flashing or epoch, as a result, the dataset is unbalanced. A classifier performs well when the training data is balanced. Otherwise, the classifier is biased towards the class with more number of samples. In the dataset, not-P300 signals are five times more than P300 signals. To balance the training dataset, it is divided into five parts and each part consists of all the P300 signals and one-fifth of not-P300 signals.

3.2. Convolutional neural network architecture for feature extraction

CNN is used for P300 classification as reported in the earlier literature [17–19]. In the developed model, mid-level CNN features are extracted and used in ESVM for P300 detection. CNN extracts hierarchical features from the data in each layer and these features are class dependent. Most of the cases, these automatically extracted features provide better results compared to the hand-crafted features. CNN uses weight-sharing and sparse connectivity between the layers, which improve the performance and reduce the computational cost. Three different combinations of the convolution layers have been proposed for P300 feature extraction in the developed system.

Combination 1: In this configuration, first convolution layer (C1) and second convolution layer (C2) have been used for convolution across the spatial and time domain, respectively as shown in Fig. 3. In C1 layer, data from all the channels are mixed together as the dimension of the convolutional kernel is the same as the number of channels. The output is the linear combination of all the channels. It is a channel mixing operation (CM). The kernel dimension of C1 layer is 64×1 with a stride of 1. The output data dimension of C1 layer is 1×160, which is the convolved output of all the channels. In C2 layer, the convolution has been performed across channel wise (CW). The kernel dimension is 1×20 with a stride of 20. For simplicity of the network, convolutional stride is used to reduce the feature dimension of the data [28]. There is no overlapping sample in the time domain. This stride acts as a down-sampler as no down-sampling has been applied at the time of preprocessing. After C2 layer, the dimension of output data is 1×8. A convolution layer contains a group of kernels and convolved output is referred as feature map (FM). Sixteen kernels have been used in each convolution layer. Therefore, the proposed CNN has sixteen FMs in both convolution layers. This network is referred as channel mixing-channel wise-CNN (CM-CW-CNN).

Combination 2: In this configuration, C1 and C2 convolution layers are interchanged. First, the convolution has been performed across channel wise and then, across the spatial domain. The output of first convolution layer is a 2D data as all the channels are present in the dataset. The output data dimension after first convolution layer is 64×8. Information loss occurs after first layer as a stride of 1×20 is used to reduce feature dimension. The output of the second layer is a single channel output, which is the convolved output of the channels obtained after the first layer. Similar to CM-CW-CNN, sixteen kernels have been used in each convolution layer. This network is referred as channel wise-channel mixing-CNN (CW-CM-CNN).

Combination 3: In this configuration, a single layer of 2D convolution has been used for feature extraction as shown in Fig. 4. The kernel dimension is 64×20 with a stride of 1×20. In a single layer both the spatial and temporal information are extracted from the EEG signal. Sixteen kernels are used in the convolution layer for feature extraction. After convolution layer, the dimension of the data is 1×8. This network is referred as 2D-CNN.

All the discussed combinations of CNN architectures have a set of convolution kernels in the convolutional layer to extract different local pattern from each region of the input space. This convolution operation is defined as follows:

$$Y_k = (W_k \otimes X) + b_k \tag{1}$$

where Y_k is the kth output FM of input X. W_k and b_k are the kth filter and bias, respectively and $\otimes$ represent the convolution operation. Let the input data dimension of the convolution layer is $[m \times n]$, convolutional kernel size is $[p \times q]$, stride is $[a \times b]$ and the output dimension is $[x \times y]$. Then x and y are calculated as follows:

$$x = \frac{m-p}{a} + 1, \quad y = \frac{n-q}{b} + 1 \tag{2}$$

In the developed model, the padding of the convolution is 'VALID' type. After the convolution layer, a FC layer and a softmax layer are used to complete the network. In the CNN architecture, batch normalization (BN) accelerates network training by reducing internal covariate shift [29]. The BN layer is used before C1 convolution layer and after C2 convolution layer. Rectified linear unit (ReLU) [30] is used as activation function after the second BN layer as it accelerates the training process by transferring the negative value of the neuron to zero. The ReLU function is defined as $ReLU(x) = max(x, 0)$. Another important operation in CNN is the dropout operation which is used to prevent the over-fitting of the network [31]. In dropout layer, some of the neurons are deactivated with the probability of p. In the developed CNN model, p is chosen as 0.5 which means at the time of training a neuron has a 50% chance that it will be dropped. The input size, kernel size, stride and feature maps of the CNN architecture mentioned in CM-CW-CNN is shown in Table 1. After tuning the parameters of the CNN architecture, features are extracted from the output of FC layer.

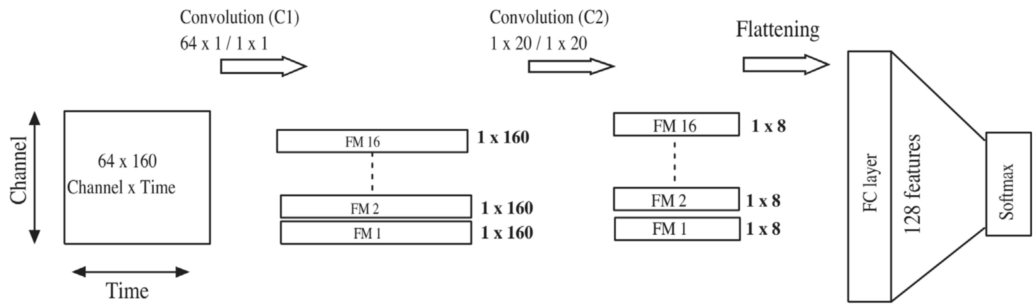

Fig. 3. Proposed CM-CW-CNN architecture for feature extraction. In C1 and C2 layer, convolution in channel and convolution in time are implemented, respectively. After the convolution layer, the feature maps are flattened into a vector and fed to the fully-connected layer.

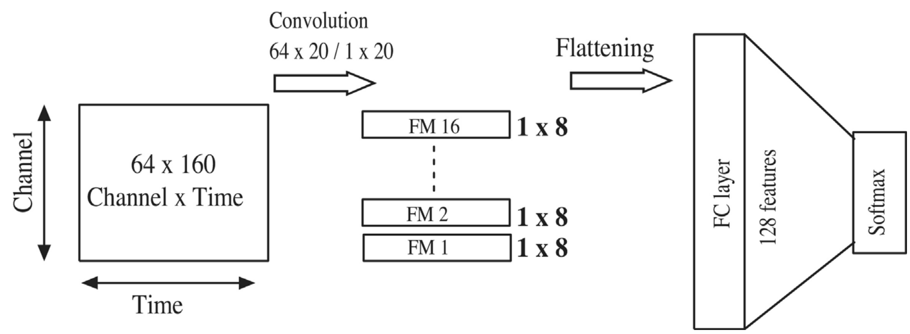

Fig. 4. Proposed 2D-CNN architecture for feature extraction. In a single layer both convolution in time and channel are performed. After the convolution layer, the feature maps are flattened into a vector and fed to the fully-connected layer.

Table 1
Proposed architecture of CM-CW-CNN model.

Layer type	Kernel size/stride	Input size	Feature map
Batch normalization layer (BN 1)	–	64×160	–
Convolution layer (C1)	$64 \times 1/1 \times 1$	64×160	16
Convolution layer (C2)	$1 \times 20/1 \times 20$	1×160	16
Batch normalization layer (BN 2)	–	1×8	–
ReLU layer	–	–	–
Dropout layer	–	–	–
Fully connected layer (FC)	–	128	–
Dropout layer	–	–	–
Softmax layer	–	128	–

3.3. Fisher ratio (F-ratio) based feature selection

Proper selection of feature is an important task for efficient classification. *F-ratio* based feature selection has been developed to extract accurate features for P300 detection. In this work, *F-ratio* technique with 10-fold cross-validation (CV) method has been proposed to find the optimal features from the CNN based extracted features. *F-ratio* [32] measures the variance of multi-class data. It is defined as

$$F\text{-}ratio = \frac{Variance \quad of \quad means \quad between \quad the \quad classes}{Average \quad variance \quad within \quad the \quad classes} \tag{3}$$

Let the number of classes present in the data is c, n_i is the number of data in ith class, μ_i represents the mean of ith class data, $\bar{\mu}$ is the mean of total data and x_{ij} is the jth data of ith class, then the Eq. (3) can be written as

$$F\text{-}ratio = \frac{(1/c)\sum_{i=1}^{c}(\mu_i - \bar{\mu})^2}{(1/c)\sum_{i=1}^{c}(1/n_i)\sum_{j=1}^{n_i}(x_{ij} - \mu_i)} \tag{4}$$

The *F-ratio* based feature selection technique involves the following steps:

- Calculate the *F-ratio* value of each feature using Eq. (4).
- Arrange the features in descending order according to their *F-ratio* value.
- Take the first feature and performs the 10-fold CV. Then, the classification performance accuracy is calculated as follows:

$$performance \quad accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

  where *TP*, *TN*, *FP* and *FN* stand for true positive, true negative, false positive, and false negative, respectively.
- Then, take the first and second feature, and perform the 10-fold CV. Calculate the classification performance accuracy.
- In the same way, use all the features and calculate the classification performance accuracy.
- Take the first few features for which the performance accuracy is maximum.

In order to speed up the process, group of eight features are concatenated together instead of single feature.

3.4. Ensemble of support vector machines (ESVM)

In the developed system, the training data has been divided into five parts as mentioned in preprocessing step. From each part, a CNN model is constructed and features are extracted from the FC layer. After *F-ratio* based feature selection, these selected features are used to train the SVM as shown in Fig. 5. In this work, ESVM is used for P300 classification. A good decision surface can be produced by the SVM if the input features are well organised [33]. Signal averaging is a well-known technique to cut down the variability of the signal and improve the SNR of EEG signal. In the constructed system, the outputs of the classifiers are averaged out to reduce the classifier variability, which will decrease the signal variability in another way. For testing, the test data is passed through each CNN model and the deep features are extracted after the FC layer. Afterwards, the features are selected based on *F-*

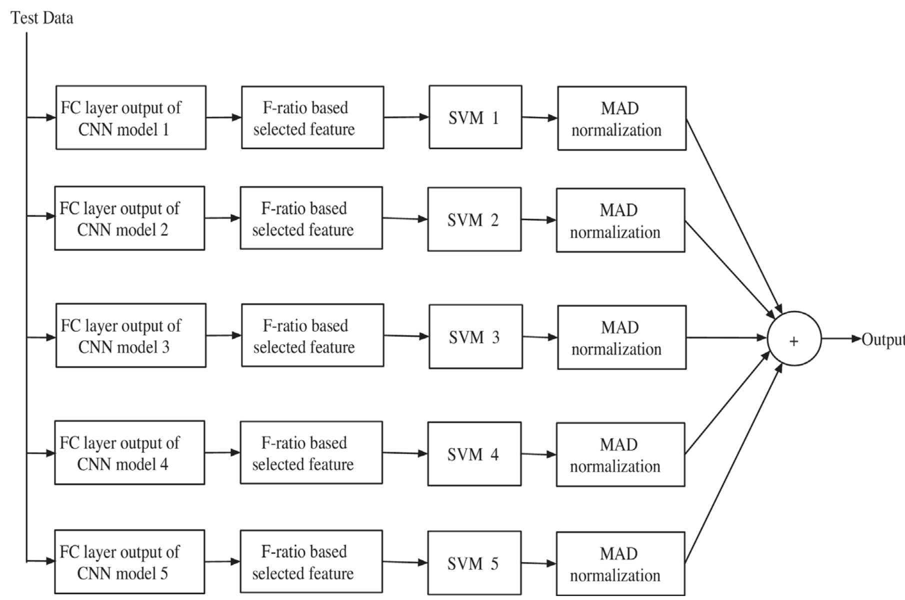

Fig. 5. Block diagram of the developed system: the developed system consists of five independent models. Here, the scores are normalized and ensembled together to predict the character.

ratio and these selected features are applied to the SVM classifier. Then, the output of each classifier is normalized according to MAD normalization technique, and these normalized scores are added together to predict a character as shown in Fig. 5. Assume, a score f_k is assigned by the kth classifier and therefore, the MAD normalization [34] is defined as follows:

$$f_{knorm} = \frac{f_k - median}{MAD} \tag{6}$$

where $MAD = median(|f_k - median|)$ and f_{knorm} is the normalized score. The MAD normalization technique is used as it is insensitive to the outliers. The averaged output score (S) is represented as

$$S = \frac{1}{J}\frac{1}{K}\sum_{j=1}^{J}\sum_{k=1}^{K} f_{knorm} \tag{7}$$

where J and K represent the number of epochs and classifiers, respectively. The row and column position are calculated from the score after every epoch as follows:

$$\begin{aligned} C_{pos} &= arg \max_{1 \le i \le 6} S(i) \\ R_{pos} &= arg \max_{7 \le i \le 12} S(i) \end{aligned} \tag{8}$$

where C_{pos} and R_{pos} are the predicted column and row of the speller matrix, respectively. After each epoch, the row and column are calculated from the output score of the classifier. The row and column intersection provides the desired character location.

3.5. Procedure

Three combinations of convolution networks have been used for feature extraction. CM-CW-CNN and CW-CM-CNN use two 1D convolution layers, which are applied to extract the spatial and temporal information separately from the dataset. On the other hand, a 2D convolution layer is applied to extract the spatio-temporal information from the data in the 2D-CNN. In the developed CM-CW-CNN and CW-CM-CNN models, 16 kernels of size [64×1] and [1×20] have been used in both C1 and C2 convolution layer, respectively for both the BCI Competition datasets. After the FC layer, the dimension of feature is 128. An optimal feature set is selected using *F-ratio* technique and this optimal feature set is used to train the SVM. For SVM, linear kernel is used and the regularization parameter (C) is selected as 0.1 empirically. In 2D-CNN model, 16 kernels of size [64×20] have been used for feature extraction. The SVM library is provided by Canu et al. [35] and the CNN model is implemented using Matlab.

4. Results and discussion

In this section, the experimental results of the proposed method are discussed. BCI Competition II and III datasets [3,4] are used in this work and the description of the datasets are provided in Section 2.

The character recognition performance of the proposed techniques is shown in Table 2 for BCI Competition III dataset. It has two subjects' data with 100 test characters each. Thus, the accuracy and the number of correctly recognised characters are same. It is observed from the table that CM-CW-CNN-ESVM achieves better result compared to the other methods. After 15 epoch, CM-CW-CNN-ESVM achieves an average accuracy of 99.0%, which is better than the other two developed networks. The performance of CW-CM-CNN-ESVM is less compared to CM-CW-CNN-ESVM as striding is performed in the first convolution layer, which causes information loss. 2D-CNN-ESVM uses a 2D convolution layer instead of two 1D convolution layers as applied in CM-CW-CNN-ESVM. Due to two 1D convolution layers in CM-CW-CNN-ESVM, there is flexibility in the network which increases the classification performance. In addition to three combinations of convolution layers with ESVM, the character recognition performance using ensemble of CNN is also shown in Table 2 for comparison purpose. From the table, it is observed that CM-CW-CNN-ESVM achieves better results compared to CNN as classifier. The proposed CM-CW-CNN-ESVM method extracts high-level features from the dataset using CNN, and *F-ratio* based optimal features are selected using 10 fold CV. Then, these features are used in SVM which is an efficient binary classifier. An optimal hyperplane is provided by the SVM for two class problem without any assumption of data distribution. After 15 epochs, the developed CM-CW-CNN-ESVM correctly recognizes 99 characters for both the subjects. The misclassified characters are shown in Table 3. From the table, it is observed that the misclassi-

Table 2
Character recognition performance of the proposed methods (in %) for BCI Competition III dataset.

Epochs Method	Subject	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CNN as classifier	A	22	34	55	57	61	71	75	76	83	85	86	89	90	95	98
	B	37	56	67	71	80	83	88	87	90	93	95	95	95	96	95
	Mean	29.5	45.0	61.0	64.0	70.5	77.0	81.5	81.5	86.5	89.0	90.5	92.0	92.5	95.5	96.5
CW-CM-CNN-ESVM	A	17	32	51	52	66	69	75	77	81	85	85	92	94	96	96
	B	43	59	64	72	75	82	86	85	92	93	93	93	92	92	94
	Mean	30.0	45.5	57.5	62.0	70.5	75.5	80.5	81.0	86.5	89.0	89.0	92.5	93.0	94.0	95.0
2D-CNN-ESVM	A	17	33	45	53	63	68	73	74	83	88	85	88	94	96	97
	B	42	60	64	68	75	80	81	86	91	94	94	94	92	92	95
	Mean	29.5	46.5	54.5	60.5	69.0	74.0	77.0	80.0	87.0	91.0	89.5	91.0	93.0	94.0	96.0
CM-CW-CNN-ESVM	A	22	32	55	59	64	70	74	78	81	86	86	90	91	94	99
	B	37	58	70	72	80	86	86	89	93	95	95	97	97	98	99
	Mean	29.5	45.0	62.5	65.5	72.0	78.0	80.0	83.5	87.0	90.5	90.5	93.5	94.0	96.0	99.0

Table 3
Confusion in character recognition for BCI Competition III dataset using proposed CM-CW-CNN-ESVM after 15 epoch.

Subject	Expected	output
A	Q	P
B	T	Z

fied characters belong to the same row or column of the expected characters. A performance comparison between proposed method and earlier reported techniques [7,8,12,13,17,18,22] is shown in Table 4. The result shows that the proposed CM-CW-CNN-ESVM method achieves better or competitive performance compared to the previous techniques.

A statistical comparison of the proposed CM-CW-CNN-ESVM method with earlier techniques for different epochs is shown in Table 5. The paired t-test result shows that the performance improvement between the proposed method and earlier reported techniques [8,12,17,18] are significant ($p < 0.05$).

The developed CM-CW-CNN-ESVM technique attains 100% accuracy at 4 epochs for BCI Competition II dataset, which includes eight test words (31 characters). Predicted characters after 1, 2, 3 and 4 epochs for the developed method is shown in Table 6. From Table 6, it is observed that most of the misclassified characters belong to the same row or column of the expected characters. After 1st epoch, 25 characters are predicted correctly. It means, the error rate of developed method is 19.35% after 1st epoch, which is a satisfying performance for building pragmatic character recognition system.

The performance comparison between the developed CM-CW-CNN-ESVM method with the earlier reported techniques [6,8,10,18] for BCI II dataset is shown in Table 7. From the table, it is observed that the constructed method achieves better or equal performance compared to other techniques. A statistical comparison between proposed CM-CW-CNN-ESVM method and previous methods [6,10,18] is shown in Table 8. The paired t-test result shows that the developed system attains significant improvement

Table 5
Paired t-test result of the proposed CM-CW-CNN-ESVM performance versus other literature methods performance for BCI Competition III dataset.

CM-CW-CNN-ESVM paired with	t stat	p-value
ESVM [12]	2.502	0.0204
CNN-1 [17]	4.148	0.0004
MCNN-1 [17]	3.449	0.0019
BN^3 [18]	2.063	0.0099
PCA-EWSVM [8]	3.506	0.0017

Table 6
The words predicted (with 1, 2, 3 and 4 epochs) and error (in %) of the developed CM-CW-CNN-ESVM technique for BCI Competition II dataset.

Epoch	Predicted words	Error
1	FOOD MOOZ HAM PIE CAHE TUHA ZNBOT 45T7	19.35%
2	FOOD MOOT HAM PIE CALE TUNA ZMAOT 4567	9.67%
3	FOOD MOOT HAM PIE CAKE TUNA ZMAOT 4567	6.45%
4	FOOD MOOT HAM PIE CAKE TUNA ZYGOT 4567	0.0%

Table 7
Performance comparison of the developed methods with the earlier methods in term of number of correctly recognized characters for BCI Competition II dataset.

Epochs Method	1	2	3	4	5	6
Vladimir [6]	20	26	29	30	30	31
Kaper et al. [10]	20	22	26	30	31	31
BN^3 [18]	24	23	27	28	29	30
PCA-EWSVM [8]	25	27	29	31	31	31
CM-CW-CNN-ESVM	25	28	29	31	31	31

Table 8
Paired t-test result of the developed method performance versus other literature methods performance for BCI Competition II dataset.

CM-CW-CNN-ESVM paired with	t stat	p-value
Vladimir [6]	1.718	0.053
Kaper et al. [10]	1.936	0.036
BN^3 [18]	2.873	0.006

Table 4
Character recognition comparison between the proposed methods and the earlier reported technique for BCI Competition III dataset.

Subject	Epoch	ESVM [12]	CNN-1 [17]	MCNN-1 [17]	HOSRDA &LDA [7]	Data partition – ESVM [13]	gsBLDA [22]	BN^3 [18]	PCA-EWSVM [8]	CM-CW-CNN-ESVM
A	10	83.0	86.0	82.0	84.0	85.0	88.0	86.0	82.0	86.0
	15	97.0	97.0	97.0	96.0	97.0	99.0	98.0	99.0	99.0
B	10	91.0	91.0	92.0	94.0	92.0	91.0	95.0	93.0	95.0
	15	96.0	92.0	94.0	97.0	95.0	95.0	95.0	97.0	99.0
Avg	10	87.0	88.5	87.0	89.0	88.5	89.5	90.5	87.5	90.5
	15	96.5	94.5	95.5	96.5	96.0	97.0	96.5	98.0	99.0

compared to the [10,18] as ($p < 0.05$). The improvement of the developed method is marginal ($p = 0.053$) compared to the [6].

Features are extracted manually in [7,8,10,12,13,22], whereas the proposed method extracts features automatically using CNN model and *F-ratio* based optimal features are selected using 10-fold CV. This automatic extracted features provide more information compared to the hand-crafted feature, which is the reason for the better performance of the developed system. A single classifier is used in [6,10], whereas the developed system uses ensemble of classifiers, which improves the classification performance. CNN is used for classification without BN and dropout layers in [17], which extracts the features automatically and classify the signal, whereas ESVM has been used for classification in the developed system, which is a strong classifier for binary classification. BN^3 [18] method uses batch normalization and dropout layer in CNN model, whereas the developed method extracts mid-level CNN feature and an optimal feature set is used in ESVM for classification.

5. Conclusion

In this paper, a novel method has been developed for the P300 speller system based on CNN and ESVM. The proposed method extracts high-level features from the data automatically using three combinations of CNNs, then *F-ratio* based feature selection has been proposed to select optimal feature and finally ESVM is used for character recognition. In the proposed CNN architecture both 1D CNN and 2D CNN are used for feature extraction. In case of 1D CNN, two CNN layers have been used to extract the spatial and temporal domain features from the dataset. In 2D CNN the features are extracted using a single convolution layer. SVM is used for classification as it provides better performance for binary classification when the features are well organised. ESVM is used to reduce the classifier variability. The performance of the proposed technique is validated on two publicly available datasets, BCI Competition II and BCI Competition III dataset, and compared with the state-of-the-art methods. The result shows that the developed CM-CW-CNN-ESVM architecture provides better or comparable performance compared to 2D-CNN-ESVM and other earlier reported methods. Therefore, the proposed model can be used as a general architecture for event-related potential classification.

Acknowledgments

This Publication is an outcome of the R&D work undertaken in the project under the Visvesvaraya Ph.D. Scheme of Ministry of Electronics and Information Technology, Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia) [grant number PhD-MLA/4(13)/2015-16].

Conflict of interest: None declared.

References

[1] N. Birbaumer, L.G. Cohen, Brain–computer interfaces: communication and restoration of movement in paralysis, J. Physiol. 579 (3) (2007) 621–636.
[2] M. Akcakaya, B. Peters, M. Moghadamfalahi, A.R. Mooney, U. Orhan, B. Oken, D. Erdogmus, M. Fried-Oken, Noninvasive brain–computer interfaces for augmentative and alternative communication, IEEE Rev. Biomed. Eng. 7 (2014) 31–49.
[3] B. Blankertz, K.-R. Muller, G. Curio, T.M. Vaughan, G. Schalk, J.R. Wolpaw, A. Schlogl, C. Neuper, G. Pfurtscheller, T. Hinterberger, et al., The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials, IEEE Trans. Biomed. Eng. 51 (6) (2004) 1044–1051.
[4] B. Blankertz, K.-R. Muller, D.J. Krusienski, G. Schalk, J.R. Wolpaw, A. Schlogl, G. Pfurtscheller, J.R. Millan, M. Schroder, N. Birbaumer, The BCI competition III: validating alternative approaches to actual BCI problems, IEEE Trans. Neural Syst. Rehabil. Eng. 14 (2) (2006) 153–159.
[5] M. Salvaris, F. Sepulveda, Wavelets and ensemble of FLDs for P300 classification, in: 2009 4th International IEEE/EMBS Conference on Neural Engineering, IEEE, 2009, pp. 339–342.
[6] V. Bostanov, BCI competition 2003-data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram, IEEE Trans. Biomed. Eng. 51 (6) (2004) 1057–1061.
[7] M.J. Idaji, M.B. Shamsollahi, S.H. Sardouie, Higher order spectral regression discriminant analysis (HOSRDA): a tensor feature reduction method for ERP detection, Pattern Recogn. 70 (2017) 152–162.
[8] S. Kundu, S. Ari, P300 detection with brain–computer interface application using PCA and ensemble of weighted SVMs, IETE J. Res. 64 (3) (2018) 406–414.
[9] N. Xu, X. Gao, B. Hong, X. Miao, S. Gao, F. Yang, BCI competition 2003-data set IIb: enhancing P300 wave detection using ICA-based subspace projections for BCI applications, IEEE Trans. Biomed. Eng. 51 (6) (2004) 1067–1072.
[10] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, H. Ritter, BCI competition 2003-data set IIb: support vector machines for the P300 speller paradigm, IEEE Trans. Biomed. Eng. 51 (6) (2004) 1073–1076.
[11] W. Gao, J.-A. Guan, J. Gao, D. Zhou, Multi-ganglion ANN based feature learning with application to P300-BCI signal classification, Biomed. Signal Process. Control 18 (2015) 127–137.
[12] A. Rakotomamonjy, V. Guigue, BCI competition III: dataset II-ensemble of SVMs for BCI P300 speller, IEEE Trans. Biomed. Eng. 55 (3) (2008) 1147–1154.
[13] Y.-R. Lee, H.-N. Kim, A data partitioning method for increasing ensemble diversity of an ESVM-based P300 speller, Biomed. Signal Process. Control 39 (2018) 53–63.
[14] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[15] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[16] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups, IEEE Signal Process. Mag. 29 (6) (2012) 82–97.
[17] H. Cecotti, A. Graser, Convolutional neural networks for P300 detection with application to brain–computer interfaces, IEEE Trans. Pattern Anal. Mach. Intell. 33 (3) (2011) 433–445.
[18] M. Liu, W. Wu, Z. Gu, Z. Yu, F. Qi, Y. Li, Deep learning based on Batch Normalization for P300 signal detection, Neurocomputing 275 (2018) 288–297.
[19] G.B. Kshirsagar, N.D. Londhe, Improving Performance of Devanagari Script input-based P300 speller using deep learning, IEEE Trans. Biomed. Eng. (2018) 1, http://dx.doi.org/10.1109/tbme.2018.2875024.
[20] Y.R. Tabar, U. Halici, A novel deep learning approach for classification of EEG motor imagery signals, J. Neural Eng. 14 (1) (2016) 016003.
[21] S. Sakhavi, C. Guan, S. Yan, Learning temporal information for brain–computer interface using convolutional neural networks, IEEE Trans. Neural Netw. Learn. Syst. 29 (11) (2018) 5619–5629.
[22] T. Yu, Z. Yu, Z. Gu, Y. Li, Grouped automatic relevance determination and its application in channel selection for P300 BCIs, IEEE Trans. Neural Syst. Rehabil. Eng. 23 (6) (2015) 1068–1077.
[23] K. Yoon, K. Kim, Multiple kernel learning based on three discriminant features for a P300 speller BCI, Neurocomputing 237 (2017) 133–144.
[24] Z. Gu, Z. Yu, Z. Shen, Y. Li, An online semi-supervised brain–computer interface, IEEE Trans. Biomed. Eng. 60 (9) (2013) 2614–2623.
[25] Y. Li, J. Pan, F. Wang, Z. Yu, A hybrid BCI system combining P300 and SSVEP and its application to wheelchair control, IEEE Trans. Biomed. Eng. 60 (11) (2013) 3156–3166.
[26] H.-J. Lee, K.-S. Hong, H. Kang, S. Lee, Photo aesthetics analysis via DCNN feature encoding, IEEE Trans. Multimedia 19 (8) (2017) 1921–1932.
[27] L.A. Farwell, E. Donchin, Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials, Electroencephalogr. Clin. Neurophysiol. 70 (6) (1988) 510–523.
[28] J.T. Springenberg, A. Dosovitskiy, T. Brox, M. Riedmiller, Striving for Simplicity: The All Convolutional Net, 2014 arXiv:1412.6806.
[29] S. Ioffe, C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, 2015 arXiv:1502.03167.
[30] A.L. Maas, A.Y. Hannun, A.Y. Ng, Rectifier nonlinearities improve neural network acoustic models, Proc. ICML, vol. 30 (2013) 3.
[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[32] J.P. Sahoo, S. Ari, D.K. Ghosh, Hand gesture recognition using DWT and F-ratio based feature descriptor, IET Image Process. 12 (10) (2018) 1780–1787.
[33] V. Vapnik, The Nature of Statistical Learning Theory, Springer Science & Business Media, 2013.
[34] S. Tulyakov, S. Jaeger, V. Govindaraju, D. Doermann, Review of classifier combination methods, in: Machine Learning in Document Analysis and Recognition, Springer, 2008, pp. 361–386.
[35] S. Canu, Y. Grandvalet, V. Guigue, A. Rakotomamonjy, SVM and Kernel Methods Matlab Toolbox, Perception Systèmes et Information, INSA de Rouen, Rouen, France, 2005.