Single Trial P300 Classification Using Convolutional LSTM and Deep Learning Ensembles Method

Raviraj Joshi[1(✉)], Purvi Goel[1], Mriganka Sur[2], and Hema A. Murthy[1]

[1] Indian Institute of Technology Madras, Chennai, India
{rjoshi,purvi,hema}@cse.iitm.ac.in

[2] Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, USA
msur@mit.edu

Abstract. The odd ball paradigm is a commonly used approach to develop Brain Computer Interfaces (BCIs). EEG signals have shown to elicit a positive deflection known as the P300 event related potential during odd ball experiments. BCIs based on these experiments rely on detection of the P300 potential. EEG signals are noisy, and therefore P300 detection is performed on an average of multiple trials, thus making them inappropriate for BCI applications. We propose a neural network model based on Convolutional Long Short Term Memory (ConvLSTM) for single trial P300 classification. EEG data encodes both spatial and temporal information using multiple EEG sensors. Convolutional neural networks (CNNs) have been known to capture spatial information whereas LSTMs are known to capture temporal information. Our experiments show that the proposed method outperforms previous CNN based approaches on raw EEG signals. The approaches were evaluated on publicly available dataset II of BCI competition III. Another dataset was recorded locally using audio beeps as stimuli to validate these approaches. The ensemble models based on CNNs and ConvLSTM are also proposed. These models perform better than individual architectures.

Keywords: Brain-computer interface · Event related potential P300 · Convolutional neural networks · Convolutional long short term memory

1 Introduction

Brain computer interfaces have made it possible to communicate our thoughts without vocalizing or performing actions. It relies on the neural activity of the brain to interface with the external world. BCIs are useful for the speech challenged since they enable such persons with a medium of communication. Electroencephalography (EEG) is a widely used technique to develop BCI applications. EEGs are non invasive and provides a comparatively low cost solution to build BCI devices.

U. S. Tiwary (Ed.): IHCI 2018, LNCS 11278, pp. 3–15, 2018.
https://doi.org/10.1007/978-3-030-04021-5_1

EEGs have a high temporal resolution as compared to other brain imaging techniques. It is therefore capable of capturing very short cognitive processes in the brain, which is elicited in the form of an Event Related Potential (ERP). ERPs are the positive or negative potentials seen in the EEG signals in response to external meaningful stimuli. These potentials are time locked and can be reliably detected to build practical BCI. However, EEG signals have low signal to noise ratio, which makes it difficult to detect these potentials in individual trials. Traditionally, multiple trials are averaged together in the time domain so that noise gets canceled and ERPs can be visualized. The focus of this work is the ERP elicited during odd ball experiment known as P300 event related potential and its single trial detection. In the odd ball experiment, a series of stimuli are presented to the subject. Two types of stimuli are presented, one which is frequent, while another that is infrequent. Infrequent stimuli cause ERP.

P300 signals have been mainly studied in the context of clinical diagnostics and psychoanalysis. Recently P300 is being considered as a possible BCI. A very popular program based on P300 is the P300 speller program [10]. P300 speller is a tool used to efficiently communicate symbols of interest, by arranging them in a 6×6 grid. There are other examples of P300 being used for cursor control [14] and wheelchair control [21].

Detecting these P300 potentials in single trials has gained a lot of attention. The speed of communication will greatly improve if P300 can be reliably detected from individual trials. Previous works have used conventional machine learning techniques and neural network based classifiers for P300 classification tasks. EEG data has both spatial and temporal characteristics. The spatial structure is attributed to different EEG sensors placed on the subject's scalp and the temporal structure is implicit as it picks up the voltage value at each time instant. Moreover, P300 has timing characteristics and has different signature across the spatial structure of the brain. Thus, the spatio-temporal information is associated with P300 potential as well. In this work, we propose a neural network model based on Convolutional Long Short Term Memory (ConvLSTM) that naturally exploits both spatial and temporal characteristics [24]. As P300 potential is seen around 300 ms after the onset of stimulus for a short duration, an LSTM based architecture was chosen to model this sequential information. The convolution units of ConvLSTM encodes the spatial information captured by electrodes placed on the scalp. This is the first work to employ ConvLSTM for EEG data analysis. ConvLSTM also requires relatively less number of parameters. Thus, ConvLSTM is easy to train given the scarcity of training data in EEG experiments.

We propose an architecture that uses a 3D representation of EEG data [6]. Although ConvLSTM has fewer parameters, it performs better than CNN architectures as it provides better modeling of EEG data. Finally, we evaluate the performance of ensemble architectures based on the individual models discussed in this paper. These ensemble models perform better than individual models. The results of these experiments are better than the previous benchmarks on publicly available BCI competition III dataset.

The rest of the paper is organized as follows: Sect. 2 describes related work done in this field. The experimental setup, pre-processing methods and the models used for P300 detection are explained in Sect. 3. The results are discussed in Sect. 4 and Sect. 5 is dedicated to Conclusion.

2 Related Work

Convolutional neural network (CNN) based approach for P300 detection was first proposed by Cecotti and Graser [7]. The proposed model with two convolutional and two fully connected layers could achieve comparable P300 detection results on raw EEG signals. The two convolutional layers were designed in such a way that the first layer captured the spatial information and the second layer captured the temporal information. They proposed seven CNN architectures including three ensemble models. This work was extended by Liu *et al.* to demonstrate the importance of batch normalization in the classification process [15]. They proposed BN3 which coupled the existing CNN models with batch normalization to prevent overfitting. BN3 could achieve the state of the art character recognition rates. In these works, the EEG data was represented in a 2D matrix of dimension channels $\times$ time, thus forming an image like structure. The convolution operations were performed along the spatial and temporal axis. A more relevant EEG data representation was proposed by Carabez *et al.* [6]. In their work, a single trial EEG was represented in 3 dimensions. The first two dimensions encode the spatial structure of EEG while the third dimension is the temporal axis. The work described in this paper uses the same technique to represent the EEG signals.

Recently, Bashivan *et al.* in their work used stacked convolutional and LSTM layers to process EEG data [5]. However, their work was based on spectral features extracted in theta, alpha, and beta frequency bands. For each time instant, EEG data was represented in (r,g,b) image like structure. The first two dimensions encoded the spatial structure of channels while the third dimension was reserved for the three frequency bands. Similar approach for EEG data representation was employed by Maddula *et al.* for classifying P300 signals [16]. They proposed different stacked CNN and LSTM architectures based on the same spectral features. Their results were evaluated on P300 speller experiment which was different from BCI competition III experiment and hence the results cannot be compared. Our work is different from these works as we do not use stacked architectures and our work is completely based on processing raw EEG signals. Various other studies of using recurrent neural networks on EEG data have been proposed over time [11,18,19], but these works fail to take advantage of spatial characteristics of EEG data.

An ensemble of multiple machine learning models has been commonly used in the classification of EEG signals [23]. Previously, to account for signal variability, an ensemble of 17 support vector machines was used by Rakotomamonjy and Guigue [20]. They also proposed a recursive channel elimination algorithm to select discriminative channels. Deep learning ensemble models based on CNNs were proposed in [4,7].

3 Methodology

Two different datasets are used in this work, one of which was collected in the EEG Lab set up at Computer Science and Engineering Department, IIT Madras and other was publicly available BCI Competition dataset [13].

3.1 BCI Competition Dataset

The dataset II from BCI competition III is used in our experiments. The data consists of P300 evoked potentials generated using the P300 Speller paradigm proposed by Farwell and Dochin [10].

The data was collected from 2 subjects in 3 sessions using a 64 channel EEG sensor net. The sampling rate was 240 Hz and the signals were bandpass filtered between 0.1 to 60 Hz. The odd ball paradigm results in unbalanced data for the two classes, so the target P300 trials are replicated four times so as to get an equal number of trials for both the classes. This is a classical oversampling approach to counter data imbalance problem [17]. The number of samples for both the classes during training and testing is shown in Table 1. A detailed description of the dataset can be found in [13].

Table 1. Training and testing trials

	Train		Test	
Subject	Target	Non-Target	Target	Non-Target
A	2550	12750	3000	15000
B	2550	12750	3000	15000

3.2 Odd Beep Experiment

The EEG data was collected using the Geodesic Sensor net comprising of 128 channels [2]. Subjects were students from different department in an age group of 20 to 30 years. A written consent was given by participants before the experiment. The subjects were seated in a comfortable chair to avoid fatigue inside an acoustically and electromagnetically shielded testing chamber. The subjects were asked to keep the eyes closed and do minimal muscle movement during the recording.

The subjects are presented with a series of audio beeps with two different frequencies. One of the beep also known as odd beep or target beep is presented less frequently than the other non target beeps. The frequencies for two beeps were chosen to be 1000 Hz (non target tone) and 2000 Hz (target tone). The participant was supposed to gently tap on the mobile screen when a rare target stimulus is presented and the other stimulus did not require any response. A train of 150 beeps is presented to the subject. These 150 beeps are divided across 25 trials. Each trial, therefore, consists of 6 beeps. Out of these 6 beeps, 1 of the beep is the target beep, while the others are non-target beeps. The target beep is chosen randomly. Four extra non-target beeps were added at the start of the experiment to make the subject familiar with non-target tone. The inter-stimulus gap is set to 800 ms. A baseline of 60 s is recorded at the start and end of the experiment. The experiment timeline is shown in Fig. 1.

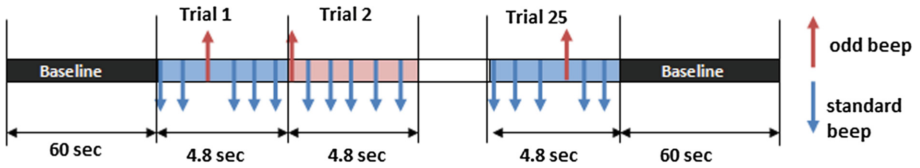


Fig. 1. Single Oddbeep experiment timeline

The performance of a neural network is defined by the availability of enough training samples. So multiple sessions of the experiment were performed on two subjects. For each subject recording was done on 5 consecutive days and 3 sessions were recorded on each day. A total of 15 sessions were recorded out of which 10 sessions were used for training and 5 sessions were used for testing. Table 2 shows the count of the target and non-target samples used for training and testing. Target trials were replicated to match the number of non-target trials. Although the number of samples is considerably less as compared to the online dataset, it is just enough to train our shallow neural networks.

Table 2. Training and testing trials for Single Odd Beep Dataset

	Train		Test	
Subject	Target	Non-Target	Target	Non-Target
A	250	1290	125	645
B	250	1290	125	645

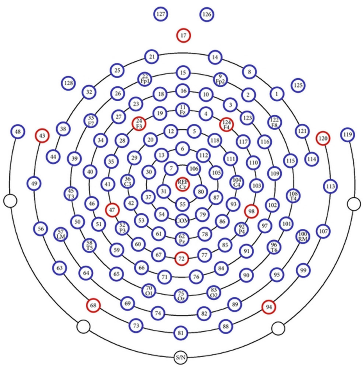
3.3 Pre-processing

BCI Competition Dataset. The EEG data is highly dominated by noise, so the data is bandpass filtered in the range 0.1–20 Hz. Moreover, the P300 wave is characterized by a low frequency component which is captured well by the filtered signal. The P300 potential is seen in EEG signals as a positive deflection around 300 ms after the onset of the stimulus, so we consider a 667 ms time window after the onset of the stimulus [15]. As the data is sampled at 240 Hz, 667 ms corresponds to 160 samples for each trial. The data segments are downsampled and the length of segment is reduced to 80 samples. Thus, each target and non-target trial is represented by a 64 × 80 matrix, where 64 defines the number of channels.

In order to retain the spatial correlation of the electrodes, the 64 electrodes are mapped to a 10 × 11 2D map as defined in [6]. Thus, each time instant is represented by a 2D matrix and each trial is now 3 dimensional. Each column vector of the 2D matrix in the original data corresponds to a 2D frame in the 3D map. The depth of the 3D map represents the time axis (80 in this case). This 3D representation, similar to video frames is given as an input to our models.

Single Odd Beep Dataset. The pre-processing steps used for single odd beep dataset are similar to the steps used for BCI competition dataset. Two major changes are induced by the fact that this data is sampled at 250 Hz and is captured using a 128 channel EEG net. The data is band passed filtered in the range 0.1–20 Hz and a window of 640 ms after the onset of stimulus is extracted for each trial. As the sampling rate is 250 Hz, 640 ms corresponds to 160 samples.

To preserve the spatial structure, the 128 electrodes are mapped to a 14×13 2D map. Some peripheral electrodes are dropped to create a dense matrix of size 10×11 as shown in Fig. 2. The 3-dimensional structure of electrodes is converted into a 2D map using software provided by EEGLAB [9].



(a) 128 channel EGI Sensor net

0	0	0	0	127	22	17	14	126	0	0	0	0
0	0	0	0	26	23	18	15	9	8	0	0	0
0	128	33	27	24	19	16	10	3	2	1	125	0
44	39	34	28	25	20	11	4	124	123	122	121	120
49	45	40	35	29	21	12	5	119	118	117	116	115
0	46	41	36	30	13	6	113	112	111	110	109	0
0	0	47	42	37	31	7	107	106	105	104	103	0
0	50	48	43	38	32	Cz	81	88	94	99	102	0
0	57	51	52	53	54	55	80	87	93	101	108	0
0	56	58	59	60	61	62	79	86	92	97	0	0
0	63	64	65	66	67	68	78	85	91	96	100	0
0	0	69	70	71	72	73	77	84	90	95	0	0
0	0	0	0	74	75	76	83	89	0	0	0	0
0	0	0	0	0	0	82	0	0	0	0	0	0

(b) Corresponding 2D Mapping

Fig. 2. Mapping of 128 electrodes to 2D matrix

4 Model Description

4.1 CNN Model for 3D Data (CNN-3D)

Convolutional neural networks have been widely used in classification tasks involving images and videos. They are good at capturing spatial information

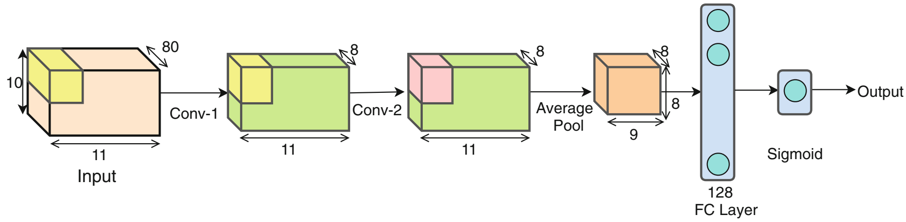


Fig. 3. Model 1: Architecture of CNN-3D

and can be used to process 3D data. CNNs are especially suited to EEG because of their translation invariance property. The EEG electrodes may be slightly misplaced across recordings. This shift of electrodes is easily accounted by convolution and pooling layers. Our model uses average pooling instead of the standard max pooling based on the fact that nearby electrodes have similar signatures. The model architecture is shown in Fig. 3 and the details of the proposed model are listed below

- The input data is batch normalized and passed to the next layer. The importance of batch normalization for EEG data is already studied in [15]. The batch size is set to 64 samples.
- The data of size $10 \times 11 \times 80$ is passed through a convolutional layer. This layer has a filter of size 3×3 which spans across the depth of input. Eight such filters are applied and appropriate padding is done to retain the input dimension. After this operation, the output is reduced to $10 \times 11 \times 8$ which is passed through *relu* non-linearity. This output is further batch normalized and a dropout of 0.2 probability is applied. The dropout operation is performed only during the training phase.
- The next layer performs a similar convolution operation with the same number of filters.
- The output is then average pooled with a filter of size 2×2 and a stride of 1. This reduces the output dimension to $8 \times 9 \times 8$. This is followed by a batch normalization layer and the 3D output is reshaped to a 1D vector.
- The flattened input is passed through a dense layer with 128 neurons and *sigmoid* activation. This is followed by a dropout of 0.2 probability.
- Since this is a binary classification problem, the final layer is a dense layer with one neuron and *sigmoid* activation. Binary cross entropy is used as a loss function with *Adam* optimizer [12]. Learning rate is kept at 0.001.

4.2 Convolutional Long Short Term Memory (ConvLSTM)

ConvLSTMs are similar to LSTMs with the dense operations replaced by convolution operations. As convolution operations use parameter sharing, the number of parameters is greatly reduced in ConvLSTM. Moreover, ConvLSTM allows

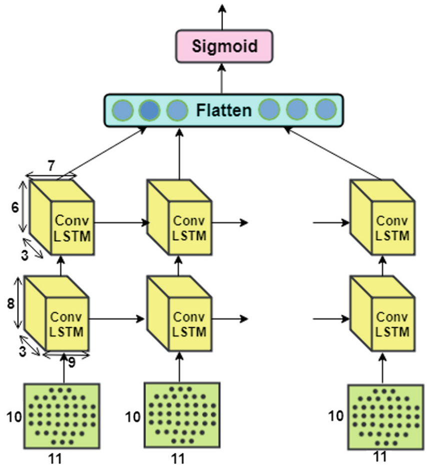


Fig. 4. Model 2: Architecture of ConvLstm

simultaneous learning of spatio-temporal features. Therefore, ConvLSTM is preferred over stacked convolution and LSTM architecture [22]. The 3D EEG data is passed as a sequence of 2D frames to the ConvLSTM. This is equivalent to passing a series of images to ConvLSTM. Although CNNs are capable of capturing temporal information, the number of parameters increases exponentially as the depth of input increases. This makes it difficult to capture long term relationships. We, therefore, explore the usage of ConvLSTM to model spatial and temporal structure of EEG. The placement of electrodes on the scalp, the mental state of the subject across sessions and trials can lead to variability. This is accounted by the convolutional operations of ConvLSTM. The temporal characteristics of EEG are captured by the LSTM structure. The model architecture is shown in Fig. 4 and the details of the proposed model is described below

– The input data is first batch normalized and passed to the next layer. The batch size is set to 64 samples.
– The sequence of inputs in the form of 2D map is passed through recurrent convolutions of filter size 3×3. Three such filters are used and the output is passed through *tanh* non-linearity. A dropout of 0.2 is used along with recurrent dropout of 0.1. This is followed by a batch normalization layer.
– A similar recurrent convolution is again applied on the output of the previous layer followed by batch normalization.
– The sequence returned is flattened and a dropout of 0.5 is applied. The final layer is a dense layer having only one neuron with *sigmoid* activation function. Binary cross entropy loss is used along with *Adam* optimizer for training. Learning rate is kept at 0.001.

4.3 Ensemble Models

The individual models proposed in this paper along with BN3 proposed in [15] were used to build ensemble classifiers. The individual classifiers were trained on the entire dataset and best models were picked using 10% validation data for ensemble testing. Scores from individual models were fused using simple averaging. Four ensemble models were evaluated using combinations of CNN-3D, BN3, and ConvLSTM.

5 Results and Discussion

The performance of single trial P300 detection is measured for BN3, CNN-3D, ConvLSTM and their ensembles. Several metrics that are used to evaluate the performance of the models are true positive (TP), false positive (FP), true negative (TN), false negative (FN) and others mentioned below

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Recognition} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \qquad \text{F-measure} = 2\frac{\text{Precision.Recall}}{\text{Precision} + \text{Recall}}$$

The number of trainable parameters for each of the models under consideration is shown in Table 3. Along with CNN-3D and ConvLSTM, the BN3 model is used in our analysis for comparison of the results [15]. ConvLSTM model has one-third parameters as compared to BN3 and CNN-3D. This allows the model to learn representations even with less data. These models were implemented using Keras library [8] with tensorflow backend [3]. AWS p2.xlarge instances were used for evaluation of models [1]. The results of BN3 were obtained under this environment, original results can be seen in [15]. Ensemble models use individual models having similar performance.

Table 3. Number of model parameters

BN3	CNN-3D	ConvLSTM
39,649	32,593	**11,213**

5.1 Analysis of BCI Competition Dataset

The complete results are shown in Table 4. Models CNN-1, MCNN-1 and MCNN-3 were proposed by Cecotti and Graser [7]. Their results are listed here for comparative analysis. CNN-3D gives better performance than the existing models as it takes into consideration the spatial positioning of electrodes. It classifies

Table 4. Performance of different models for Online Dataset.

Data Set	Model	TP	TN	FP	FN	Recog.	Recall	Precision	F1-score
III A	BN3 + CNN3D + CONVLSTM	1926	11,748	3252	1074	**0.7597**	0.6420	0.3720	**0.4710**
	CNN3D + CONVLSTM	1963	11,592	3408	1037	0.7531	0.6543	0.3655	0.4690
	BN3 + CONVLSTM	1919	11,701	3299	1081	0.7567	0.6397	0.3678	0.4670
	BN3 + CNN3D	1952	11,573	3427	1048	0.7513	0.6507	0.3629	0.4659
	CONVLSTM	1928	11,582	3418	1072	**0.7505**	0.6426	0.3606	**0.4620**
	CNN3D	1938	11,536	3464	1062	0.7486	0.6460	0.3588	0.4613
	BN3*	1910	11,229	3771	1090	0.7299	0.6367	0.3362	0.4400
	CNN-1	2021	10,645	4355	979	0.7037	0.6737	0.3170	0.4311
	MCNN-1	2071	10,348	4652	929	0.6899	0.6903	0.3080	0.4260
	MCNN-3	2023	10,645	4355	977	0.7038	0.6743	0.3172	0.4314
III B	BN3 + CNN3D + CONVLSTM	2003	12,810	2190	997	**0.8229**	0.6677	0.4777	**0.5569**
	CNN3D + CONVLSTM	2115	12,371	2629	885	0.8048	0.7050	0.4458	0.5462
	BN3 + CONVLSTM	1956	12,864	2136	1044	0.8233	0.6520	0.4780	0.5516
	BN3 + CNN3D	1990	12,762	2238	1010	0.8196	0.6633	0.4707	0.5506
	CONVLSTM	2099	12,330	2670	901	**0.8016**	0.6997	0.4401	**0.5404**
	CNN3D	2139	12,099	2901	861	0.7910	0.7130	0.4244	0.5321
	BN3*	2009	12,329	2671	991	0.7966	0.6697	0.4293	0.5232
	CNN-1	2035	12,039	2961	965	0.7819	0.6783	0.4073	0.5090
	MCNN-1	2202	11,453	3547	798	0.6899	0.7340	0.3830	0.5034
	MCNN-3	2077	11,997	3003	923	0.7038	0.6923	0.4089	0.5141

the examples of both the classes more accurately than the existing models. The proposed CNN-3D model has a high recognition rate as well as high precision and recall.

Results show that ConvLSTM models the spatio-temporal information well and performs better than other architectures. Table 4 shows that CNN-3D performs well for the target class whereas ConvLSTM performs better for the non-target class. Therefore, in order to retain the properties of both the models,

Table 5. Performance of different models for Single Odd Beep Dataset.

Data Set	Model	TP	TN	FP	FN	Recog.	Recall	Precision	F1-score
A	**BN3 + CNN3D + CONVLSTM**	88	603	42	37	0.8974	0.7040	0.6769	0.6902
	CNN3D + CONVLSTM	90	599	46	35	0.8948	0.7200	0.6618	0.6897
	BN3 + CONVLSTM	84	599	46	41	0.8870	0.6720	0.6462	0.6588
	BN3 + CNN3D	88	604	41	37	**0.8987**	0.7040	0.6822	**0.6929**
	CONVLSTM	82	596	49	43	0.8805	0.6560	0.6259	0.6406
	CNN3D	91	595	50	34	0.8909	0.7280	0.6454	0.6842
	BN3	84	588	57	41	0.8727	0.6720	0.5957	0.6316
B	**BN3 + CNN3D + CONVLSTM**	74	606	39	51	0.8831	0.5920	0.6549	0.6218
	CNN3D + CONVLSTM	74	592	53	51	0.8649	0.5920	0.5827	0.5873
	BN3 + CONVLSTM	75	607	38	50	**0.8857**	0.6000	0.6637	**0.6302**
	BN3 + CNN3D	77	598	47	48	0.8766	0.6160	0.6210	0.6185
	CONVLSTM	72	590	55	53	0.8597	0.5760	0.5669	0.5714
	CNN3D	78	573	72	47	0.8454	0.6240	0.5200	0.5673
	BN3	74	572	73	51	0.8389	0.5920	0.5034	0.5441

we propose ensemble models consisting of the existing BN3 architecture, the proposed CNN-3D, and ConvLSTM architectures.

The ensemble models consisting of BN3 + CNN-3D architectures and BN3 + ConvLSTM architectures have a very high recognition rate since they classify the non-target class examples with a very high precision whereas the model consisting of CNN-3D + ConvLSTM architectures have a very high recall.

5.2 Analysis of Single Odd Beep Dataset

Table 5 shows the results for different models. The results are similar to previous analysis, CNN-3D and ConvLSTM based models perform better than BN3. However, the recognition rates for odd beep dataset is better than BCI competition dataset. This can be due to the better spatial resolution provided by 128 channel EEG net. Although its scalability to more number of trials still needs to be validated.

6 Conclusions

The results obtained in this work show that the 3D input for single trial P300 classification works better than the 2D input. The reason can attributed to the fact that 3D input capture spatial surroundings of the data very well. CNNs based on 3D input captures the information well but may lack the temporal information. Spatio-temporal characteristics of EEG are captured well by ConvLSTM, and hence give better results using less parameters. Ensemble approaches that take into account the properties of both models perform significantly better.

References

1. Amazon ec2 - p2 instances. https://aws.amazon.com/ec2/instance-types/p2/. Accessed 09 Jan 2018
2. The geodesic sensor net. https://www.egi.com/research-division/geodesic-sensor-net. Accessed 9 Jan 2018
3. Abadi, M., et al.: Tensorflow: a system for large-scale machine learning. OSDI **16**, 265–283 (2016)
4. Barsim, K.S., Zheng, W., Yang, B.: Ensemble learning to EEG-based brain computer interfaces with applications on P300-spellers
5. Bashivan, P., Rish, I., Yeasin, M., Codella, N.: Learning representations from EEG with deep recurrent-convolutional neural networks (2015). arXiv preprint. arXiv:1511.06448
6. Carabez, E., Sugi, M., Nambu, I., Wada, Y.: Convolutional neural networks with 3D input for P300 identification in auditory brain-computer interfaces. Comput. Intell. Neurosci. **2017** (2017)
7. Cecotti, H., Graser, A.: Convolutional neural networks for P300 detection with application to brain-computer interfaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**(3), 433–445 (2011)
8. Chollet, F., et al.: Keras (2015). https://github.com/fchollet/keras
9. Delorme, A., Makeig, S.: Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. J. Neurosci. Methods **134**(1), 9–21 (2004)
10. Farwell, L.A., Donchin, E.: Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalogr. Clin. Neurophysiol. **70**(6), 510–523 (1988)
11. Fedjaev, J.: Decoding EEG brain signals using recurrent neural networks (2017)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014). arXiv preprint arXiv:1412.6980
13. Krusienski, D., Schalk, G.: Wadsworth BCI dataset (P300 evoked potentials), BCI competition III challenge (2004). http://www.bbci.de/competition/iii/
14. Li, Y., Long, J., Yu, T., Yu, Z., Wang, C., Zhang, H., Guan, C.: An EEG-based BCI system for 2-D cursor control by combining mu/beta rhythm and P300 potential. IEEE Trans. Biomed. Eng. **57**(10), 2495–2505 (2010)
15. Liu, M., Wu, W., Gu, Z., Yu, Z., Qi, F., Li, Y.: Deep learning based on batch normalization for P300 signal detection. Neurocomputing **275**, 288–297 (2018)
16. Maddula, R., Stivers, J., Mousavi, M., Ravindran, S., de Sa, V.: Deep recurrent convolutional neural networks for classifying P300 BCI signals. In: Proceedings of the Graz BCI Conference (2017)
17. Mazurowski, M.A., Habas, P.A., Zurada, J.M., Lo, J.Y., Baker, J.A., Tourassi, G.D.: Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance. Neural Netw. **21**(2–3), 427–436 (2008)
18. Naderi, M.A., Mahdavi-Nasab, H.: Analysis and classification of EEG signals using spectral analysis and recurrent neural networks. In: 17th Iranian Conference of Biomedical Engineering (ICBME), pp. 1–4. IEEE (2010)
19. Petrosian, A., Prokhorov, D., Homan, R., Dasheiff, R., Wunsch II, D.: Recurrent neural network based prediction of epileptic seizures in intra-and extracranial EEG. Neurocomputing **30**(1–4), 201–218 (2000)

20. Rakotomamonjy, A., Guigue, V.: BCI competition III: dataset II-ensemble of SVMs for BCI P300 speller. IEEE Trans. Biomed. Eng. **55**(3), 1147–1154 (2008)
21. Rebsamen, B., Guan, C., Zhang, H., Wang, C., Teo, C., Ang, M.H., Burdet, E.: A brain controlled wheelchair to navigate in familiar environments. IEEE Trans. Neural Syst. Rehabil. Eng. **18**(6), 590–598 (2010)
22. Sainath, T.N., Vinyals, O., Senior, A., Sak, H.: Convolutional, long short-term memory, fully connected deep neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4580–4584. IEEE (2015)
23. Sun, S., Zhang, C., Zhang, D.: An experimental evaluation of ensemble methods for eeg signal classification. Pattern Recognit. Lett. **28**(15), 2157–2163 (2007)
24. Xingjian, S., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.c.: Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: Advances in neural information processing systems, pp. 802–810 (2015)