



Universiteit
Leiden
The Netherlands

Towards high performance and efficient brain computer interface character speller : convolutional neural network based methods

Shan, H.

Citation

Shan, H. (2020, February 25). *Towards high performance and efficient brain computer interface character speller : convolutional neural network based methods*. Retrieved from <https://hdl.handle.net/1887/85675>

Version: Publisher's Version

License: [Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden](#)

Downloaded from: <https://hdl.handle.net/1887/85675>

Note: To cite this publication please use the final published version (if applicable).

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/85675> holds various files of this Leiden University dissertation.

Author: Shan, H.

Title: Towards high performance and efficient brain computer interface character speller : convolutional neural network based methods

Issue Date: 2020-02-25

Towards High Performance and Efficient Brain Computer Interface Character Speller: Convolutional Neural Network based Methods

Hongchang Shan

**Towards High Performance and Efficient
Brain Computer Interface Character Speller:
Convolutional Neural Network based Methods**

PROEFSCHRIFT

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus Prof.mr. C.J.J.M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op woensdag 25 februari 2020
klokke 16:15 uur

door

Hongchang Shan
geboren te Heilongjiang, China
in 1989

Promotors:	Dr. Todor P. Stefanov Prof. Dr. Aske Plaat	Universiteit Leiden Universiteit Leiden
Promotion Committee:	Prof. Dr. Tom Heskes Prof. Dr. Shaowei Cai Prof. Dr. Holger Hoos Prof. Dr. Fons Verbeek Dr. Wojtek Kowalczyk	Radboud Universitet Chinese Academy of Sciences Universiteit Leiden Universiteit Leiden Universiteit Leiden

Towards High Performance and Efficient
Brain Computer Interface Character Speller:
Convolutional Neural Network based Methods
Hongchang Shan. -
Dissertation Universiteit Leiden. - With ref. - With summary in Dutch.

Copyright © 2020 by Hongchang Shan. All rights reserved. No part of this thesis may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without prior permission from the author.

This dissertation was typeset using L^AT_EX in Linux and version controlled using Git.

Contents

Contents	v
List of Tables	ix
List of Figures	xiii
List of Abbreviations	xv
1 Introduction	1
1.1 Development Trends in P300-based Brain Computer Interface Systems	2
1.1.1 High Performance P300-based Brain Computer Interface Systems	2
1.1.2 Efficient P300-based Brain Computer Interface Systems	4
1.2 Problem Statement	6
1.2.1 Problem 1	7
1.2.2 Problem 2	8
1.3 Research Contributions	9
1.4 Dissertation Outline	13
2 Background	15
2.1 Machine Learning	15
2.2 Neural Network	17
2.2.1 Neurons	17
2.2.2 The Architecture of a Neural Network	19
2.2.3 Learning Process of a Neural Network	22
2.3 Convolutional Neural Network	27
2.3.1 The Convolution Operation	29
2.3.2 The Characteristics of Convolutional Neural Network	29
2.3.3 The Architecture of Convolutional Neural Network	30
2.4 P300-based Brain Computer Interface	31

2.4.1	P300 Signal	31
2.4.2	P300 Speller	33
2.4.3	Performance Assessment of P300 Speller	34
2.5	Datasets	35
3	A Simple Convolutional Neural Network for P300 Signal Detection and Character Spelling	39
3.1	Related Work	41
3.2	Proposed Convolutional Neural Network	44
3.2.1	Input to the Network	44
3.2.2	Network Architecture	45
3.2.3	Training	47
3.3	Experimental Evaluation	48
3.3.1	Experimental Setup	48
3.3.2	Complexity	48
3.3.3	P300 Signal Detection Accuracy	49
3.3.4	Character Spelling Accuracy	50
3.3.5	Information Transfer Rate	52
3.4	Conclusions	55
4	Ensemble of Convolutional Neural Networks for P300 Signal Detection and Character Spelling	57
4.1	Proposed Network	59
4.1.1	Ensemble of Convolutional Neural Networks	59
4.1.2	Proposed OSLN and OTLN	59
4.1.3	Training	61
4.1.4	P300 Signal Detection and Character Spelling using EoCNN	61
4.2	Experimental Evaluation	62
4.2.1	Complexity	63
4.2.2	P300 Signal Detection Accuracy	63
4.2.3	Character Spelling Accuracy	64
4.2.4	Information Transfer Rate	67
4.3	Discussions	69
4.3.1	Analysis of Our Proposed OTLN and OSLN	69
4.3.2	Ablation Study on EoCNN	71
4.3.3	Exploration on the Importance of Extracting P300-related Features from Raw Signals	72
4.4	Conclusions	73

5 A Novel Sensor Selection Method based on Convolutional Neural Network for P300 Speller	75
5.1 Related Work	77
5.2 Our Sensor Selection Method	78
5.2.1 Spatial Learning based Elimination Selection	78
5.2.2 Parameterized OSLN	79
5.2.3 Ranking Function	81
5.3 Experimental Evaluation	82
5.3.1 Experimental Setup	82
5.3.2 Experimental Results	84
5.4 Discussions	85
5.4.1 Configuration of E_s in SLES	86
5.4.2 Exploring the Impact of the CNN Architecture on Sensor Selection	88
5.5 Conclusions	92
6 An Improved Ensemble of Convolutional Neural Networks for P300 Speller with a Small Number of Sensors	95
6.1 Study on EoCNN-based P300 Speller with Different Number of Sensors	97
6.1.1 Experimental Setup	97
6.1.2 Experimental Results	98
6.2 Our Solution Approach	100
6.2.1 Parameterized Ensemble Processing	100
6.2.2 Parameter Configuration for Parameterized Ensemble Processing	101
6.3 Experimental Evaluation	103
6.3.1 Experimental Setup	103
6.3.2 Experimental Results	105
6.4 Conclusions	108
7 Summary and Conclusions	109
Bibliography	113
List of Publications	121
Samenvatting	123
Acknowledgments	127
Curriculum Vitae	129

List of Tables

2.1	Number of P300s/non-P300s for each dataset.	37
3.1	CCNN architecture.	43
3.2	BN3 architecture.	43
3.3	CNN-R architecture.	44
3.4	OCLNN architecture.	46
3.5	Complexity comparison of different CNNs.	49
3.6	P300 signal detection accuracy of different CNNs on Dataset II, III-A and III-B.	50
3.7	Spelling accuracy achieved by different methods on Dataset II.	51
3.8	Spelling accuracy achieved by different methods on Dataset III-A. .	51
3.9	Spelling accuracy achieved by different methods on Dataset III-B. .	51
3.10	Spelling accuracy achieved by OCLNN when using and not using the Batch Normalization operation on Dataset II.	53
3.11	Spelling accuracy achieved by OCLNN when using and not using the Batch Normalization operation on Dataset III-A.	53
3.12	Spelling accuracy achieved by OCLNN when using and not using the Batch Normalization operation on Dataset III-B.	53
3.13	The ITR of the P300 speller based on different methods on Dataset II.	54
3.14	The ITR of the P300 speller based on different methods on Dataset III-A.	54
3.15	The ITR of the P300 speller based on different methods on Dataset III-B.	54
4.1	OSLN architecture.	60
4.2	OTLN architecture.	60
4.3	Complexity of different CNNs.	63
4.4	P300 signal detection accuracy of different CNNs on Dataset II, III-A, and III-B.	64

4.5	Spelling accuracy achieved by different methods on Dataset II.	65
4.6	Spelling accuracy achieved by different methods on Dataset III-A.	65
4.7	Spelling accuracy achieved by different methods on Dataset III-B.	66
4.8	The ITR of the P300 speller based on different methods on Dataset II.	67
4.9	The ITR of the P300 speller based on different methods on Dataset III-A.	67
4.10	The ITR of the P300 speller based on different methods on Dataset III-B.	68
4.11	Spelling accuracy achieved by OTLN, OSLN and EoCNN on Dataset III-A.	70
4.12	Spelling accuracy achieved by EoCNN after removing a separate CNN.	71
5.1	The symbols used in Algorithm 1.	79
5.2	<i>OSLN_(S)</i> architecture.	81
5.3	Methods compared with SLES.	83
5.4	Minimal number of sensors selected by different methods for Dataset II. The P300 speller is implemented using the CNN-based classifier OCLNN.	85
5.5	Minimal number of sensors selected by different methods for Dataset III-A. The P300 speller is implemented using the CNN-based classifier OCLNN.	86
5.6	Minimal number of sensors selected by different methods for Dataset III-B. The P300 speller is implemented using the CNN-based classifier OCLNN.	87
5.7	Minimal number of sensors selected by different methods for Dataset II. The P300 speller is implemented using the CNN-based classifier EoCNN.	88
5.8	Minimal number of sensors selected by different methods for Dataset III-A. The P300 speller is implemented using the CNN-based classifier EoCNN.	89
5.9	Minimal number of sensors selected by different methods for Dataset III-B. The P300 speller is implemented using the CNN-based classifier EoCNN.	90
5.10	Minimal number of sensors selected by different methods for Dataset II, The P300 speller is implemented using the SVM-based classifier ESVM [RG08].	91
5.11	Minimal number of sensors selected by different methods for Dataset III-A, The P300 speller is implemented using the SVM-based classifier ESVM [RG08].	92

5.12	Minimal number of sensors selected by different methods for Dataset III-B, The P300 speller is implemented using the SVM-based classifier ESVM [RG08].	93
5.13	Minimal number of sensors selected by SLES with different E_s configurations.	94
5.14	Minimal number of sensors selected by analysing different CNNs.	94
6.1	Minimal number of sensors needed to acquire EEG signals in the P300 speller based on different CNNs without losing the state-of-the-art spelling accuracy of the P300 speller on Dataset II.	105
6.2	Minimal number of sensors needed to acquire EEG signals in the P300 speller based on different CNNs without losing the state-of-the-art spelling accuracy of the P300 speller on Dataset III-A.	106
6.3	Minimal number of sensors needed to acquire EEG signals in the P300 speller based on different CNNs without losing the state-of-the-art spelling accuracy of the P300 speller on Dataset III-B.	106
6.4	Minimal number of sensors needed to acquire EEG signals in the P300 speller based on different CNNs without losing the state-of-the-art max-ITR on Dataset II, III-A, and III-B.	107

List of Figures

1.1	Workflow of a typical BCI.	1
1.2	An example of a traditional P300-based BCI system.	5
1.3	An example of an efficient P300-based BCI system.	5
2.1	The workflow of machine learning.	17
2.2	The model of a neuron.	18
2.3	Architectural graph to model a neuron.	20
2.4	An example of a single-layer neural network.	21
2.5	An example of a multi-layer neural network with one hidden layer and one output layer.	21
2.6	An example of a cost function C with two parameters v_1 and v_2	23
2.7	The analogy of using gradient descent to minimize a cost function.	24
2.8	An example of the architecture of a CNN used for the handwritten digit recognition.	31
2.9	P300 signal.	32
2.10	P300 speller character matrix.	33
2.11	An example of a set of signal samples, where F_s is the signal sampling frequency	36
3.1	Abstraction of the raw signals in the spatial convolution layer in current CNNs. x denotes a signal sample in the input tensor. f denotes a datum in a feature map. Every column in the input tensor contains a set of C signal samples. These samples come from C sensor at a certain sampling time point. The spatial convolution operation converts each column of spatial data (receptive field) from the input tensor into an abstract datum in a feature map.	42
3.2	Input tensor for our proposed OCLNN.	45
3.3	Illustration of OCLNN for P300 signal detection.	46
4.1	Workflow of our EoCNN	59

4.2	Spelling accuracy achieved by OTLN, OTLN-31 and OTLN-61 on Dataset III-A.	70
4.3	Spelling accuracy achieved by networks in set RAW_networks and networks in set unRAW_networks on Dataset III-A.	72
5.1	Input tensor to $OSLN_{(S)}$, where $s_j \in S$	80
6.1	Spelling accuracy of different P300 speller implementations when different number of sensors m is used to acquire EEG signals.	99
6.2	max-ITR of different P300 speller implementations when different number of sensors m is used to acquire EEG signals.	99
7.1	Overview of how each chapter's contributions improve the performance and/or the efficiency of a P300 speller.	110

List of Abbreviations

ALS	Amyotrophic Lateral Sclerosis
AP	Action Potential
AUC	Area Under the Receiver Operating Characteristic
BCI	Brain Computer Interface
CNN	Convolutional Neural Network
CWT	Continuous Wavelet Transform
DWT	Discrete Wavelet Transform
ECoG	Electrocorticography
EEG	Electroencephalography
EoCNN	Ensemble of Convolutional Neural Networks
ERD	Event Related Desynchronization
ERP	Event-Related Potential
FLD	Fisher's Linear Discriminants
ITR	Information Transfer Rate
LDA	Linear Discriminant Analysis
LFP	Local Field Potential
max-ITR	maximum ITR
MSE	Mean Squared Error
NN	Neural Network
OCLNN	One Convolution Layer Neural Network
OSLN	One Spatial Layer Network
OTLN	One Temporal Layer Network
PEoCNN	EoCNN with parameterized ensemble processing
ReLU	Rectified Linear Unit

SAE	Stacked Autoencoder
SGD	Stochastic Gradient Descent
SLES	Spatial Learning based Elimination Selection
SMAC	Sequential Model-based Algorithm Configuration
SNR	Signal to Noise Ratio
SSNR	Signal to Signal and Noise Ratio
SSVEP	Steady State Visual Evoked Potential
SVM	Support Vector Machine
SWLDA	Stepwise Linear Discriminant Analysis
II	BCI Competition II - Data set IIb
III-A	BCI Competition III - Data set II Subject A
III-B	BCI Competition III - Data set II Subject B