

Cortically Coupled Image Computing

Zhengwei Wang

B.Eng., M.Sc.

A dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy (Ph.D.)

to the



Insight Centre for Data Analytics

School of Computing

Dublin City University

Advisors: Prof. Tomás E. Ward and Dr. Graham Healy

September 2019

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: _____ (Candidate) ID No: 18210204 Date: _____

致我的父母

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisors Prof. Tomás E. Ward and Dr. Graham Healy. I am fortunate to have these two extraordinary advisors during my PhD. Thanks for your consistent support, understanding, technical guidance and encouragement through these four years. I met Tomás when I was a final year undergraduate and he was my supervisor for the final year project. Given this chance, I had an opportunity to do this project in the area of neuroscience. He introduced lots of neuroscientific concepts to me, which I found quite hard to understand but very appealing to me at that time, and that was my first time hearing the terminology “brain-computer interface” in my life, which sowed a seed of doing research in this area in my mind. Thank you Tomás for bringing me over to this exciting area. At the beginning of my research, I was overwhelmed by sophisticated explanation and concepts in neuroscience and everything was new to me when reading papers with my limited neuroscience background. Tomás always sat down with me and we went through papers line by line together. Looking back my PhD, Tomás constantly encouraged me when I was frustrated and was open-minded to the research that I chose to explore. He is my mentor but more like a friend, giving me encouragement when I was frustrate, providing me valuable opinion when my paper was rejected, and giving me the bravery to try new ideas. I am lucky to have Tomás on board throughout years of my PhD. Graham has expertise in neuroscience field. He gave me tremendous help and guidance for my research through my PhD. When recording EEG signals from participants, I was inexperienced at scratching between electrodes and participants’ scalps as I was worried about hurting people. Graham taught me step by step and shared experience on doing this kind of scratch. While I was in my third year, I struggled to look for my PhD direction and Graham introduced generative adversarial networks to me, which brought me over to this exciting field. He suggested me to interact between deep neural networks and brain-computer interfaces. I had little experience on deep learning at that time and no one did research in that area before but I decided to have a try because of his encouragement and my own interest. We discussed lots of interesting problems, exchanged our ideas and inspired thinking to each other. I really enjoy working with him in the past four years. I appreciate everything that is given, taught and contributed by my two advisors for my PhD.

I also wish to thank Prof. Alan F. Smeaton for his numerous valuable suggestions on papers and rebuttals. I have known Dr. Qi She for over ten years since my secondary school and I am very excited that we have an opportunity to collaborate in research. I appreciate inspirations and knowledge in the area of machine learning and deep learning from Qi. He is not only my collaborator but also a soul mate.

I spent first two and half years of my PhD at Maynooth University. I would like to express my gratitude to all members in the Department of Electronic Engineering at Maynooth University for producing a warm working atmosphere. In particular I would like to thank John Maloco for his technical help throughout the research and Joanne Bredin and Ann Dempsey for always being at the end of the email when needed.

I would also like to thank all of my wonderful friends and colleagues in both Dublin City

Acknowledgments

University and Maynooth University as well as the entire Insight DCU family for their constant support. Jose Juan Dominguez Veiga, Damien Kearney, Eoin Brophy, Lili Zhang and all of my friends, thanks for making a lovely atmosphere in which work, and fun, came easily. I really enjoy talking and discussing with you during my PhD.

In the last four years, I was supported by Insight Centre for Data Analytics which is sponsored by Science Foundation Ireland under Grant Number SFI/12/RC/2289. Their generous support is also highly acknowledged.

My parents always encourage me to chase my dreams and give me endless support for my life and study. They cared about developing my hobbies and interests since I was a very young child. They gave me enough freedoms to learn anything I want. I was interested in Chinese Chess since I was very young and they let me learn it without any hesitation even the Chinese Chess is much less popular than Go in China. I have made great achievements in Chinese Chess and this hobby also develops my logic and strategic thinking skills, which has been utilized during my study. I appreciate their love and endless support throughout my life. I spent lots of my childhood with my grandparents and they taught me how to become a decent person. Thanks for your education and I will remember the words you have told me. To Anqi Hu, my girl friend, it is my pleasure to meet you in Ireland and thanks for your support, love and devotion throughout last few years.

Last but not the least, I would like to express my gratitude to those who I have met along the way and helped me in both study and life.

List of Publications

The following are **journal/book chapter** papers that have been **submitted/published** during the course of my PhD.

- **Z. Wang**, Q. She and T. E. Ward. “Generative Adversarial Networks: A Survey and Taxonomy,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, June 2019. [Submitted]
- **Z. Wang**, Q. She, A. F. Smeaton, T. E. Ward and G. Healy. “Neuroscore: Using A Neuro-AI Interface for Evaluating Generative Adversarial Networks,” *Neurocomputing*, June 2019. [Submitted]
- **Z. Wang**, G. Healy, A. F. Smeaton and T. E. Ward. “Use of Neural Signals to Evaluate the Quality of Generative Adversarial Network Performance in Facial Image Generation,” *Cognitive Computation*, Aug 2019. [Published]
- **Z. Wang**, G. Healy, A. F. Smeaton and T. E. Ward. “Spatial Filtering Pipeline Evaluation of Cortically Coupled Computer Vision System for Rapid Serial Visual Presentation,” *Brain-Computer Interfaces*, vol. 5(4), pp. 132-145, Jan 2019. [Published]
- **Z. Wang**, G. Healy, A. F. Smeaton and T. E. Ward. “A Review of Feature Extraction and Classification Algorithms for Image RSVP based BCI,” in *Signal Processing and Machine Learning for Brain-machine Interfaces*, pp. 243-270, The Institute of Engineering and Technology. Michael Faraday House, Six Hills Way, Stevenage, SG1 2AY, UK, 2018. [Published]

The following are **conference** papers that have been published during the course of my PhD.

- E. Brophy, J. J. Dominguez, **Z. Wang**, A. F. Smeaton and T. E. Ward. “An Interpretable Machine Vision Approach to Human Activity Recognition using Photoplethysmograph Sensor Data”. *Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, Ireland, 2018.
- E. Brophy, J. J. Dominguez, **Z. Wang** and T. E. Ward. “A Machine Vision Approach to Human Activity Recognition using Photoplethysmograph Sensor Data”. *29th Irish Signals and Systems Conference (ISSC)*, UK, 2018.
- Y. Wang, **Z. Wang**, W. Clifford, C. Markham, T. E. Ward and C. Deegan. “Validation of Low-cost Wireless EEG System for Measuring Event-related Potentials”. *29th Irish Signals and Systems Conference (ISSC)*, UK, 2018.

- **Z. Wang**, G. Healy, A. F. Smeaton and T. E. Ward. “An Investigation of Triggering Approaches for the Rapid Serial Visual Presentation Paradigm in Brain Computer Interfacing”. *27th Irish Signals and Systems Conference (ISSC)*, UK, 2016.

The following are **workshop papers/manuscripts** that have been published during the course of my PhD.

- E. Brophy, **Z. Wang** and T. E. Ward. “Quick and Easy Time Series Generation with Established Image-based GANs”. *arXiv preprint arXiv:1902.05624*, 2019.
- G. Healy, **Z. Wang**, C. Gurrin, T. E. Ward and A. F. Smeaton. “An EEG Image-search Dataset: A First-of-its-kind in IR/IIR. NAILS: Neurally Augmented Image Labelling Strategies”. In *Proceedings of CHIIR Workshop on Challenges in Bringing Neuroscience to Research in Human-Information Interaction*, Oslo, Norway, March 2017.

Contents

Declaration	ii
Acknowledgments	iv
List of Publications	vi
List of Abbreviations	xi
List of Figures	xiv
List of Tables	xvii
Abstract	xviii
1 Introduction	1
1.1 Brain-computer Interfaces and Event-related Potentials	2
1.2 Rapid Serial Visual Presentation Paradigm	5
1.3 Cortically Coupled Computer Vision System	7
1.4 Generative Adversarial Networks	8
1.5 Neuro-AI Interfaces	10
1.6 Research Motivation and Contributions	12
1.7 Research Questions	14
1.8 Overview and Organization of the Thesis	16
2 Data Description	19
2.1 Introduction	19
2.2 Neurally Augmented Image Labelling Strategies	21
2.3 Neural Indices for Face Perception Analysis	24
2.4 Conclusion	26
3 Cortically Coupled Computer Vision Processing Methods	27
3.1 Introduction	27
3.2 Overview of RSVP Experiments and EEG Data	30
3.2.1 RSVP Experiment for EEG Data Acquisition	30
3.2.2 Brief Introduction to RSVP-EEG Pattern	31
3.2.3 RSVP-EEG Data Pre-processing and Properties	33
3.2.4 Performance Evaluation Metrics	35
3.3 Feature Extraction Methods Used in CCCV Research	35
3.3.1 Spatial Filtering	36
3.3.2 Time-frequency Representation	41

3.3.3	Other Feature Extraction Methods	44
3.3.4	Summary	44
3.4	Survey of Classifiers Used in CCCV Research	44
3.4.1	Linear Classifiers	45
3.4.2	Artificial Neural Networks	49
3.5	Conclusion	54
4	Spatial Filtering Pipelines for Cortically Coupled Image Classification	56
4.1	Introduction	57
4.2	Methodology	59
4.2.1	Pipeline Description	59
4.2.2	Supervised Spatial Filtering	60
4.2.3	Feature Generation	65
4.2.4	Linear Classifiers	65
4.2.5	Evaluation	65
4.3	Results	67
4.3.1	Impact of Number of Spatial Filters	67
4.3.2	Performance Evaluation	69
4.3.3	Source Reconstruction	70
4.4	Discussion	75
4.5	Conclusion	79
5	Generative Adversarial Networks: A Survey and Taxonomy	81
5.1	Introduction	82
5.2	Search Strategy and Results	83
5.3	Previous Related Literature Reviews on GANs	86
5.4	Generative Adversarial Networks	86
5.5	Architecture-variant GANs	87
5.5.1	Fully-connected GAN (FCGAN)	87
5.5.2	Laplacian Pyramid of Adversarial Networks (LAPGAN)	88
5.5.3	Deep Convolutional GAN (DCGAN)	88
5.5.4	Boundary Equilibrium GAN (BEGAN)	89
5.5.5	Progressive GAN (PROGAN)	90
5.5.6	Self-attention GAN (SAGAN)	91
5.5.7	BigGAN	91
5.5.8	Summary	92
5.6	Loss-variant GANs	94
5.6.1	Wasserstein GAN (WGAN)	97
5.6.2	WGAN-GP	98
5.6.3	Least Square GAN (LSGAN)	99
5.6.4	f-GAN	101
5.6.5	Unrolled GAN (UGAN)	101
5.6.6	Loss Sensitive GAN (LS-GAN)	102
5.6.7	Mode Regularized GAN (MRGAN)	104
5.6.8	Geometric GAN	104
5.6.9	Relativistic GAN (RGAN)	104
5.6.10	Spectral Normalization GAN (SN-GAN)	106
5.6.11	Summary	107

5.7	Discussion	111
5.7.1	Interconnections between Architecture and Loss	112
5.7.2	Future Directions	112
5.8	Conclusion	112
6	Use of Neural Signals to Evaluate GANs	114
6.1	Introduction	114
6.2	Related Work	117
6.3	Methodology	119
6.3.1	P300 Reconstruction	119
6.3.2	Neuroscore	120
6.4	Results	120
6.4.1	Behavioral Task Performance	120
6.4.2	Rapid Serial Visual Presentation Task Performance	122
6.4.3	Comparison to Other Evaluation Metrics	127
6.5	Discussion	128
6.6	Conclusion	130
7	Pseudo Neuroscore: Using A Neuro-AI Interface for Evaluating GANs	131
7.1	Introduction	131
7.2	Related Work	133
7.3	Methodology	135
7.3.1	Neuro-AI Interface	135
7.3.2	Training Details	136
7.4	Results	139
7.4.1	EEG Improves Model Performance	139
7.4.2	Neuroscore Aligns with Human Perceptions	142
7.4.3	Neuroscore Needs Much Smaller Samples	142
7.4.4	Neuroscore Can Rank Images	143
7.4.5	Generalization of Neuroscore	144
7.5	Conclusion	145
8	Conclusion	146
8.1	Summary	146
8.2	Stepping into ERP Research	148
8.3	Stepping into Computational Neuroscience	149
8.4	Stepping into Neuro-AI Interface	150
	Bibliography	152
	Appendix	176
A	Investigation of Triggering Issue	176
B	Supplementary Tables	182
C	Notation on Chapter 5	183
C.1	Lipschitz Continuity	183
C.2	Matrix Norm	183
D	Supplementary Figures	184
E	NAILS Ethic Approval	187
F	NIFPA Ethic Approval	188

List of Abbreviations

AI	Artificial Intelligence
ANNs	Artificial Neural Networks
AUC	Area Under the Curve
BA	Balanced Accuracy
BCIs	Brain-computer Interfaces
BEGAN	Boundary Equilibrium GAN
BLR	Bayesian Linear Regression
BNNs	Biological Neural Networks
CAR	Common Average Reference
CCCV	Cortically Coupled Computer Vision
CNNs	Convolutional Neural Networks
CSP	Common Spatial Pattern
DBN	Deep Belief Nets
DCGAN	Deep Convolutional GAN
DGMs	Deep Generative Models
DNNs	Deep Neural Networks
EEG	Electroencephalography
ERPs	Event-related Potentials
ERSP	Event-related Spectral Perturbation
FCGAN	Fully-connected GAN
FID	Fréchet Inception Distance
FPR	False Positive Rate
GANs	Generative Adversarial Networks
GLMs	Generalized Linear Models
JS	Jensen-Shannon Divergence
IC	Independent Component
ICA	Independent Component Analysis

IS	Inception Score
ITC	Inter-trial Coherence
KL	Kullback-Leibler Divergence
LAPGAN	Laplacian Pyramid of Adversarial Networks
LDA	Linear Discriminant Analysis
LDR	Light Diode Resistor
LDRCC	Light Diode Resistor Comparator Circuit
LR	Logistic Regression
LSGAN	Least Square GAN
LS-GAN	Loss Sensitive GAN
LSL	Lab Streaming Layer
MLP	Multi-layer Perception
MMD	Kernel Maximum Mean Discrepancy
MRGAN	Mode Regularized GAN
MTWLB	Multiple Time Window LDA Beamformers
NAILS	Neurally Augmented Image labelling Strategies
NIFPA	Neural Indices For Face Perception Analysis
PCA	Principle Component Analysis
PROGAN	Progressive GAN
RBM	Restricted Boltzmann Machine
RGAN	Relativistic GAN
RNNs	Recurrent Neural Networks
ROC	Receiver Operating Characteristic
RSVP	Rapid Serial Visual Presentation
SAGAN	Self-attention GAN
SNAP	Simulation and Neuroscience Application Platform
SN-GAN	Spectral Normalization GAN
SNR	Signal-to-noise Ratio
SSNR	Signal-to-signal-plus Noise Ratio
STFT	Short Time Fourier Transform
SVM	Support Vector Machine
TPR	True Positive Rate
UGAN	Unrolled GAN
VAE	Variational Autoencoder

List of Abbreviations

WGAN	Wasserstein GAN
WGAN-GP	WGAN Gradient Penalty

List of Figures

1.1	An overview of a generic BCI system.	3
1.2	Three RSVP modes	6
1.3	An example of CCCV system.	7
1.4	Architecture of a GAN.	9
1.5	Schematic of action potentials and postsynaptic potential.	10
1.6	Comparison between a biological neuron and an artificial neuron.	11
1.7	The word cloud of keywords presented in this thesis.	16
2.1	Electrode locations of 10-20 system used through this thesis.	20
2.2	Examples of target images in the NAILS task.	21
2.3	ERP butterfly plot example in the NAILS dataset.	22
2.4	ICs of the NAILS EEG dataset.	23
2.5	An example of eye-related artifacts present in EEG.	23
2.6	ERP butterfly plot example for BE task in the NIFPA dataset.	25
2.7	ERP butterfly plot example for RSVP task in the NIFPA dataset.	25
2.8	ICA components of the NIFPA EEG dataset.	26
3.1	RSVP paradigm protocol.	28
3.2	Block diagram of a typical BCI system.	29
3.3	RSVP experiment set up.	31
3.4	The P300 response example at the Pz channel.	32
3.5	Examples of ICA components (left) and ERP images (right).	40
3.6	Example of ERSP representation.	43
3.7	Projection of two different classes (with equal covariance) onto a line by LDA .	46
3.8	An example of a MLP architecture.	50
3.9	An example of CNN architecture for EEG classification.	51
4.1	Two spatial filtering pipelines for RSVP-based EEG.	60
4.2	Spatial pattern estimation for LDA beamformer using whole EEG epoch via training data using CAR: <i>Participant 2</i>	62
4.3	ERPs topography plot.	67
4.4	Example of estimated spatial patterns/filters for three spatial filtering approaches.	68
4.5	Spatial patterns topographical plots produced by xDAWN and MTWLB.	71
4.6	Time-course source N170 and P300 reconstructed by xDAWN and MTWLB. .	72
4.7	SNR for reconstructed N170 (top) and P300 (bottom).	73
4.8	AUC score of using the source signal (N170 top and P300 bottom) reconstructed by xDAWN and MTWLB.	74
4.9	Correlation analysis between SNR and AUC.	76

5.1	Number of papers in each year from 2014 to 17th May 2019.	84
5.2	Categories of papers from 2014 to 17th May 2019.	84
5.3	Percentages of each category take account the total number of papers in each year.	85
5.4	Timeline of architecture-variant GANs.	87
5.5	Up-sampling process of generator in LAPGAN (from right to left).	88
5.6	Detail of DCGAN architecture for generator.	89
5.7	Illustration of BEGAN architecture.	89
5.8	Progressive growing step for PROGAN during the training process.	90
5.9	Self-attention mechanism architecture proposed in the paper.	91
5.10	Summary of recent architecture-variant GANs for solving the three challenges.	92
5.11	Illustration of training progress for a GAN.	95
5.12	JS divergence and gradient change with the distance between p_r and p_g	96
5.13	Comparison of parameter distribution between WGAN and WGAN-GP.	99
5.14	Decision boundary illustration of original GAN and LSGAN.	100
5.15	An example of computation for an unrolled GAN with 3 unrolling steps.	102
5.16	Demonstration of the loss in equation (5.18) for LS-GAN.	103
5.17	SVM hyperplane used in Geometric GAN.	105
5.18	D output comparison between RGAN and original GAN.	106
5.19	Current loss-variants for solving the challenges.	107
5.20	Loss and gradient for the generator of different loss-variant GANs.	108
6.1	Schematic of the first type of neuro-AI interface.	116
6.2	Face image examples used in the experiment.	121
6.3	Reconstructed averaged (via LDA beamformer) P300 signal across 12 participants.	123
6.4	Averaged P300 topography of each participant for each category.	124
6.5	Box plot of Neuroscore for each image category.	125
6.6	Correlation between Neuroscore and BE accuracy.	126
7.1	Schematic of different types of recorded neural signals.	133
7.2	Schematic of the second type of neuro-AI interface.	135
7.3	A neuro-AI interface and training details with adding EEG information.	136
7.4	Architecture of Shallow network used in this work.	137
7.5	Testing error of 3 models with and without EEG.	140
7.6	Scatter plot on the testing set of predicted and real Neuroscore of 6 models with and without EEG for training.	141
7.7	Neuroscore of different evaluated sample size for each type of GAN.	143
7.8	P300 amplitude predicted by proposed framework for each single image.	144
7.9	Generalization performance of the proposed framework for testing images.	145
A.1	LDRCC architecture.	176
A.2	Captured hardware and software triggers.	177
A.3	Histograms of latencies derived from (paired) differences between hardware and software trigger timestamps.	179
A.4	Distribution of interval differences in timestamps for hardware triggers (in blue) and software triggers (in orange).	180
D.1	Correlation between Neuroscore and BE accuracy with normalization (including RFACE category).	184
D.2	Correlation between Neuroscore and BE accuracy without normalization.	184

D.3	Correlation between Neuroscore and BE accuracy without normalization (including RFACE category).	185
D.4	Two-stage training details for Chapter 7.	186

List of Tables

3.1	CNNs architectures in the literature.	53
4.1	Hyperparameter summary for each pipeline discussed in this chapter.	66
4.2	AUC score (%) for different pipelines across nine participants in testing session.	69
5.1	Summary of loss-variant for GANs.	109
6.1	Accuracy for face images generated from three GANs and real face images in the BE task.	122
6.2	Computed Neuroscore of each participant for each category.	125
6.3	Score comparison for each GAN category.	127
7.1	Comparison between Neuroscore and other metrics.	133
7.2	Number of trials for each stimulus type remaining after artifact rejection across each participant (ID) and different GAN categories.	138
7.3	Errors of 9 models for cross participants.	139
7.4	Performance of three conventional scores and Neuroscore.	142
A.1	Time-related latencies between image presentation in software and physical image presentation of different groups.	178
B.1	Details of SNR for Fig. 4.7 in Chapter 4.	182
B.2	Details of AUC score for Fig. 4.8 in Chapter 4.	182

Abstract

Cortically Coupled Image Computing

Zhengwei Wang

In the 1970s, researchers at the University of California started to investigate communication between humans and computers using neural signals, which lead to the emergence of brain-computer interfaces (BCIs). In the past 40 years, significant progress has been achieved in application areas such as neuroprosthetics and rehabilitation. BCIs have been recently applied to media analytics (e.g., image search and information retrieval) as we are surrounded by tremendous amounts of media information today. A cortically coupled computer vision (CCCV) system is a type of BCI that exposes users to high throughput image streams via the rapid serial visual presentation (RSVP) protocol. Media analytics has also been transformed through the enormous advances in artificial intelligence (AI) in recent times. Understanding and presenting the nature of the human-AI relationship will play an important role in our society in the future. This thesis explores two lines of research in the context of traditional BCIs and AI. Firstly, we study and investigate the fundamental processing methods such as feature extraction and classification for CCCV systems. Secondly, we discuss the feasibility of interfacing neural systems with AI technology through CCCV, an area we identify as neuro-AI interfacing. We have made two electroencephalography (EEG) datasets available to the community that support our investigation of these two research directions. These are the neurally augmented image labelling strategies (NAILS) dataset and the neural indices for face perception analysis (NIFPA) dataset, which are introduced in Chapter 2.

The first line of research focuses on studying and investigating fundamental processing methods for CCCV. In Chapter 3, we present a review on recent developments in processing methods for CCCV. This review introduces CCCV related components, specifically the RSVP experimental setup, RSVP-EEG phenomena such as the P300 and N170, evaluation metrics, feature extraction and classification. We then provide a detailed study and an analysis on spatial filtering pipelines in Chapter 4, which are the most widely used feature extraction and reduction methods in a CCCV system. In this context, we propose a spatial filtering technique named multiple time window LDA beamformers (MTWLB) and compare it to two other well-known techniques in the literature, namely xDAWN and common spatial patterns (CSP). Importantly, we demonstrate the efficacy of MTWLB for time-course source signal reconstruction compared to existing methods, which we then use as a source signal information extraction method to support a neuro-AI interface. This will be further discussed in this thesis i.e. Chapter 6 and Chapter 7.

The latter part of this thesis investigates the feasibility of neuro-AI interfaces. We present two research studies which contribute to this direction. Firstly, we explore the idea of neuro-AI interfaces based on stimulus and neural systems i.e., observation of the effects of stimuli produced by different AI systems on neural signals. We use generative adversarial networks

(GANs) to produce image stimuli in this case as GANs are able to produce higher quality images compared to other deep generative models. Chapter 5 provides a review on GAN-variants in terms of loss functions and architectures. In Chapter 6, we design a comprehensive experiment to verify the effects of images produced by different GANs on participants' EEG responses. In this we propose a biologically-produced metric called Neuroscore for evaluating GAN performance. We highlight the consistency between Neuroscore and human perceptual judgment, which is superior to conventional metrics (i.e., Inception Score (IS), Fréchet Inception Distance (FID) and Kernel Maximum Mean Discrepancy (MMD) discussed in this thesis). Secondly, in order to generalize Neuroscore, we explore the use of a neuro-AI interface to help convolutional neural networks (CNNs) predict a Neuroscore with only an image as the input. In this scenario, we feed the reconstructed P300 source signals to the intermediate layer as supervisory information. We demonstrate that including biological neural information can improve the prediction performance for our proposed CNN models and the predicted Neuroscore is highly correlated with the real Neuroscore (as directly calculated from human neural signals).

Chapter 1

Introduction

***Abstract:** This chapter introduces basic concepts and presents the background information in related research areas to be discussed in this thesis. It also introduces the motivation and highlights the contributions in this thesis.*

This thesis investigates the feasibility of deploying brain-computer interfaces (BCIs) in the area of media analytics. Research presented in this thesis spans fields including neuroscience, machine learning and deep learning. We discuss BCI research basically from two perspectives: (1) We investigate the imagery triaging ability of a traditional BCI system that is exposed to a rapid serial visual presentation (RSVP) protocol and we call this type of system as cortically coupled computer vision (CCCV) system. We explore and inspect the efficacy and performance of different spatial filtering pipelines for this type of system. Contents related to this type of research line will be introduced in Chapter 3 and Chapter 4; and (2) We deploy human neural responses to interface with AI systems. We demonstrate the concept of neuro-AI interfaces using two different frameworks. The first framework presents images produced by generative adversarial networks (GANs) to participants and uses participants' neural feedback, electroencephalography (EEG) is used in this case, to assess the quality of images produced by GANs. This work will be demonstrated in Chapter 6. The second framework is to demonstrate that neural responses can be used as supervisory information to train a deep neural network (DNN), which can assist a DNN to accomplish some difficult tasks in the future i.e., evaluating image quality in our case. This work will be introduced in Chapter 7. Chapter 2 describes EEG datasets used in this thesis while Chapter 5 provides a review on generative adversarial networks (GANs) where GANs were used to produce the image stimuli in our experiment. Before starting to discuss the research work in this thesis, we first introduce some background knowledge

in related areas.

1.1 Brain-computer Interfaces and Event-related Potentials

Electroencephalography (EEG) is a non-invasive measurement of a human being's brain waves that was firstly measured by the German psychiatrist, Hans Berger in 1924. EEG is the electric recording of the summed electric activity of populations of neurons, which is measured by using electrodes placed on the scalp. It has been widely applied in clinical contexts. It is used in the evaluation of several types of brain disorders such as epilepsy detection [1,2], lesions in the brain [3–5], sleep disorders [6, 7], Alzheimer's disease [8,9] and psychoses [10]. EEG is also able to provide indications for evaluating trauma [11], drug intoxication [12] and the extent of brain damage [13]. With successful deployment in clinical contexts, internal characteristics are also well researched in the literature. EEG comprises different waveforms, which can be generally characterized by their frequencies, amplitudes, shapes as well as the locations i.e., sites on the scalp where they are recorded. Frequency is a key characteristic for classifying different types of EEG waveforms. Alpha rhythm [14] appears in the frequency ranging between 8 Hz and 13 Hz, which is activated during the relaxed wakefulness human brain and it is normally attenuated or abolished by visual attention and affected transiently by other sensory stimuli and by other mental alerting activities [15]. Theta rhythm appears in the frequency ranging between 4 Hz and 7 Hz, which is encountered in the front central regions and is usually related to drowsiness or heightened emotional states [16]. Alpha and theta reflect human cognitive and memory performance [17]. Delta rhythm [18] appears between 0.5 Hz and 4 Hz and is associated with sleep [19]. Beta rhythm appears between 14 Hz and 25 Hz, sometimes can be augmented by drugs [20] and increases with heavy breathing. Gamma appears over 30 Hz and is implicated in creating the unity of conscious perception [21]. Research related to those internal characteristics behind EEG provides the neurophysiological evidence that EEG is related to human behavior, brain function, external stimulus etc.

From an engineering perspective, we are generally interested in leveraging patterns of activity in EEG for some real-world applications. BCIs enable such a way for engineering researchers to create a direct communication channel between the brain and computers. BCIs are basically divided into two types: (1) Non-invasive BCI where the sensors are placed on the head for the purposes of measuring signals related to brain activity e.g., EEG; and (2) Invasive BCIs,

where the electrodes are placed directly into cortex by using surgical operation e.g., electrocorticography (ECoG) and local field potential (LFP). BCIs discussed in this thesis are EEG-based which all belong to the non-invasive category. A typical BCI comprises five parts [22]: (1) Data acquisition — the EEG (electroencephalogram) signals are recorded by an EEG amplifier; (2) Pre-processing — this step includes signal denoising e.g., filtering, artifact rejection, normalization and re-referencing. Good pre-processing yields EEG useful for subsequent analysis; (3) Feature extraction — this part aims to extract meaningful features that can be used for training machine learning algorithms later. The advantages of a suitable feature extraction method can include improving the signal-to-noise ratio (SNR) and allowing dimensionality reduction, which has effects on the later training part; (4) Classifiers and machine learning algorithms – in order to translate the features into device commands, it is necessary to apply machine learning techniques to the extracted features where both linear and nonlinear methods are often used in this step [23, 24]; and (5) Output device — the output device can be any type of electronic element that can receive the command in the last step e.g., a computer screen. Figure 1.1 illustrates a visual overview of a generic BCI system.

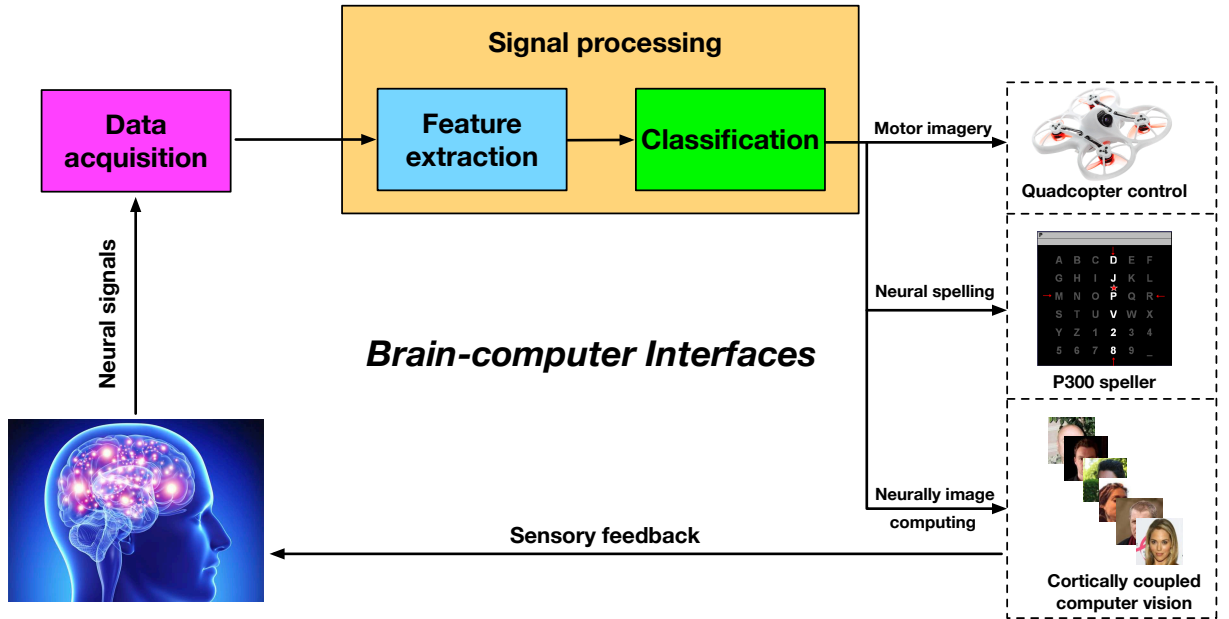


Figure 1.1: An overview of a generic BCI system.

illustrates a visual overview of a generic BCI system. Three examples, specifically quadcopter control [25], P300 speller [26] and CCCV [27], are demonstrated. The end goal of a BCI system is to use brain signals (e.g., EEG is used in this work) to make the output device responsive to thoughts.

Event-related potentials (ERPs) are very small voltages (typically 10^{-6} V) generated in the

brain structure in response to specific events or stimuli, which are time-locked to sensory, motor or cognitive events [28]. Two types of ERPs are stated in the literature, namely exogenous ERPs and endogenous ERPs [29]. Exogenous (also referred to as sensory) components appear relatively early (50 ms to 250 ms) after stimulus presentation and depend on the physical parameters of the stimulus e.g., shape and brightness [30]. Endogenous (or named as cognitive) components appear later (> 250 ms) after stimulus presentation and they are not sensitive to stimulus type. These components are typically related to cognition and information processing in the brain e.g., attention and memory. Each ERP is unique and is typically categorized using three properties: (1) Amplitude (10^{-6} V) is measured at the peak of ERPs. Polarity is used as the discriminant information; (2) Location spatial topography, where different ERPs can be recorded from different locations on the scalp; and (3) Latency refers to the time measured when the ERP's peak appears after stimulus presentation. Latency is determined by the type of ERP. The early ERP components peaking at short latencies are normally related to exogenous components. The later ERP components peaking with a large latency usually belong to the family of endogenous components. This thesis is built up upon one of the endogenous ERPs — the P300 component, which reflects aspects of the attention process in humans.

The P300 component was first reported in 1965 [31] and can be elicited by an attention-driven task i.e., the P300 will appear when participants are paying attention for some specific stimuli to appear [32, 33]. The terminology “P300” indicates the amplitude and latency properties mentioned before for the component. “P” refers to a positive-going peak of the ERPs waveform and “300” refers to the time window 300 ms – 600 ms that the P300 may appear in. The location of the P300 is distributed over the middle electrodes (Fz, Cz, Pz), which typically increases in amplitude from frontal to parietal electrode sites (it depends on the type of stimuli) [34]. Moreover, the P300 comprises two subcomponents, which are known as P3a and P3b. The P3a can be elicited by an infrequent distractor stimulus randomly inserted in a presentation stream and it comes from frontal or central electrode sites on the scalp [35, 36]. The P3b is task-relevant, which is elicited during target stimulus processing [37, 38] and presents parietally on the scalp [39]. This thesis mainly focuses on discussing the P3b because our experiments are designed upon target detection in this work.

The first P300-BCI was introduced in 1988 [40] and the P300 is the most widely used ERP component for a BCI system currently [41]. A P300-BCI benefits from its fast speed, a straightforward test (e.g., detect target) for participants and no requirement for training par-

ticipants. Despite the well known P300 speller system [42], recent research has shown that P300-BCIs have a wide range of different usages. For example, they can be used to assist the disabled [43–45]. In addition, new experimental paradigms related to P300-BCI applications have been demonstrated in the literature [26, 46, 47] and a P300-BCI was also demonstrated to improve human attention [48]. This thesis focuses on the use of a visual P300-BCI for searching and evaluating images.

1.2 Rapid Serial Visual Presentation Paradigm

Rapid serial visual presentation (RSVP) involves a series of images being rapidly presented to participants at a specific position on a screen. It was first introduced in [49]. It can be divided into text RSVP and image RSVP [50]. For the text RSVP, individual words are presented sequentially in a fixed place on a screen where it has applications such as high-speed in reading and assist reading for the disabled [50]. Similar to the text RSVP, the image RSVP involves rapid presentation of images to participants during the experiment. This thesis employs the image RSVP paradigm.

The concept of RSVP can be introduced using a familiar example, that of rapidly riffling through the pages of a book in order to locate a needed image [50]. In RSVP, a rapid succession of target and standard (non-target) images are presented to a participant on a display at a rate of 4 Hz – 10 Hz. The location of target images within the high-speed presentation is not known in advance by users and hence requires them to actively look out for targets i.e., to attend to target images. This paradigm where users are instructed to attend to target images amongst a larger proportion of standard images is known as an oddball paradigm and is commonly used to elicit ERPs such as the P300, a positive voltage deflection that typically occurs between 300 ms – 600 ms after the appearance of a rare visual target within a sequence of frequent irrelevant stimuli [39]. Since participants do not know when target images will appear in the presentation sequence, their occurrence causes an attentional-orientation response that is characterized by the presence of a P300 ERP.

RSVP was designed to explore the human visual processing system in the literature [51–55]. [51] showed how long a target image needs to be displayed so that it can be perceived by participants, implying an upper-bound in the display rate in a RSVP paradigm. An important finding during this procedure was that participants experienced a condition named attentional

1.2. Rapid Serial Visual Presentation Paradigm

blink [51]. This condition was characterized by failure to detect a target when it follows another target immediately presented within a stream of visual stimuli in rapid succession at the same spatial location on a screen. This happened when the second target appeared between 180 ms and 450 ms [52, 55] after the first. This result indicates an important point when designing a RSVP experiment, where the time interval between two targets cannot be too small otherwise the target interference may deteriorate the performance of the RSVP experiment [56].

There are three modes of RSVP paradigms (see Fig. 1.2): (1) Static mode, where images appear and disappear without moving; (2) Moving mode, where images appear and disappear sequentially; and (3) Multiple entries/exits, where images appear and disappear in many locations or move along several paths [50]. Static mode has been shown to be more robust and

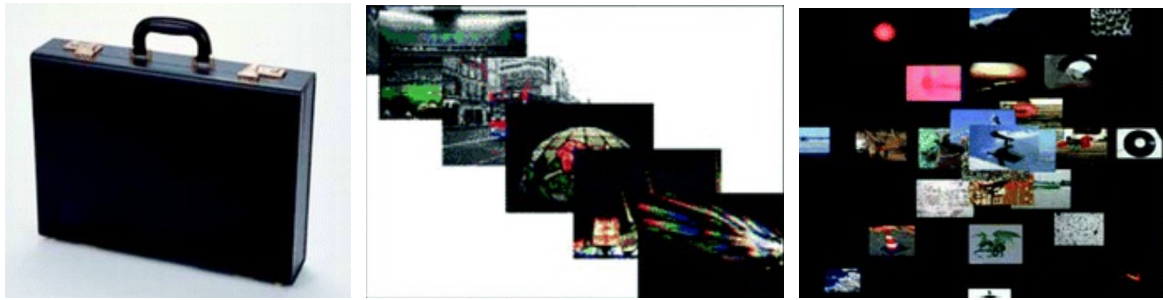


Figure 1.2: Three RSVP modes i.e., static mode, moving mode and multiple entries/exits mode (from left to right).

effective compared to other modes, where static mode is associated with a higher target recognition rate and a higher presentation rate (i.e., number of images presented per second) [57]. Multiple entries/exits mode is a more complex RSVP mode, which can be used for exploring gaze movement along the routes taken by images [58]. A good RSVP experiment should be designed with the following considerations: (1) Examine the tasks associated with the application to see what benefits can be provided by RSVP; (2) Select the mode that is influenced by availability of context presentation and manual control of rate and direction; (3) Presentation rates should be slower when a task has an increased complexity; and (4) Be aware of a potential link between eye-gaze movement and the success with which images may be recognized [50]. We follow previous researchers' steps by using the static mode for designing our experiment in this thesis [59–61].

1.3 Cortically Coupled Computer Vision System

Cortically coupled computer vision (CCCV) is the integration of BCIs and the RSVP paradigm. CCCV uses neural signals to implement image computation (e.g., image detection/search) for a rapid image stream. Figure 1.3 illustrates a typical example of a CCCV system. A participant

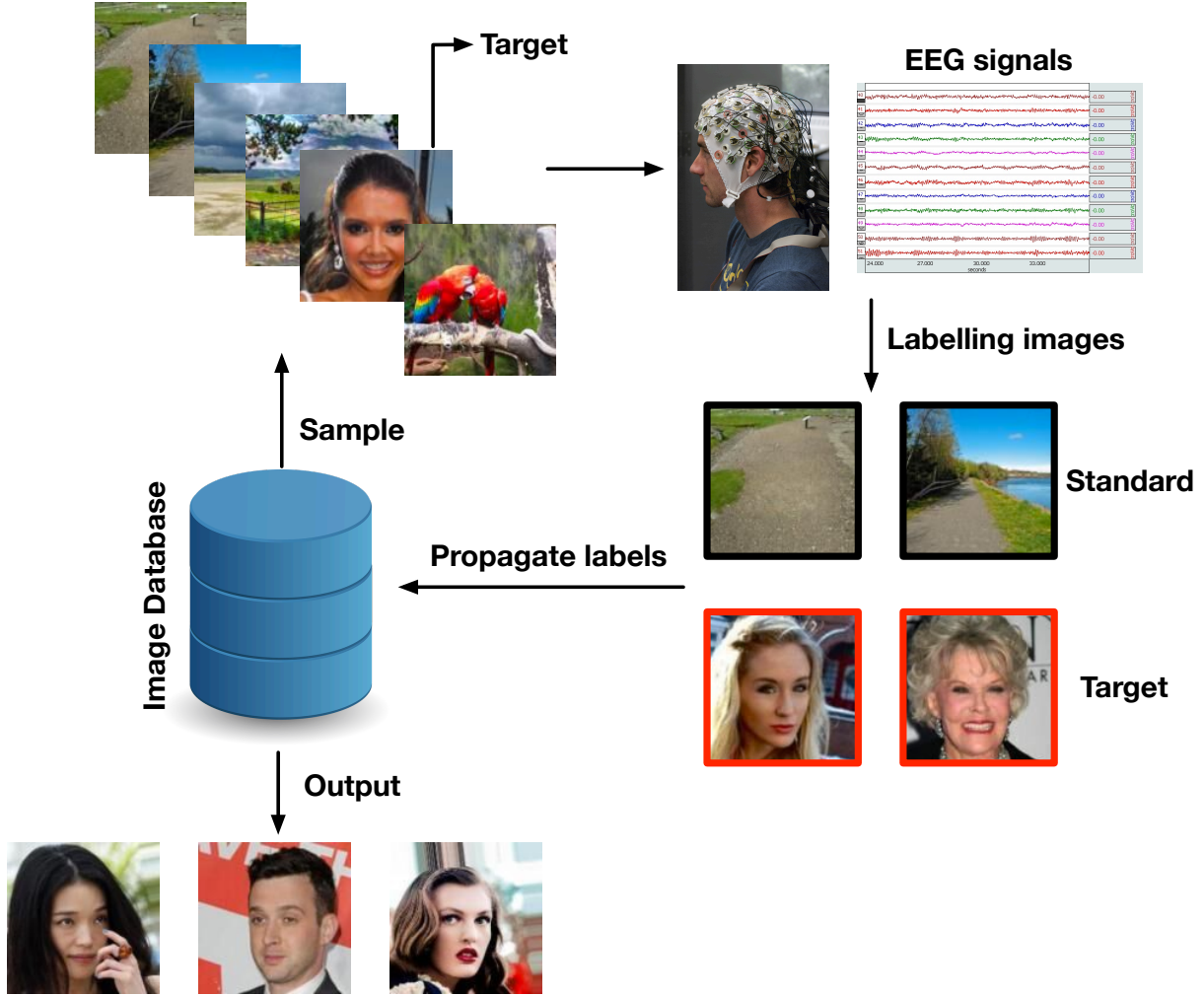


Figure 1.3: An example of CCCV system. The image stream sampled from the image database is presented to the participant and EEG signals are recorded simultaneously. Images are then triaged by using a participant's EEG signal. The labels are then propagated back to the image database and output image of interest [62].

is asked to search for specific type of images (face images in particular in this case) in an image stream presented by a RSVP paradigm and EEG signals are recorded simultaneously. The P300 component would be elicited when participants see the appearance of target images. With proper machine learning techniques applied, images can then be labelled based on whether the P300 component is detected or not. Compared to current advanced computer vision systems, a CCCV system has some advantages: (1) Participants do not need to be trained regarding the

generation of P300; (2) This system is capable of being applied to those small image datasets (e.g., satellite image datasets [63]) whereas current deep learning approaches typically depend on a number of training samples; (3) This system has relatively light computation and literature has already demonstrated the possibility of real-time processing [27, 61, 64].

In this thesis, CCCV is explored with respect to searching for target images and evaluating images produced by generative adversarial networks (GANs). Regarding the first aspect, we explore and demonstrate efficient spatial filtering pipelines for a CCCV system. Moreover, we propose a spatial filtering approach called multiple time window LDA beamformer (MTWLB) for better reconstructing the time-course source signals for a CCCV system. In terms of the second aspect, we couple CCCV systems with two types of artificial intelligent (AI) systems, namely GANs and convolutional neural networks (CNNs), where we call this approach neuro-AI interfacing. We demonstrate the use of neuro-AI interfaces for (1) use in CCCV to produce a biologically neuro-produced metric called Neuroscore to evaluate the performance of GANs and (2) use in CCCV to provide biological neural information to train a CNN.

1.4 Generative Adversarial Networks

Generative adversarial networks (GANs) [65] are increasingly attracting attention in computer vision, natural language processing, speech synthesis and similar domains. Arguably the most striking results have been in the area of image synthesis. Figure 1.4 demonstrates the architecture of a typical GAN. The architecture comprises two components, one of which is a discriminator (D) distinguishing between real images and generated images while the other one is a generator (G) creating images to fool the discriminator. Given a distribution $\mathbf{z} \sim p_{\mathbf{z}}$, generator defines a probability distribution p_g as the distribution of the samples $G(\mathbf{z})$. The objective of a GAN is to learn the generator's distribution p_g that approximates the real data distribution p_r . Optimization of a GAN is performed with respect to a joint loss function for D and G

$$\min_G \max_D \mathbb{E}_{\mathbf{x} \sim p_r} \log[D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}} \log[1 - D(G(\mathbf{z}))] \quad (1.1)$$

A GAN is a member of deep generative models (DGMs) family and was proposed to be trained by using a min-max game theory strategy between two players (a discriminator D and generator G). The generator behaves like a “forger”, which aims to produce generated data to fool the discriminator. The discriminator behaves like a “detective”, which aims to distinguish the

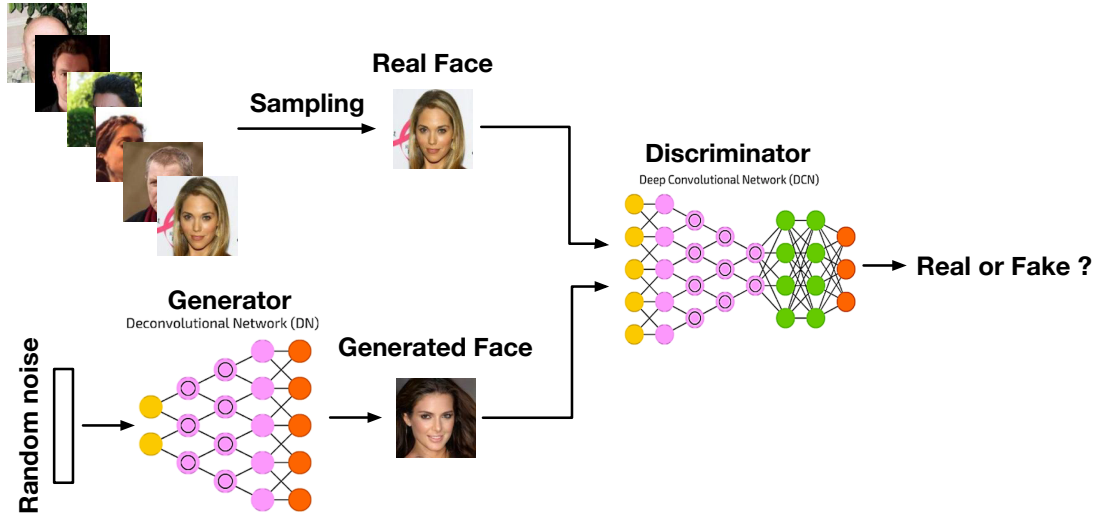


Figure 1.4: Architecture of a GAN. Two deep neural networks (discriminator (D) and generator (G)) are synchronously trained during the learning stage. The discriminator is optimized in order to distinguish between real images and generated images while the generator is trained for the sake of fooling discriminator from discerning between real images and generated images.

real data from the generated data. The optimal generator will produce the synthetic data that discriminator cannot tell is synthetic from real data. GANs have benefits compared to the traditional DGMs. First, the architectures in GANs are very flexible. Any type of architecture for a specific application can be used as the generator and the discriminator. Second, GANs are able to handle sharp density functions and produce high quality images. Third, they are able to produce diverse image samples. However, GANs suffer from problems such as being difficult to train and hard to evaluate. The first problem is more theoretic, which is discussed in Chapter 5. Regarding the second challenge, this thesis explores the use of human perception, which utilizes a CCCV pipeline to contribute that area.

There is limited research combining GANs and neuroscience in the literature. Current literature focuses on using GANs to improve the spatial resolution of EEG signals or to produce synthetic EEG signals [66–68]. Research on exploring the use of outputs of GANs as experimental stimuli in the area of neuroscience is limited. This research direction has mutual benefits for both the deep learning and neuroscience domains. Firstly, the use of neurally-inspired metrics to evaluate the performance of GANs would improve the process of training GANs. Secondly, for the automatic generation of stimuli for a neuroscientific experiment. Traditional preparation of stimulus for a neuroscientific experiment is a time-consuming process. By using the modern deep learning techniques such as GANs, it saves lots of time and more importantly it enables experimental stimuli to be very flexible e.g., researchers can easily customize the type of image

stimuli use for the experiment. We explore this research direction to bridge the gap in current literature by using CCCV systems with GANs.

1.5 Neuro-AI Interfaces

Biological Neural Networks In neuroscience, biological neural networks (BNNs) describe recognizable pathway of groups of interconnected neurons where these neurons communicate with each other by using electrical impulses. There are two main types of electrical activity associated with neurons, which are action potentials and postsynaptic potentials [69]. Action potentials are discrete voltages spikes that travel from the beginning of the axon at the cell body to the axon terminals (red flow as seen in Fig. 1.5), where neurotransmitters are released. A spike train is such a type of neural activity, where a neuron fires an action potential at a

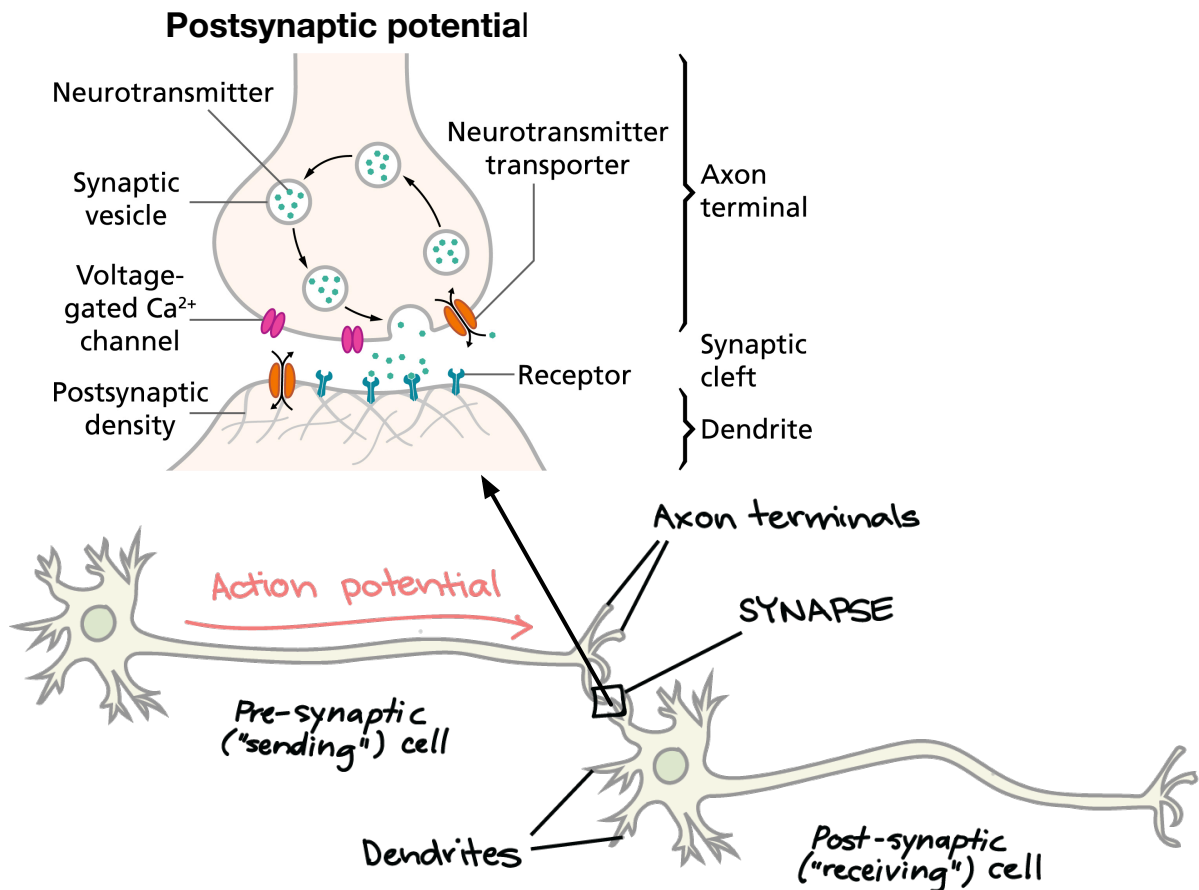


Figure 1.5: Schematic of action potentials and postsynaptic potential. Figure was constructed via [70] and [71].

sequence of recorded times [72]. Postsynaptic potentials (as seen in Fig. 1.5) are the voltages that arise when the neurotransmitters bind to receptors on the membrane of the postsynaptic

cell, causing ion channels to open or close, leading to a graded change in the potential across the cell membrane [69]. ERPs discussed in this thesis are produced by postsynaptic potentials.

Artificial Neural Networks In general, an artificial neural network (ANN) comprises four components: (1) Neurons; (2) Weights; (3) Connectivity (topology of network e.g., fully-connected networks and CNNs); and (4) A learning rule [73]. An ANN is normally trained with weights carefully initialized in order to speed up convergence [74] and fixed connectivity. An optimization problem of ANNs is to map inputs to desired outputs, and weights in ANNs are updated through solving the optimization problem by using backpropagation [75].

Relationship between AI and Neuroscience It is well known that ANNs are inspired by BNNs. As seen in Fig. 1.6, an artificial neuron receives a number of inputs produced by neu-

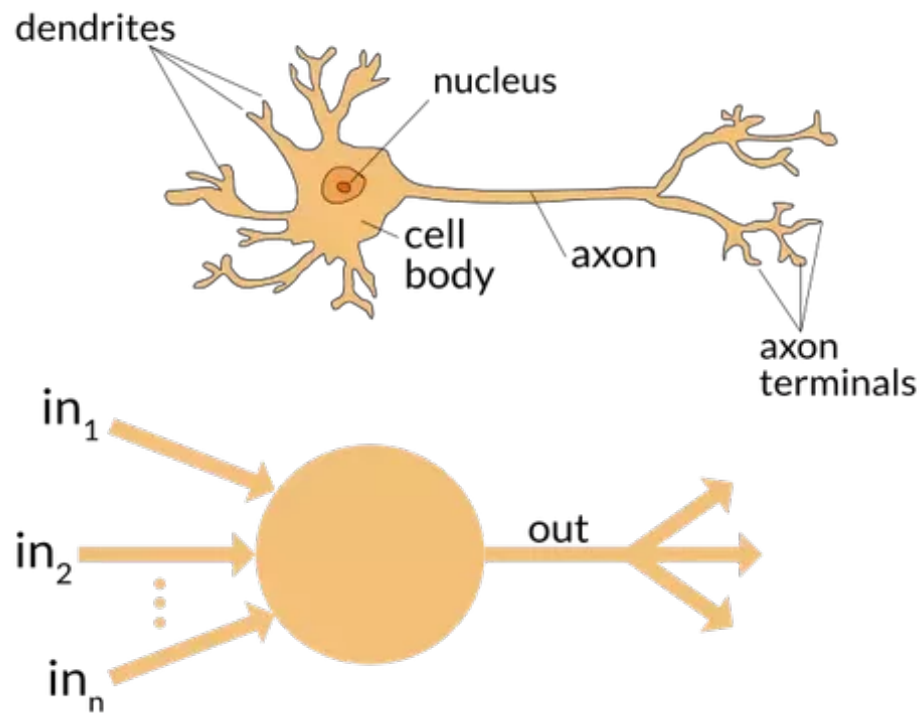


Figure 1.6: Comparison between a biological neuron and an artificial neuron. Top figure describes a biological neuron while the bottom one demonstrates an artificial neuron. Figure from [76].

rons in previous layer and the neuron will be activated when the sum of input values are above the threshold, which is similar to that of a biological neuron that has dendrites to receive signals from other neurons. A cell body in effect processes the inputs and via its axon(s) signals with action potential fired by other neurons. However, BNNs have numerous differences to ANNs. Firstly, the initial state of BNNs are not random and are genetically influenced [77, 78].

Secondly, the connectivity and the weights are always changing over time and across different tasks. On the contrary, the connectivity of ANNs is fixed over time and the weights of ANNs are fixed after training. In the deep learning area, researchers focus on developing ANNs that are more biologically plausible in the past 30 years [79–81]. For instance, convolutional neural networks (CNNs) were developed by emulating human retina and brain visual system, which split images into tiny areas for object recognition [82–85]. Another example is recurrent neural networks, which use the neuron’s output as a feedback so it introduces memory functionality to the model [86]. This is also similar to BNNs as functional connectivity in BNNs, described by statistical dependencies such as transfer entropy, coherence, and correlations, are significantly similar in some respects [87–89]. Benefiting from the fundamental knowledge achieved in the areas such as neuroscience, deep neural networks (DNNs) and artificial intelligence (AI) have grown up and are revolutionizing our society today. From the other side, benefit by DNNs prevailing today, neuroscience research is pushed significantly forward by utilizing DNNs technology. For example, work [90] demonstrated that CNNs can be used to predict neural response in the V4 cortex when doing image recognition tasks. Other recent [91] showed that current ANNs are able to control the activity of neural populations, which has wide reaching therapeutic applications such as treatment for depressive disorder.

Both AI and neuroscientific research have achieved significant progress in their own fields, however, research on interconnections between these two areas are still limited and unexplored. Broadly speaking, it is under investigated what utility neural information can provide for AI systems. In this thesis, we explore this area by introducing the concept of “neuro-AI interfaces”, which creates a pathway between a human neural systems and AI systems. We introduce neuro-AI interfaces that use CCCV to interact with AI systems. We introduce neuro-AI interfaces via one of the fundamental problems in GANs: designing a proper evaluation metric to assess the quality of images produced by GANs, which is discussed in detail in Chapter 6 and Chapter 7.

1.6 Research Motivation and Contributions

Traditional CCCV research focuses on using a human’s neural signals (EEG) to find target images from a large database. A review of feature extraction and classification methods is presented in Chapter 3, where the original content has been published as a book chapter in the *Signal Processing and Machine Learning for Brain-Machine Interfaces* [92]. In the literature,

lots of research has been carried out to improve the classification performance for a CCCV system. In our work, we pay close attention to both classification performance and the neurophysiological interpretability when using a CCCV system. Motivated by this objective, we investigate the spatial filtering pipeline (i.e., the most widely used strategy in CCCV research) for a CCCV system in Chapter 4, which enables interpretation of EEG signals from both a spatial space and a temporal space. LDA beamformers based on multiple time windows are proposed for a CCCV system and we also demonstrate its efficacy on the reconstructed time-course source signals. This pipeline is then utilized to reconstruct a time-course source signal (the P300) for our neuro-AI interface in the later part of the thesis. Part of the content presented in Chapter 4 has been published in the *Journal of Brain-Computer Interfaces* [93].

GANs are attracting growing attention in the deep learning community and have been applied to lots of different domains. Some reviews [94–98] on GANs have been published in the literature but discussion on the performance of GAN-variants is still lacking. In Chapter 5, we provide a survey on GAN-variants based on architectures and loss functions, where we discuss and analyze their performance (i.e., high quality image generation and stable training) in the context of the area of computer vision. We hope this review will help researchers both from and outside the deep learning community when deploying GANs in their research.

Lots of work has been carried out in regard to using DNNs for classifying the EEG responses [99–101] where many approaches have achieved good classification performance for CCCV systems [102, 103]. With the blossoming of deep learning techniques today, the exploration of the interconnections between areas of deep learning and biological neural systems deserves attention. Compared to traditional machine learning/deep learning systems, biological neural systems can be more robust to small perturbations on inputs, are more easily generalized to other tasks and directly reflect functionality in the brain system such as for cognition, attention and memory. Studying the interconnections between AI and neuroscience can be useful to enhance our understanding of how the brain works and improve the performance for AI systems through using informative knowledge derived from biological neural systems (or building more biologically plausible DNNs models). Thus investigation on this area has potential impact for both AI and neuroscience fields. In this thesis, we explore the interconnections between deep learning and neuroscience domains through CCCV technology. Firstly, we explore the feasibility of using the P300 component, which is produced by the spatial filtering pipeline mentioned in the previous paragraph, to evaluate the performance of GANs in Chapter 6. We propose

Neuroscore as a biologically neurally-produced metric for scoring a GAN and demonstrate that Neuroscore is highly correlated with human perceptual judgment. To the best of our knowledge, this is the first research line that combines human perception with GANs. This chapter has been published in the *Cognitive Computation*. Secondly, we consider if a DNN is able to learn useful information from neural signals and model the performance generated by human brain signals. In Chapter 7, we propose a CNN based neuro-AI interface to evaluate GANs, which synthesizes the Neuroscore. Importantly, we show that including neural responses during the training phase of the network significantly improves the prediction performance of the proposed model.

Finally, this thesis contributes two open EEG datasets: neurally augmented image labelling strategies (NAILS) and neural indices for face perception analysis (NIFPA). We hope these two datasets will be helpful to other researchers working in this field.

1.7 Research Questions

This thesis visits the areas of neuroscience, machine learning and deep learning. With respect to neuroscience, we aim to investigate human beings' neural responses (i.e., ERPs in EEG signals) to image stimuli. We are interested in using neural responses in image target search and in image quality evaluation, in what we refer to as a CCCV system. From the machine learning aspect, we explore methods (i.e., feature extraction and classification) that efficiently performing classification and maintain neurophysiological interpretability. In terms of deep learning, we explore the possibility of using neural signals to evaluate image stimuli produced by GANs and the efficacy of adding neural responses as supervisory information to DNNs.

Below are three research questions that arise in these three areas (namely neuroscience, machine learning and deep learning), which shape the content of this thesis:

1. **Can we improve on the extraction of discriminative ERP components while preserving neurophysiological interpretability for a CCCV system?**

This research question explores the effectiveness of different spatial filtering approaches in a CCCV system with respect to their classification performance and neurophysiological interpretability. Chapter 3 visits this area and we propose a spatial filtering approach named multiple time window LDA beamformer (MTWLB). Its neurophysiological interpretability is demonstrated and it is compared to existing approaches in the literature.

An important problem for ERP research is that EEG signals suffer from a low SNR is-

sue, especially for CCCV systems. The ability to reconstruct ERP source signals is not only beneficial to the classification performance of a CCCV system but may also have positive effects on ERP research carried out using a RSVP protocol i.e., high SNR signals (compared to the signals recorded from one electrode) can be used for ERP analysis which gives more robust results in terms of different experimental conditions. We also investigate the efficacy of our proposed approach in this aspect — the performance of reconstructing the ERP source signals under the RSVP experimental protocol.

2. Can neural signals be used to provide indications on image quality that is consistent with human perceptual judgment and is it possible to use this as a biological score to evaluate generative models such as GANs?

AI has a significant impact on our society. Research into the interaction between humans and AI deserves further exploration and has only recently become a research focus. We explore the possible interface between generative models (i.e., GANs used in this case) and human neural systems. As a starting point, we investigate the feasibility of using neural signals (EEG) to evaluate the performance of GANs. The main concern of evaluating the performance of GANs in the current literature is whether current evaluation metrics are consistent with human perceptual judgment. In Chapter 6, we address this question by introducing a neurally-produced metric called Neuroscore, which is produced by using a CCCV system. A systematic comparison between Neuroscore, human perceptual judgment and other evaluation metrics is carried out.

3. Is it possible to interface biological neural systems and AI systems and if so, can biological neural signals provide any type of informative knowledge for helping AI systems to learn a difficult task?

The current literature has demonstrated the use of DNNs in emulating the encoding processes of the brain during an image object recognition task via invasive measurements, which indicates the encoding processes of the brain and of DNNs are similar to each other. Given this evidence, we are interested in understanding if biological neural information via a non-invasive measurement (EEG in this case) is transferable to DNNs to help DNNs accomplish difficult tasks. As a starting point, we introduce the concept of a neuro-AI interface, which uses the P300 signal as a source of supervisory information to help CNNs produce a Neuroscore for the purpose of evaluating the performance of GANs (Chapter 7).

thetic images produced by GANs along with the real images of the same type. Both N170 and P300 ERPs were elicited in this dataset. We introduce the properties of ERPs from the perspectives of time course and topographical distribution. Details of analysis and feature extraction of ERPs are introduced in Chapter 3.

- **Chapter 3** presents a comprehensive review on the processing methods for RSVP-EEG. We introduce the experimental set up for recording the RSVP-EEG data and study the feature extraction and classification algorithms applied for this type of data. In terms of the feature extraction, we mainly consider two most popular areas, spatial filtering and time-frequency representation, to extract informative features from ERPs. Regarding the classification methods, we study the traditional linear classification methods and the recent momentum deep learning methods.
- **Chapter 4** analyzes and discusses the spatial filtering pipeline deployed in cortically coupled image classification. We propose a spatial filtering approach called multiple time window LDA beamformers (MTWLB), which uses LDA beamformers based on multiple time windows. The proposed MTWLB is compared with other two famous approaches in the literature, which are xDAWN and common spatial pattern (CSP). Moreover, we compare the performance between xDAWN and MTWLB in terms of reconstructing the time course of the source signals. The performance of three linear classifiers, linear discriminant analysis (LDA), Bayesian linear regression (BLR) and logistic regression (LR) are also discussed.
- **Chapter 5** provides a review on GAN-variants from architecture and loss perspectives. We analyze the problems in the original GAN and study the GAN-variants that deal with those problems in the literature.
- **Chapter 6** demonstrates a neuro-AI interface, which deploys a novel cortically coupled paradigm that uses neural signals to evaluate the performance of GANs. Neuroscore has been proposed as an evaluation metric for GANs, which closely mirrors behavioral human perception on the images produced by GANs. Neuroscore is compared to three conventional evaluation metrics in the literature.
- **Chapter 7** extends the work in Chapter 6, in which a CNN based neuro-AI interface is proposed to produce the Neuroscore. We show that DNNs are able to learn informative

knowledge from human neural responses and successfully demonstrate a CNN trained by using neural responses to evaluate the performance for GANs.

- **Chapter 8** concludes the thesis. Future directions are also discussed with respect to the different research areas, which are ERP research, computational neuroscience and neuro-AI interfaces.

Chapter 2

Data Description

Abstract: *This chapter describes electroencephalography (EEG) datasets that are used in this thesis. Two EEG datasets are introduced: (1) The neurally augmented image labelling strategies (NAILS) dataset is used to support a collaborative evaluation task in which participating researchers benchmarked machine learning strategies against each other. The experimental protocol used to capture the dataset was designed to encompass a broad range of image search activities and coincident neural signals; and (2) The neural indices for face perception analysis (NIFPA) dataset is to explore human being’s EEG responses to synthetic images generated by using generative adversarial networks (GANs) and real images of the same type. By doing so, we will be able to compare differences in neural responses that indicate whether images are perceived as being real or fake. Identifying and operationalizing such EEG responses would provide an alternative method for the evaluation of GANs and a feedback signal to improve their effectiveness of GAN. The NAILS dataset has already been released [104] and the NIFPA dataset will be released as well at some stage.*

2.1 Introduction

A brain-computer interface (BCI) provides a communication pathway between the human brain and a computer system. This type of interface requires the development of algorithms for translating the measured brain signals into computer commands. Brain signals can be measured invasively [105] and non-invasively [106]. Non-invasive BCIs are more commonly being researched in the BCI community [107]. Electroencephalography (EEG) signals, acquired via a non-invasive manner to enable BCIs, have been widely studied in the literature. Applications

2.1. Introduction

designed using such type of BCIs span broad areas e.g., information retrieval [108], rehabilitation engineering [109] and cortically coupled computer vision [59].

The P300, a type of event-related potential (ERP), is a well known EEG component that has been widely used to drive BCI systems (e.g. the P300 speller applications [110]). P300-BCIs for multimedia information retrieval have attracted growing interests in recent years [93, 103, 111]. Open EEG datasets related to this field are severely lacking. In this chapter, we provide details of the two recorded EEG datasets used in this thesis: (1) The neurally augmented image labelling strategies (NAILS) dataset, which has an affiliated workshop to support the collaborative evaluation of best-practice strategies for rapid serial visual presentation (RSVP) image search using EEG signals; and (2) The neural indices for face perception analysis (NIFPA) dataset, which is to explore a human being's EEG responses to synthetic images produced by generative adversarial networks (GANs).

In this thesis, we used 32 channel BrainVision actiCHamp at 1000 Hz for recording EEG signals and electrode locations defined by 10-20 system were carried out as seen in Fig. 2.1. A list of related neurophysiologically-relevant terminology and associated explanations used in

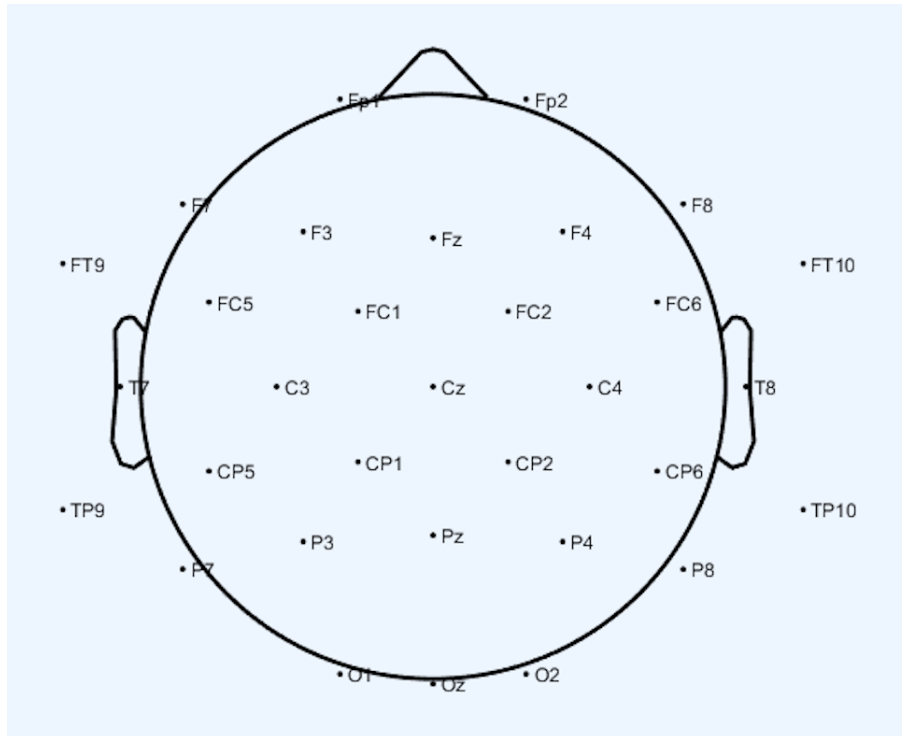


Figure 2.1: Electrode locations of 10-20 system used through this thesis.

this thesis is presented below:

- *Trial*: Each individual image presentation is called a trial.

- *Epoch*: An epoch is a specific time window which is extracted from the continuous EEG signal. Each epoch is time-locked with respect to an event i.e., image stimulus presentation in our case. *This is different from the epoch that is used for training a deep neural network.*
- *ERP*: An ERP signal is the averaged EEG epochs which corresponds to the target stimulus.
- *ERP difference*: An ERP difference signal is the averaged target EEG epochs minus the averaged standard EEG epochs.
- *Trial rejection*: Remove those epochs corresponding to the selected events that contain artifacts e.g., eye blink.

2.2 Neurally Augmented Image Labelling Strategies

EEG data from 9 participants was recorded. Data collection was carried out with approval from Dublin City University’s Research Ethics Committee (DCU REC/2016/099) (see Appendix E). Each participant completed 6 different tasks (INSTR, WIND1, WIND2, UAV1, UAV2 and BIRD). For each task, participants were asked to search for specific target images from the presented images (i.e., an airplane has the role of target in UAV1 and UAV2 tasks, a keyboard instrument is the target for the INSTR task, while a windfarm is the target in WIND1 and WIND2 tasks, and parrot being the target in BIRD task. See Fig. 2.2).



Figure 2.2: Examples of target images in NAILS task. Images from left to right are airplane, windfarm, macaw, keyboard instrument. The size of target and standard are all 256×256 .

Each of the tasks was divided into 9 blocks, where each block contained 180 images (9 targets/171 standards) thus there were 486 target images and 9,234 standard images available for

each participant. Images were presented to participants at a 6 Hz (6 images per second) presentation rate. EEG data was recorded along with timestamping information for image presentation, which was triggered via a photodiode to allow for precise epoching of the EEG signals for each trial [112]. A 32 channel BrainVision actiCHamp at 1000 Hz (1000 samples per second) sampling frequency, using electrode locations as defined by the 10-20 system, was used for EEG acquisition. Epochs were filtered to exclude those with a peak-to-peak amplitude greater than $70 \mu\text{V}$ on EOG and frontal EEG channels. Independent component analysis (ICA) was used alongside a morlet wavelet based analysis to confirm that the remaining epochs did not contain non-neural sources of discriminative information. We used trial rejection for eye-related removal instead of removing eye-related independent components by using ICA because ICA sometimes cannot fully remove the eye-related artifacts [113].

Figure 2.3 shows an example of butterfly plot for one participant. The plot was produced

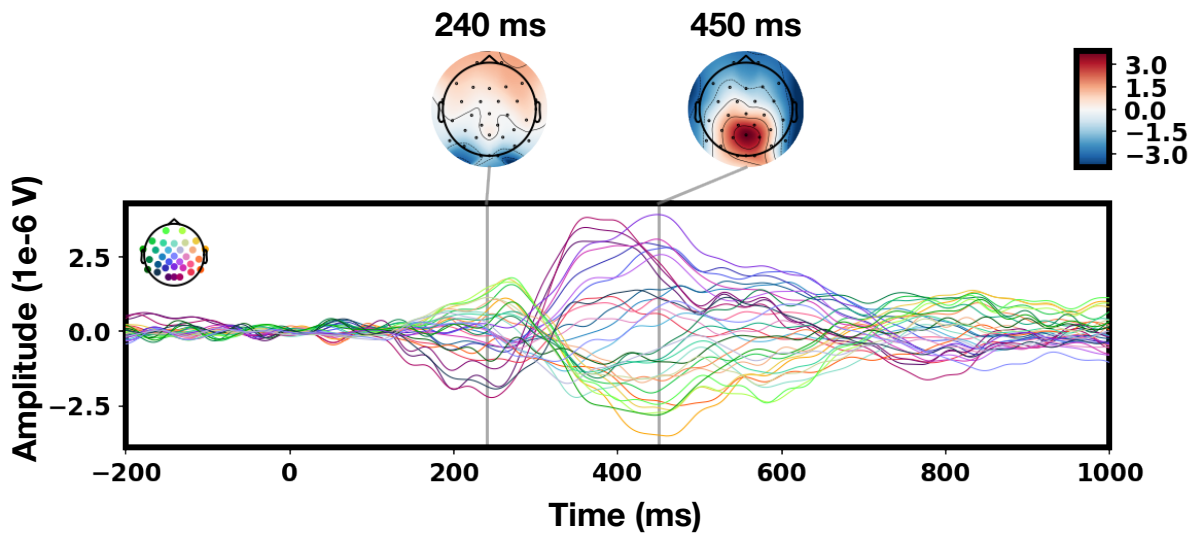


Figure 2.3: Example of ERP difference butterfly plot (target ERPs minus standard ERPs) across all trials after rejecting the eye-related artifacts. Plots were generated using common average reference (CAR). Both an early ERP (e.g., the N200) and a later ERP (e.g., the P300) can be seen in the plot. The N200 can be seen at occipital electrode sites peaking at 240 ms while the P300 can be seen at parietal electrode sites peaking at 450 ms. Channels are color added using the scalp mapping legends in the top left corner.

by the ERPs difference between target ERPs and standard ERPs using all trials (excluding eye-related artifacts). It can be seen that both an early ERP (e.g., the N200) and a later ERP (e.g., the P300) are clearly elicited. The N200 appears at the occipital electrode sites peaking at 240 ms while the P300 is detected at the parietal electrode sites peaking at 450 ms. These two types

of ERPs can be then used by machine learning strategies for this image search task. We also listed the first ten independent components (ICs) of the epochs with eye-related artifacts trials being rejected (see Fig. 2.4). We used first ten ICs for visualization because these components mostly contribute to the raw EEG signals. It can be seen following trial rejection that there is no eye-related ICs, which indicates that the eye-related artifacts (as seen in Fig. 2.5) have been successfully removed.

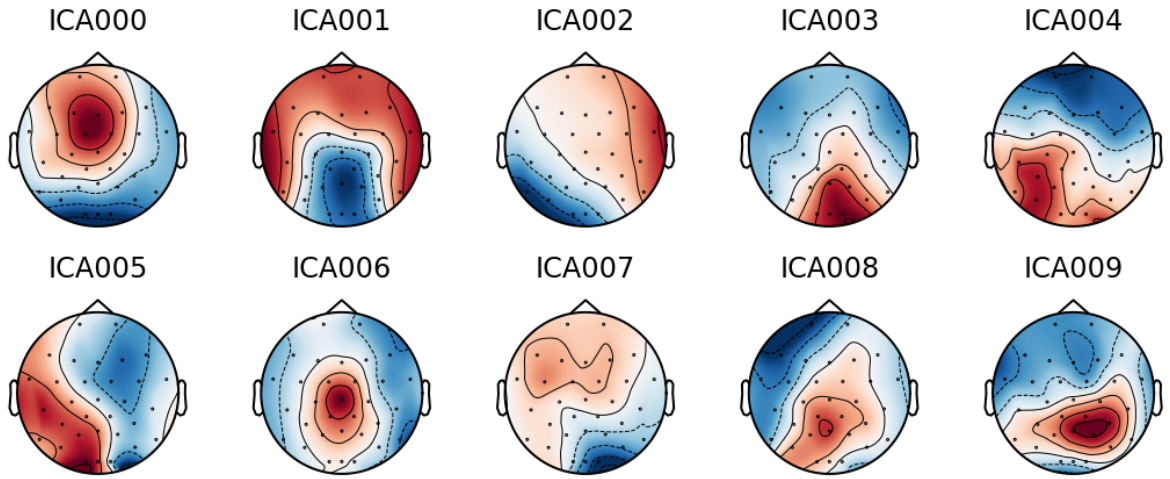


Figure 2.4: First ten ICs of EEG epochs in the NAILS dataset, which has rejected eye-related artifacts trials.

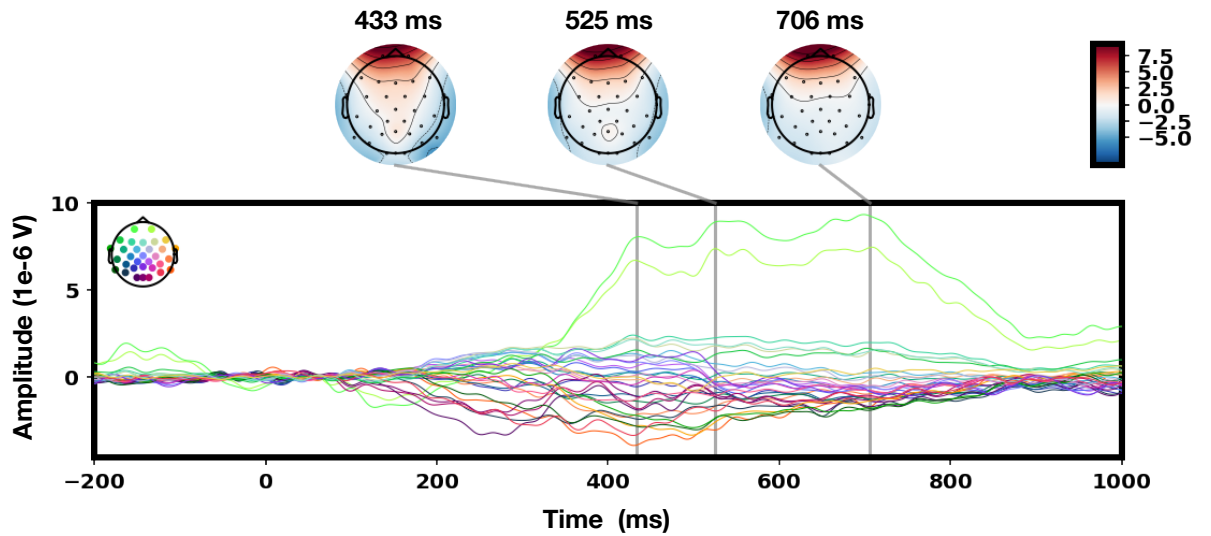


Figure 2.5: An example of eye-related artifacts present in EEG.

2.3 Neural Indices for Face Perception Analysis

Data collection was carried out with approval from Dublin City University’s Research Ethics Committee (DCUREC/2018/115) (see Appendix F). EEG data from 12 participants was recorded. Each participant completed two types of tasks which we call behavioral experiment (BE) task and rapid serial visual presentation (RSVP) task. The sequence of blocks presented in the experiment was: BE → RSVP → BE → RSVP → BE.

The objective of the BE task was to record participants’ behavioral responses (i.e., physical responses by pressing buttons) to each type of image category while in the RSVP task it was to record EEG responses when participants see the rapid presentation of images. The ultimate goal of this study was to compare whether the EEG responses in the RSVP task are consistent with the participants’ behavioral responses in the BE task.

The BE task contained three blocks, where each block contained 90 images (18 images for each face category resulting in 72 face images in total and 18 non-face images) thus there were 216 face images and 54 non-face images in the BE task in total. Participants were presented with one image at a time and asked to press a button corresponding to a “Yes” if they perceived a real face (i.e., belonging to the real face (RFACE) set) or a “No” for anything they perceived as not being a real face (including fake face and non-face). Following each response, feedback was given on whether or not the presented image was indeed a real face to make participants pay more attention to the task. The accuracy (number of correct in labelled images divided by number of presented images) of each participant’s responses was recorded. This recorded accuracy is subsequently referred to the “human perceptual judgment” metric. Figure 2.6 demonstrates an example of ERPs responses in the BE task. It can be seen that the N170 ERP was successfully elicited in this task.

The RSVP task contained 26 blocks. Each RSVP block contained 240 images (6 images for each face category thus 24 face targets in total and 216 non-face images), thus there were 6,240 images (624 face targets/5,616 non-face images) available for each participant. In the RSVP task, image streams were presented to participants at a 4 Hz (i.e., 4 images per second) presentation rate. Participants were asked to search for the RFACE images in this task so as to elicit a P300. We compared the P300 amplitude in the RSVP task to the human perceptual judgment measured in the BE task to determine if they were consistent with each other. Figure 2.7 shows ERPs responses in the RSVP task, where N170 and P300 were both clearly activated in this task.

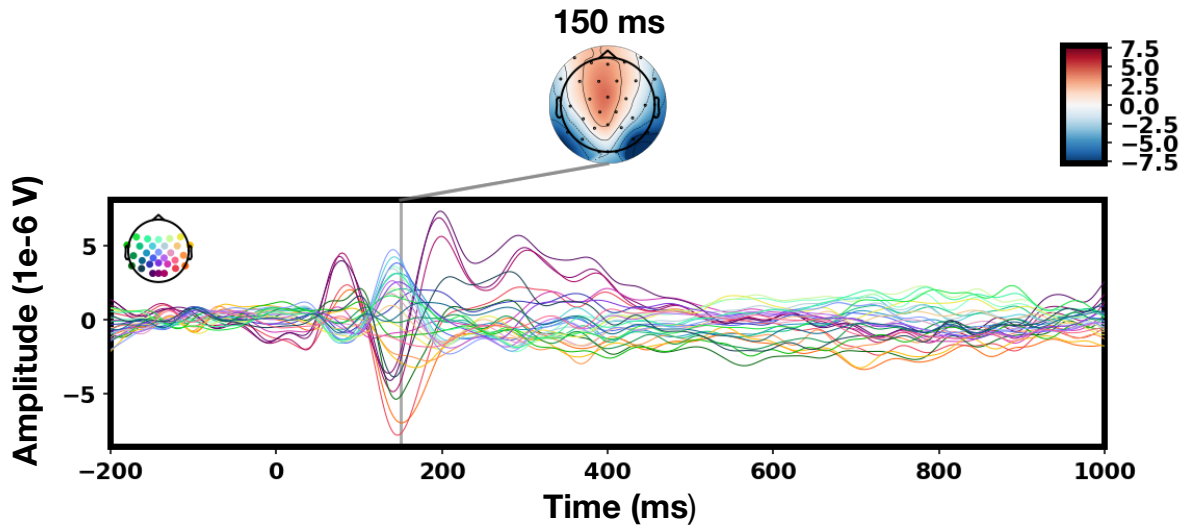


Figure 2.6: Butterfly plot for the ERP difference in the behavioral (BE) task from one participant. N170 was elicited peaking at 150 ms.

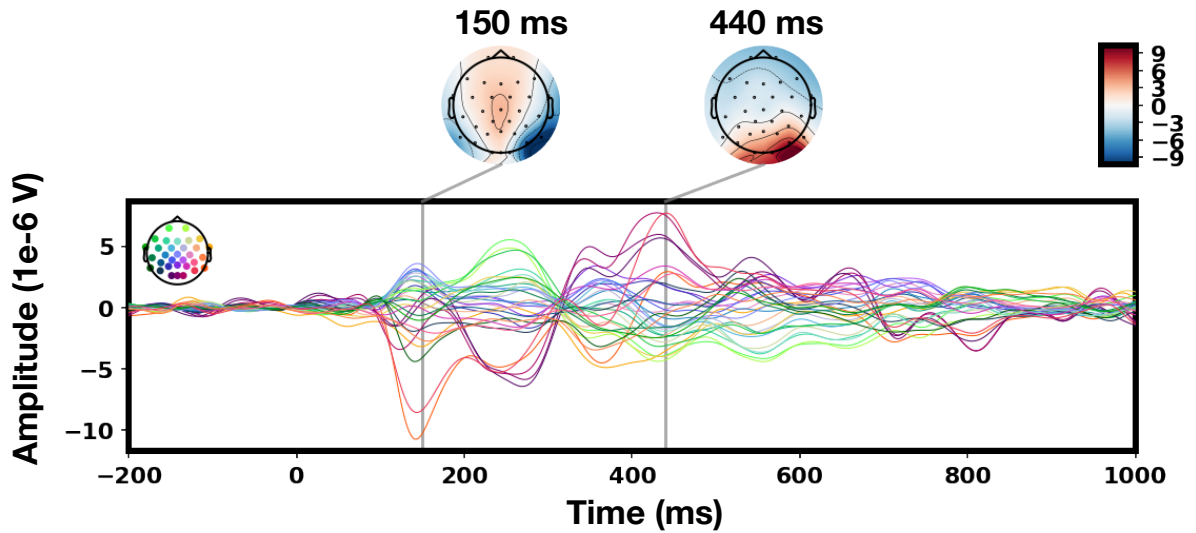


Figure 2.7: Butterfly plot for the ERP difference in the rapid serial visual presentation (RSVP) task from one participant. The N170 was elicited peaking at 150 ms and the P300 was elicited peaking at 440 ms.

EEG was recorded for both of these two types of tasks along with timestamping information for image presentation and behavioral responses (via a photodiode and hardware trigger) to allow for precise epoching of the EEG signals for each trial [112]. EEG data was acquired by using the same amplifier and specification as in the NAILS recording. In this work, a CAR was

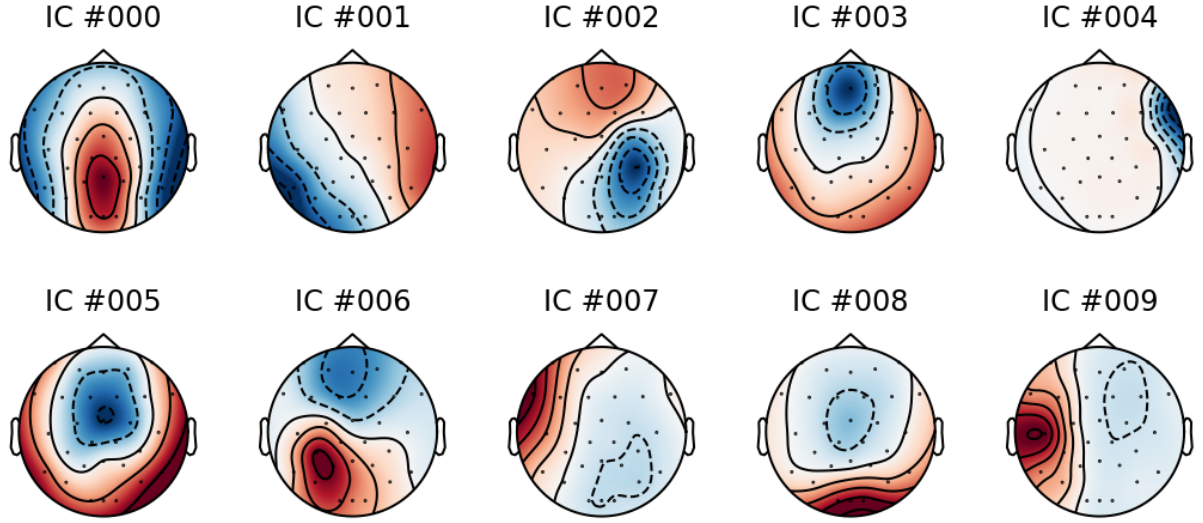


Figure 2.8: First ten ICs of EEG epochs in the NIFPA dataset, to which trial rejection was applied regarding the eye-related artifacts.

utilized and a bandpass filter (e.g., 0.5 Hz – 20 Hz) was applied prior to epoching. EEG data was then downsampled to 250 Hz. Only behavioral responses occurring between 0 s and 1 s after the presentation of a stimulus were used. Trial rejection was carried out to remove those trials containing noise such as eye-related artifacts (via a peak-to-peak amplitude threshold across all electrodes). Figure 2.8 shows the ICs of the epochs with eye artifacts related trials being rejected. It can be seen that there is no eye-related artifacts ICs contained in the first ten ICs.

2.4 Conclusion

In this chapter, we introduced details of the EEG datasets used in this thesis. We have demonstrated some elicited ERPs (the N170, the N200 and the P300) in these two datasets and the pre-processing steps including filtering, artifacts removal etc. The more comprehensive analysis and usage of these two datasets will be discussed in the later chapters.

Chapter 3

Cortically Coupled Computer Vision Processing Methods

***Abstract:** In this chapter, we introduce an architecture for cortically coupled computer vision (CCCV) systems that combines brain-computer interfaces (BCIs) with rapid serial visual presentation (RSVP). Hereafter, we refer to the coupling of the RSVP protocol with electroencephalography (EEG) to support a target-search BCI as RSVP-EEG. Our focus in this chapter is on a review of feature extraction and classification techniques applied in RSVP-EEG. We briefly present the commonly used algorithms and describe their properties based on the literature. We conclude with a discussion on the future trajectory of this exciting branch of BCI research. Work in this chapter has been published as a book chapter in the *Signal Processing and Machine Learning for Brain-Machine Interfaces* [92].*

3.1 Introduction

The rapid serial visual presentation (RSVP) protocol is a method that can be used to extend the brain-computer interface (BCI) approach to enable high throughput target image recognition applications [59, 61, 114]. Using electroencephalography (EEG) signals to label or rank images is of practical interest as many types of images cannot be automatically labeled by a computer [59]. A common example here is to enhance the performance of satellite imagery analysts, by performing selection to get a smaller number of images for later and more detailed inspection [61]. In the RSVP target-search paradigm (see Fig. 3.1) , there is a rapid succession of images presented on screen, in which only a small percentage contain target images. Images

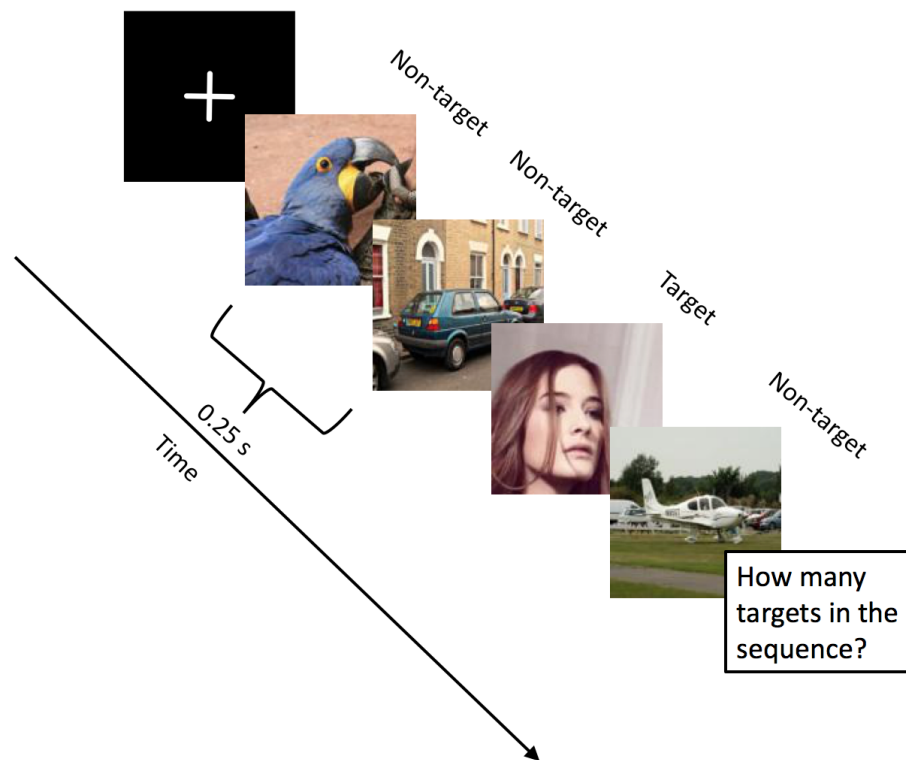


Figure 3.1: RSVP paradigm protocol.

are typically presented to participants at a very fast speed on a monitor (4 – 10 images per second). These infrequent target images are known to elicit the P300 ERP, a type of brain response that has a well-established history of study [39]. The idea is that the participant is unaware when a target stimulus is going to appear, hence its presentation on screen elicits the P300 ERP reflecting the orientation of participant’s attention to the stimuli. These brain activity related responses, when extracted, can be used through application of signal processing and machine learning techniques to enable labeling and/or ranking of images.

In order to use a cortically coupled computer vision (CCCV) system in this way, a participant must be capable of responding with brain activity patterns that can be identified automatically (in a sequential sense). In this regard, the use of an “oddball” paradigm to elicit the P300 ERP responses is ideal as targets searched for tend to be infrequent in many datasets and the response has characteristic features.

Figure 3.2 shows the stages of a typical BCI system. Pre-processing, feature extraction, classification and post-processing are BCI system components. Changes in any one of these components can alter the performance of a BCI system. Pre-processing refers to denoising signals i.e., filtering, artifact rejection, normalization, etc. Feature extraction and classification

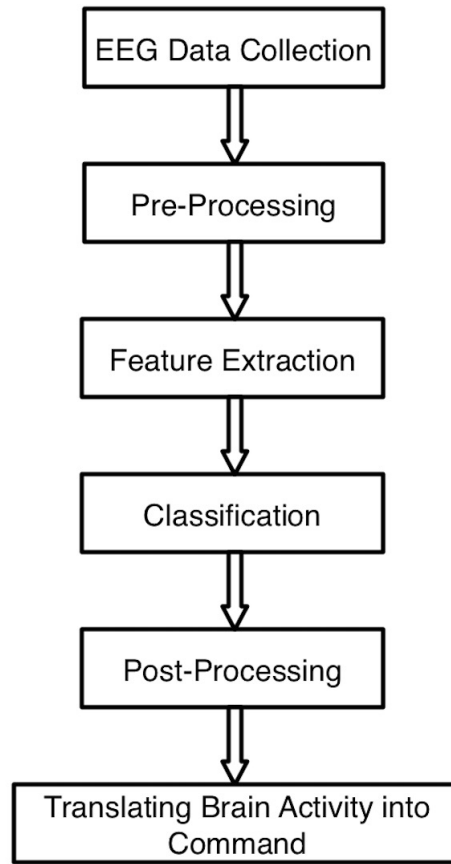


Figure 3.2: Block diagram of a typical BCI system.

both belong to the machine learning section and are essential elements of BCI systems. Post-processing refers to the use of context information to eliminate outliers which can improve the performance of a classifier.

To date, a number of thorough reviews of classification techniques for BCIs have been published [23, 115] but none have been specifically dedicated to the review of feature extraction and classification algorithms used for CCCV research. More broadly, the RSVP target-detection problem is part of a wider field of study that investigates single-trial detection methods [59, 61, 116].

In section 3.2, we give a brief introduction to the RSVP-EEG experimental setup and demonstrate the latency issue that arise when using software triggers (details are presented in Appendix A). We then show several spatio-temporal signals that are typically present in RSVP-EEG have discriminative properties e.g., the P300. It should be noted that the common objective of all BCI systems is to maximize classification accuracy rather than providing an interpretation of the underlying neurophysiology. Finally, we describe the pre-processing step and the problems of RSVP-EEG data availability in the literature which has an impact on algorithm

development, reproducibility and benchmarking.

In section 3.3, we outline common strategies used to extract useful features from RSVP-EEG data, namely spatial filters achieved using (un-)supervised techniques such as independent component analysis (ICA) that aims to find a linear representation of non-Gaussian data so as to maximize a statistical independence metric, time-frequency representation which decomposes RSVP-EEG to the time-frequency domain and some other feature extraction methods. Spatial filtering allows for dimensionality reduction by transforming high dimensional spatial EEG to a subspace according to different optimization objectives e.g., improving the SNR. Reducing data dimensionality in this way is often essential to overcoming issues with having relatively fewer training examples than when there are a high number of features — a scenario commonly referred to as the “curse of dimensionality” [117].

In section 3.4, we explore a number of commonly used classification strategies covering both linear and non-linear techniques. Linear classification techniques include linear discriminant analysis (LDA), Bayesian linear regression (BLR), logistic regression (LR) and support vector machine (SVM). Non-linear classification approaches are mainly focused on artificial neural networks (ANNs).

In summary, this chapter aims to survey the different feature extraction and classification techniques used in CCCV research and to identify their critical properties, shortcomings and advantages. It also provides newcomers to the CCCV area with an introduction — a framework within which an analysis of RSVP-EEG data can be understood.

3.2 Overview of RSVP Experiments and EEG Data

3.2.1 RSVP Experiment for EEG Data Acquisition

Data acquisition for RSVP-EEG experiments is typically carried out using two computers. One computer is used for stimulus presentation and the other for recording and monitoring of EEG data from participants. A typical setup is shown in Fig. 3.3. The EEG amplifier is used for recording the EEG signals measured from participants. When displaying the image sequence to participants, a timestamp for each image must be recorded and aligned with the multi-channel time-series EEG captured on the acquisition computer. These are commonly referred to as triggers.

In CCCV research, triggers are normally sent from the presentation software (e.g., Psy-

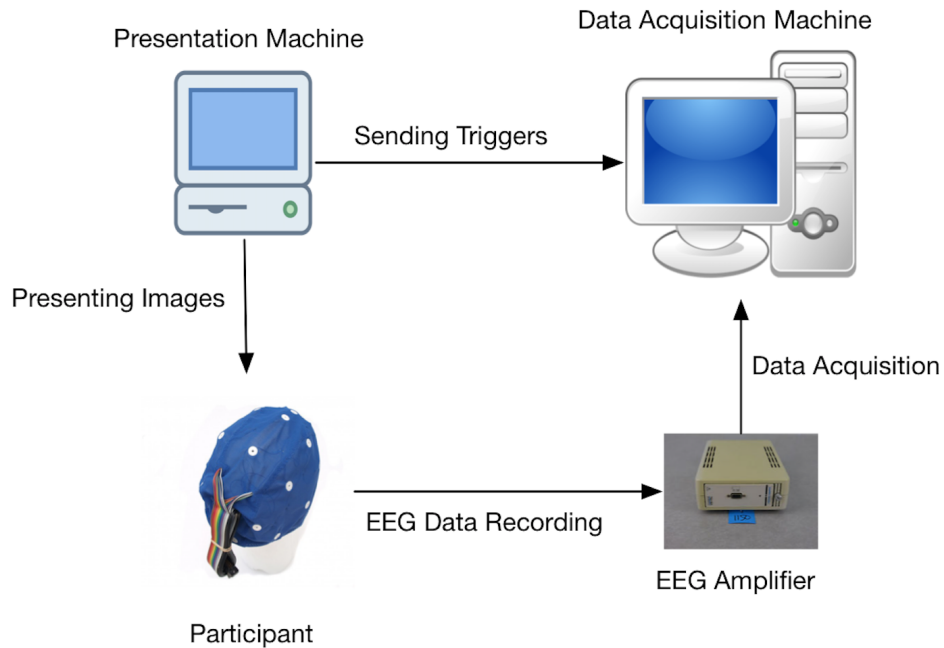


Figure 3.3: RSVP experiment set up.

choPy [118] and E-prime [119]) either to the EEG acquisition device directly [116, 120] via a physical port or to the acquisition software [121]. Due to the fast presentation speeds involved with RSVP-EEG, careful attention needs to be given to ensure that stimulus presentation timings are correct. We describe a study to investigate in a representative RSVP implementation whether or not software-derived stimulus timing can be considered an accurate reflection of the physical stimuli timing [112] (details are included in Appendix A).

3.2.2 Brief Introduction to RSVP-EEG Pattern

The most widely used pattern in the EEG signals acquired in a CCCV system is the P300 component. The P300 is a complex endogenous response that can be subdivided into a novelty-related P3a component and a posterior occurring component commonly encountered in RSVP search, which is referred to as P3b [39]. The discovery of the P300 arose from the confluence of increased technological capability for signal averaging applied to human neuroelectric measures and the impact of information theory on psychological research [122]. The P300 is often characterized by its amplitude and latency, where it is defined as the largest positive-going peak in the time range of 300 ms – 800 ms following a stimulus presentation. Its latency and amplitude can vary depending on stimulus modality, task conditions, participant age, and other

factors [39]. This is why it is necessary to learn participant and task specific machine learning models. Figure 3.4 shows an example of a P300 (P3b) response at channel Pz in one RSVP

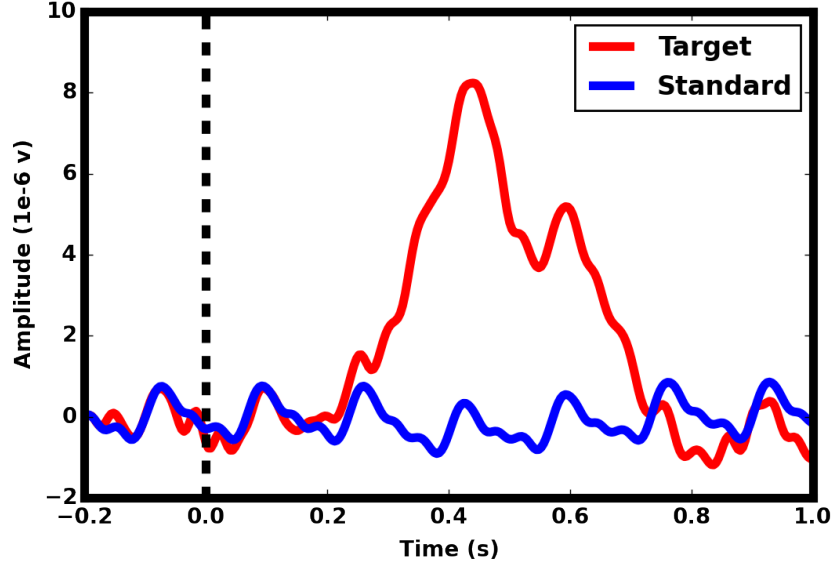


Figure 3.4: The P300 response example at the Pz channel in an RSVP experiment, the EEG signal has been band-passed between 0.1 Hz and 30 Hz.

search task. It can be seen that the P300 peak occurs at around 480 ms. The periodic oscillation that can be seen in the ERPs average for standard images is due to a steady-state visual evoked potential (SSVEP) response [123]. To facilitate presentation of the concepts involved in RSVP-EEG, we made use of the NAILS dataset [104], which has already been introduced in Chapter 2. This EEG dataset was part of a collaborative benchmarking task in 2017 [102].

It is worth noting that not all participants display the stereotyped P300 responses, with some displaying characteristics such as low amplitude components leading to unfavorable SNR properties. Reasons for this will not be explored here but further information can be found in [124].

Single-trial detection methods are not restricted solely to BCI-related contexts, and are often found as part of a researcher’s toolkit in developing an analysis pipeline when working with neurophysiological data. What such methods typically strive to accomplish is striking a balance between the neurophysiological interpretability of a model and its complexity. Transparent models are more likely to lend themselves to meaningful interpretations. While a strategy of enforcing simplistic models with few parameters may aid in interpretability, the purpose of RSVP-EEG is to maximize information throughput as defined by some performance metric.

In reality, a BCI can make use of non-neural signal sources present in the EEG. For exam-

ple, some participants may, without realizing, blink their eyes upon seeing a target in a RSVP experiment, where the eye-blink will impart a large voltage deflection in the EEG. We are primarily concerned, however, with direct neural signal sources in this chapter and strategies to utilize these. Hence, trial rejection is used to support this aim.

3.2.3 RSVP-EEG Data Pre-processing and Properties

Pre-processing of some kind is generally a required step before any meaningful interpretation or use of the EEG data can be realized. Pre-processing typically involves re-referencing (changing the reference channel), filtering the signal (by applying a bandpass filter to remove environmental noise or to remove activity in non-relevant frequencies), epoching (extracting a time epoch typically surrounding the stimulus onset), trial/channel rejection (to remove those containing artifacts) etc. See [69] for further information. In RSVP-EEG, a CAR or mastoid reference is often used. A bandpass filter (e.g., 0.1 Hz – 30 Hz) is commonly applied in RSVP-EEG. The EEG signal is preferably analyzed as epochs (i.e., the whole EEG data is cut by using a fixed time window e.g., 0 ms – 1000 ms corresponding to each trigger onset) and each segment is named as an epoch. These epochs can then be used for analysis e.g., feature extraction and classification.

The presence of many artifacts such as those related to muscle movements in the EEG signal can be sometimes removed by using a bandpass filter as the frequencies of interest in RSVP-EEG do not always detrimentally overlap. During RSVP-EEG experiments, it can be very common for eye-blink behavior to occur in response to target images. This perhaps arises as a result of the participant withholding eye-blinks until a target is seen. While this may be favorable for improving the detection rate of targets, without inspection of the data it may lead to erroneous conclusions on what discriminative information is actually driving the performance of a BCI. One common strategy to investigate this involves identifying any spatial components in the EEG signal related to eye-blinks via independent component analysis (ICA) to determine if it is trial-locked to targets in any way. Additionally, ICA allows for such activity to be in part attenuated. In this thesis, we use ICA to confirm the absence of eye-related artifacts after trial rejection rather than to remove this activity. An investigation of commonly used strategies (and subtle pitfalls) can be found in [113].

Before extracting features from RSVP-EEG data, some critical properties of EEG signals have to be considered concerning the design of a CCCV system:

3.2. Overview of RSVP Experiments and EEG Data

- Low signal-to-noise ratio (SNR): EEG in a noninvasive BCI has an inherently poor SNR and task-related ERPs are typically overwhelmed by strong ongoing EEG background activity in single trials;
- Curse of dimensionality: In a CCCV system, EEG data can have high dimensionality spanning space and time. However typically limited training sets are available especially considering the target image class which are usually infrequent;
- Overlapping epochs: There is substantial overlap between adjacent target epochs and standard epochs because of short interstimulus interval (ISI) used in the RSVP paradigm;
- Imbalanced datasets: Target images are overwhelmed by standard images in a RSVP application which leads to an imbalanced classification problem.

These critical properties have to be considered before feature extraction. The last one can be overcome through cost-sensitive learning [125] while the first three are inherent challenges in the design of a CCCV system.

A key difference between the RSVP-EEG paradigm and other ERP paradigms is that the former requires single-trial detection in the presence of overlapping epochs. Traditional ERP analysis typically computes a grand average ERPs where phase-locked activity in the signal remains after averaging whilst other non-locked background activity increasingly attenuates as more trials are averaged [126]. For example, the P300 speller is an ERP paradigm that has been a benchmark for the P300-BCI system. In this paradigm, each desired symbol is spelt several consecutive times by a participant where the epochs corresponding to each row/column are averaged over the trials. This averaging process is able to improve the EEG SNR for the system because averaging reduces the contribution of random background EEG oscillations [41]. This repetition of an image stimulus is not always applicable in the RSVP-EEG paradigm because it can introduce unintentional behaviors such as a participant attending to an image due to it being a salient repetition rather than it being a target. In single-trial detection, low SNR is a challenge for the detection of discriminative ERP activity. Furthermore, the overlapping epochs problem may contribute to overfitting when training a machine-learning model [127]. In summary, low SNR and overlapping epochs are two challenging problems for RSVP-EEG when compared to other ERP paradigms.

3.2.4 Performance Evaluation Metrics

A machine learning model's performance can be evaluated by a variety of evaluation metrics. Area under the receiver operating characteristic curve (AUC) is widely used as it illustrates the discriminative ability of a binary classifier system as its discrimination threshold is varying [128]. One may want to adjust this threshold for example to optimize for fewer false positives at the cost of more false negatives. The AUC is the most widely used evaluation metric in RSVP-EEG research [59, 61, 116]. However, the AUC score may not be suitable when evaluating some real-world systems because it gives an unified measure of the performance of a classifier across all potential thresholds and in effect sidesteps the issue of the impact of threshold selection.

Balanced accuracy (BA) is well-suited for evaluating RSVP-EEG systems that utilize binary classifications [129]. The BA can be calculated as below

$$BA = \frac{1}{2} (\text{sensitivity} + \text{specificity}) \quad (3.1)$$

where $\text{sensitivity} = \frac{TP}{TP+FN}$ and $\text{specificity} = \frac{TN}{TN+FP}$. Choosing evaluation metrics critically depends on the application. For example, if the classification system is used to rank target above standard images, then the AUC would be the preferred evaluation metric. If the classification system is designed to give a binary classification (target vs standard), then the BA can be a good evaluation metric. Both the AUC and the BA are robust to targets/standards ratio imbalances in dataset.

3.3 Feature Extraction Methods Used in CCCV Research

The challenge for feature extraction methods is to find intrinsic characteristics of the EEG signals that relate to certain cognitive responses. Feature extraction in BCI systems plays an important role since it can significantly affect the SNR and the classification strategy used, which in turn determines the performance of the BCIs.

This section focuses on the feature extraction of RSVP-EEG from three aspects: (1) Spatial filtering (supervised and unsupervised); (2) Time-frequency representation; and (3) Other feature extraction methods.

3.3.1 Spatial Filtering

Supervised Spatial Filtering

As mentioned in the previous section, RSVP-EEG data suffers from low SNR and often high spatial dimensionality. Spatial filtering is an efficient technique for mitigating these concerns. In the area of BCI research, xDAWN [130], beamformer [131] and common spatial pattern (CSP) [132] are widely used for generating supervised spatial filters. In this chapter, we focus on three methods for generating spatial filters, which are xDAWN, CSP and LDA beamformer. For xDAWN, the goal is to maximize signal-to-signal-plus noise ratio (SSNR) whereas for CSP, the goal in terms of maximizing the variance (power) of EEG signals between target trials and standard trials. LDA beamformer is used for source signal reconstruction where it maximizes the SNR.

Problem Formulation Let $\mathbf{X} \in \mathbb{R}^{N_c \times N_t}$ be an EEG epoch corresponding to an image stimulus, where N_c is channel number and N_t is epoch time length. The problem (as seen in equation (3.2)) of spatial filtering is to find a set of projection vectors (each comprised of weights for each channel) $\mathbf{w} \in \mathbb{R}^{N_c \times N_f}$ (N_f is the number of components) to project \mathbf{X} to a subspace, where \mathbf{w} is calculated by different algorithms i.e., xDAWN, beamformer, CSP etc.

$$\mathbf{X}_{\text{sub}} = \mathbf{w}^\top \mathbf{X} \quad (3.2)$$

Common Spatial Pattern (CSP) CSP generates sets of channel weights that can be used to project multi-channel EEG data to a low-dimensional subspace, where this transformation can maximize the variance of two-class signal matrices. Let $\mathbf{X}_+(i) \in \mathbb{R}^{N_c \times N_t}$ and $\mathbf{X}_-(i) \in \mathbb{R}^{N_c \times N_t}$ be the i th event-locked EEG epochs (N_c is the channel number and N_t is the time length) in two experimental conditions i.e., $\mathbf{X}_+(i)$ for the target image condition and $\mathbf{X}_-(i)$ for the standard image condition. Covariance matrices in the two conditions can be estimated as

$$\Sigma_c = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_c(i) \mathbf{X}_c^\top(i) \quad (c \in \{+, -\}) \quad (3.3)$$

where $\Sigma_c \in \mathbb{R}^{N_c \times N_c}$ and n is the number of corresponding trials belonging to each condition. The CSP optimization problem can be formulated as

$$\underset{\mathbf{w} \in \mathbb{R}^{N_c \times N_f}}{\operatorname{argmax}} \quad \frac{\mathbf{w}^\top \Sigma_+ \mathbf{w}}{\mathbf{w}^\top \Sigma_- \mathbf{w}} \quad (3.4)$$

This optimization problem is given by the simultaneous digitalization of the two covariance matrices. This can be achieved by solving the generalized eigenvalue problem

$$\Sigma_+ \mathbf{w} = \lambda \Sigma_- \mathbf{w} \quad (3.5)$$

Note: The objective of CSP is to maximize the variance in one class while minimizing the variance in the other class. Maximizing the variance in this way corresponds to maximizing the frequency-power of target-related activity in the signal. When using CSP, multiple spatial filters will be obtained and cross validation is normally deployed to choose a spatial filter(s). This strategy can be adapted to use multiple different band-passed versions of the same EEG signal epoch to leverage different sources of discriminant information present across different frequencies (that often also differ in spatial characteristics). CSP is widely applied in motor imagery based BCIs [133]. In Yu's work, CSP has been applied for producing spatial filters for CCCV [120].

xDAWN The xDAWN algorithm has been successfully applied in the P300 speller BCI application [130]. The basic goal of xDAWN is to enhance the SSNR of the responses corresponding to the target stimulus. Let recorded signals be $\mathbf{X} \in \mathbb{R}^{N_t \times N_c}$, where N_t is the time length of recorded EEG signals and N_c is the number of channels. It considers the following model

$$\mathbf{X} = \mathbf{D}\mathbf{A} + \mathbf{H} \quad (3.6)$$

where $\mathbf{D} \in \mathbb{R}^{N_t \times N_e}$ (N_e is the number of temporal samples of ERPs corresponding to the target stimulus) is the real Toeplitz matrices and $\mathbf{A} \in \mathbb{R}^{N_e \times N_c}$ is the ERPs response to the target. \mathbf{D} has its first column elements set to zero except for those that correspond to a target stimulus onset and \mathbf{H} is the on-going EEG activity.

The problem statement for xDAWN becomes how to estimate the spatial filter for (equation (3.6)) such that the synchronous response is enhanced by spatial filtering

$$\mathbf{X}\mathbf{w} = \mathbf{D}\mathbf{A}\mathbf{w} + \mathbf{H}\mathbf{w} \quad (3.7)$$

where $\mathbf{w} \in \mathbb{R}^{N_c \times N_f}$ (N_f is the number of spatial filters). The optimized solution can be achieved

by

$$\hat{\mathbf{w}} = \arg \max_{\mathbf{w}} \frac{\text{Trace}(\mathbf{w}^\top \hat{\mathbf{A}}^\top \mathbf{D}^\top \mathbf{D} \hat{\mathbf{A}} \mathbf{w})}{\text{Trace}(\mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w})} \quad (3.8)$$

where $\hat{\mathbf{A}}$ is the least squares estimation of response \mathbf{A} . More details about the computation method can be found in [130].

Note: As distinguished from CSP, the numerator in xDAWN in equation (3.8) is the ERP response rather than target EEG epochs i.e., ERP response is the mean value of target EEG epochs. xDAWN aims to enhance the SSNR of the response corresponding to the target stimulus and it is originally designed for enhancing the P300 evoked potential for the P300 speller BCI [130]. Similar to CSP, xDAWN generates multiple spatial filters as well. It is suggested to use cross-validation to determine the number of spatial filters. In recent published work, xDAWN has been applied to CCCV for spatial filtering [134].

LDA Beamformer LDA beamformer has been proposed to maximize the SNR of EEG in a way which is robust to correlated sources [135]. The generation of a spatial filter using LDA beamformer is comprised of three steps: (1) Spatial pattern estimation; (2) Covariance matrix estimation; and (3) Spatial filter optimization. Let column vectors $\mathbf{p}_1 \in \mathbb{R}^{N_c \times 1}$ and $\mathbf{p}_2 \in \mathbb{R}^{N_c \times 1}$ be the spatial pattern of a specific component in two different experimental conditions, where N_c is the number of channel. We denote the difference pattern as $\mathbf{p} := \mathbf{p}_1 - \mathbf{p}_2$ and the covariance matrix $\Sigma \in \mathbb{R}^{N_c \times N_c}$. The optimization problem for the LDA beamformer can be stated as

$$\underset{\mathbf{w}}{\text{minimize}} \quad \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^\top \mathbf{p} = 1 \quad (3.9)$$

and the optimal solution can be determined as

$$\mathbf{w} = \Sigma^{-1} \mathbf{p} (\mathbf{p}^\top \Sigma^{-1} \mathbf{p})^{-1} \quad (3.10)$$

where $\mathbf{w} \in \mathbb{R}^{N_c \times 1}$.

Note: The traditional beamformer is used to localize the source. The LDA beamformer has an ability as it is able to estimate the spatial pattern for the source (i.e., the spatial pattern is a type of spatial information). As distinguished from the previous two methods, the LDA

beamformer method generates an optimal spatial filter (only one spatial filter) that maximizes the SNR (i.e., from equation (3.9), $\mathbf{w}^\top \mathbf{p}$ is constrained to 1 and the minimization of the cost aims to optimize the SNR). One optimal projection vector may not be able to fully capture all available information from the original EEG epoch due to the sources of variability such as different spatial characteristics of early and late target-discriminative ERPs across tasks and participants. Therefore, multiple time window LDA beamformers (where the researcher trains the LDA beamformer in multiple time windows) are often applied for improved performance with RSVP-EEG data.

So far we have introduced three supervised approaches for generating useful spatial filters. After applying the spatial filter(s) to the original epoch, it can be appreciated that the dimensionality of the projected subspace has been reduced significantly and that this subspace signal may have different properties (optimized SSNR for xDAWN, optimized SNR for LDA beamformer, maximum difference of variance between two classes for CSP) depending on which algorithm has been used for generating the spatial weightings. This projected subspace can then be used as the basis of a feature set for training a practical classifier. It is worth noting that the overall effect of employing spatial filtering methods is an improvement in the SNR, a reduction in computational cost and a more favorable situation for many classification algorithms that suffer issues with high dimensional feature vectors (particularly when few training examples are available). These are desirable properties of a signal processing pipeline for CCCV.

Unsupervised Spatial Filtering

Independent Component Analysis EEG source activity refers to the time-varying far-field potentials arising within an EEG source and volume-conducted to the scalp electrodes. The recorded EEG signals are then, according to this interpretation, the summation of neural activity, contributions of non-brain sources such as scalp muscle, eye movement, and cardiac artifacts, plus (ideally small) electrode and environmental noise [136]. Successful separation of contributions from these non-neural activity related sources can improve the SNR of the signals of interest. ICA is a technique that can aid here and can be used to find linear representations such that time-series signals obtained via its components' projections are statistically independent from each other (or as independent as possible). Such a representation is capable of capturing the essential structure of the data in many applications, including feature extrac-

3.3. Feature Extraction Methods Used in CCCV Research

tion and signal separation [137]. Essentially, ICA produces a matrix of spatial filters. ICA has been widely applied to EEG signal fields for denoising [138] and artifact removal [139]. Figure 3.5 illustrates the characteristics of three independent components (ICs) corresponding

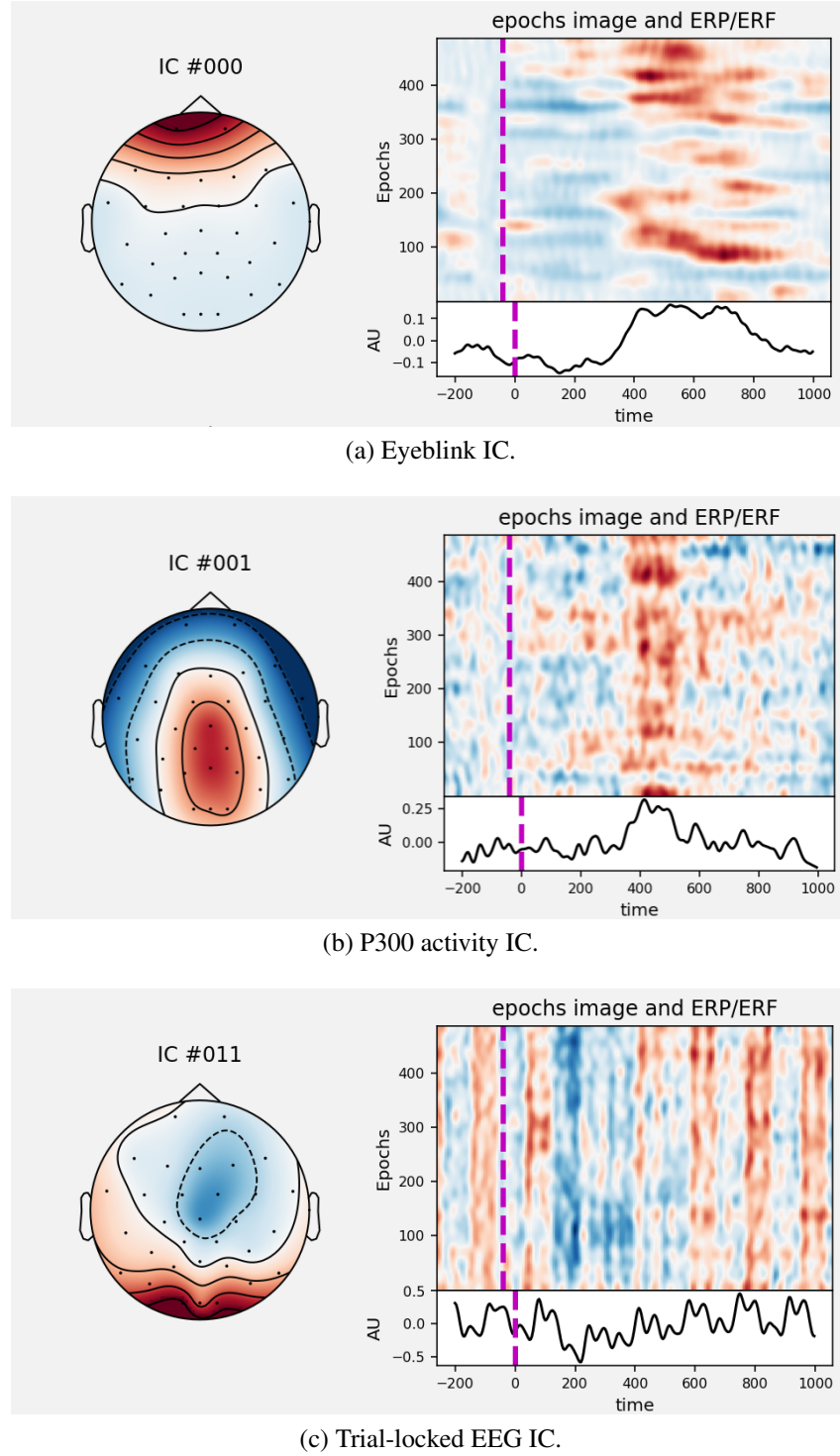


Figure 3.5: Examples of ICA components (left) and ERP images (right) for eye blink related activity (A), posterior P300 activity (B) and other trial-locked EEG activity (C). Results are generated using one participant's dataset (from the NAILS dataset) and only includes RSVP target-related trials.

to eyeblinks, P300 activity and other trial-locked ERP activity respectively. From the IC plots (left), it is noticeable that different signal sources have different ICs localizations i.e., eyeblink is distributed frontally while the P300 is localized posteriorly. From an ERP image and ERP time series plot (right hand side top and bottom plots), it can be seen that eyeblink activity and P300-related activity are locked to target stimuli. Moreover, it can be seen that the eyeblink time-locked activity is noticeably prominent at around 400 ms which is close to the time region where we also see P300 activity. This indicates that this participant sometimes blinked his/her eyes when presented with a target image likely as a result of an active effort to suppress eyeblinks up to that point in case they missed a target. Eyeblink artifacts in such instances can potentially be beneficial for the classification process. However, we are going to remove these (via trial rejection) as we only consider signals of direct neural origin in this exposition. It is worth emphasizing here that ICA successfully resolved the signal into a distinct neuro physiologically interpretable source in this instance. Published work by Bigdely-Shamlo and his colleagues, demonstrated ICA successfully applied to CCCV generating ICs and independent time-course templates for each IC. These independent time-course templates were selected as features for training the classifier [61] quite successfully.

Principal Component Analysis Principal component analysis (PCA) is a statistical technique which uses eigenvalue decomposition to convert a set of correlated variables into a set of linearly uncorrelated variables where each of the resultant variables is referred to as a principal component (PC) [140]. For multivariate datasets, notably data in a high dimensional space, PCA can be particularly effective for dimensionality reduction. PCA has been applied in EEG signal analysis for dimensionality reduction [141] and the production of spatial filters [142]. In CCCV literature, PCA has only been applied for feature dimensionality reduction rather than extracting informative features to date [61, 143].

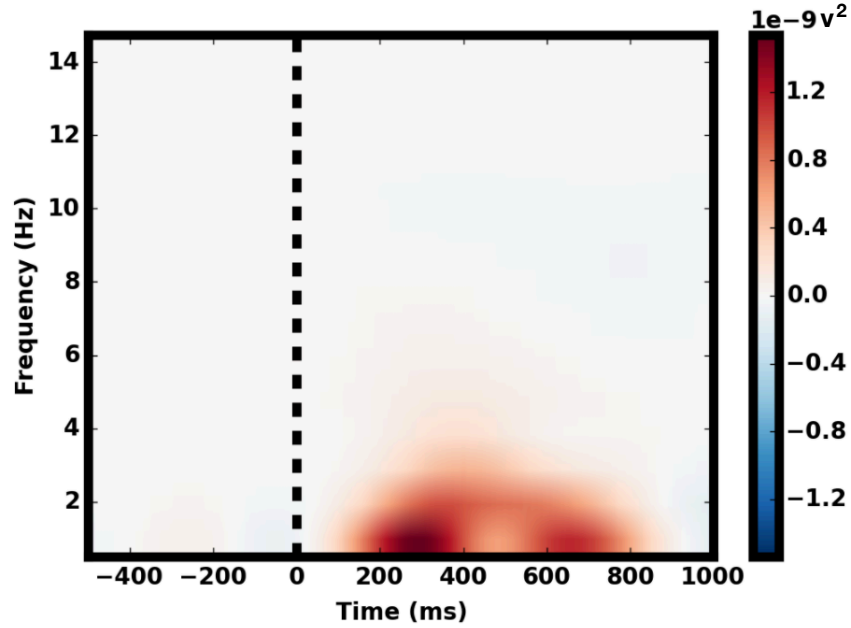
3.3.2 Time-frequency Representation

Feature extraction in BCIs can be achieved in the time domain, the frequency domain and the combined time-frequency domain. In the time domain, time regions coinciding with ERPs such as the P300 are used when extracting features for single-trial event detection [114]. Frequency domain features such as the amplitudes of μ (8-13 Hz) and β (14-26 Hz) are widely used in sensorimotor control BCIs as it has been shown changes occur in these when a participant imagines (or engages) in certain types of movements [144]. However, frequency domain features have

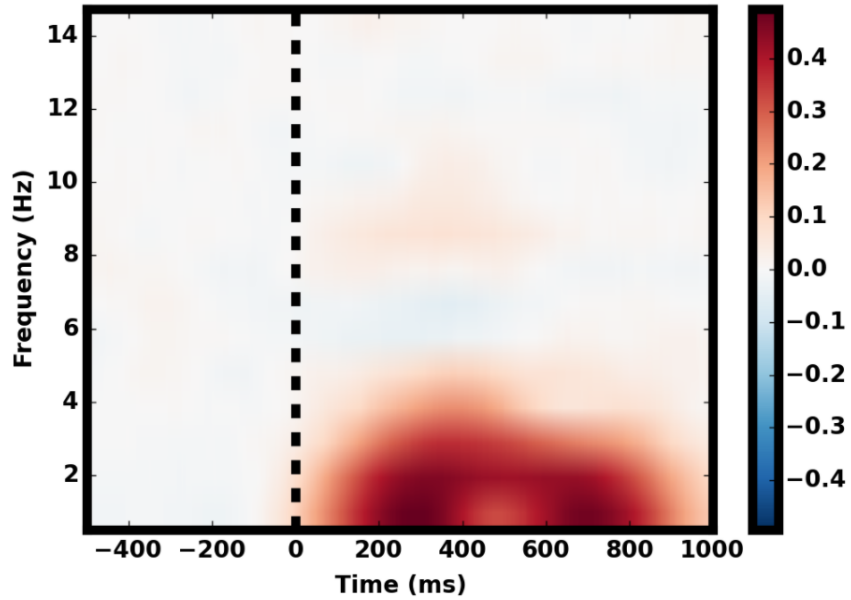
not been used in the CCCV systems in the literature to date because it does not have critical information such as time and topographic distribution. Time-frequency representations can be generated using methods such as short time Fourier transform (STFT) and morlet wavelet transforms. The wavelet transform method is often preferred over STFT in many instances as it produces a time-frequency decomposition of a signal over a range of characteristic frequencies that separates individual signal components more effectively than the STFT method does. A number of other properties of the source signal such as stationarity assumptions should be considered when utilizing one approach over another [126]. With the time-frequency techniques applied, two useful features can be then generated for analyzing neural dynamics, which are inter-trial coherence (ITC) [126] and event-related spectral perturbation (ERSP) [145]. ITC is calculated as

$$\text{ITC} = \left| \frac{1}{n} \sum_{r=1}^n e^{ik_{tf}r} \right|, \quad \text{ITC} \in [0, 1] \quad (3.11)$$

where e^{ik} is from Euler's formula and provides the complex polar representation of a phase angle k on trial r at a time-frequency representation point tf (time & frequency). ITC measures the extent to which a distribution of phase angles at instantaneous point across trials is non-uniformly distributed in polar space [126]. ITC is useful for measuring phase-locked neural dynamics (e.g., the P300 and N170) corresponding to stimuli. Compared to ERPs, ERSP provides insights for neural dynamics in both time domain and frequency domain. A set of common types of time-frequency features found in RSVP-EEG has been proposed in Meng's work [60]. In Fig. 3.6 we show the time-frequency mean power generated using target EEG epochs from the NAILS dataset. It can be seen that both high power and strong ITC appear in low frequencies (0 Hz – 6 Hz) and in two time regions (200 ms – 400 ms and 600 ms – 800 ms). The discriminative ERSP related activity appears in both an early time region and a late time region. The time-frequency representation strongly depends on the type of the image stimulus and the experimental environment. To summarize, time-frequency representation is more used for analyzing and interpreting neural dynamics e.g., visualize a specific neural activity elicited by a typical type of experimental stimulus. Features generated by this technique for classification can be less effective.



(a) Event-related spectral perturbation (ERSP). Power was normalized by subtracting mean power from the baseline.



(b) Inter-trial coherence.

Figure 3.6: The ERSP representation example corresponding to target images at the Pz channel for averaging 9 participants in an RSVP experiment using Morlet wavelet transform, the EEG signal has been band-passed between 0.1 Hz and 30 Hz. EEG epochs used baseline from -500 ms to 0 ms (i.e. subtract the mean for the baseline signals).

3.3.3 Other Feature Extraction Methods

In Huang's work, EEG signals from the stimulus onset to 500 ms post-stimulus were extracted for each channel and concatenated to form a feature vector [116]. This resulted in each trial containing 32×129 features (where 32 is the number of channels and 129 is the number of time points). This strategy of building feature vectors as a concatenation over time regions (and channels) of interest is commonly found in the CCCV literature and often yields good results. Hierarchical discriminant component analysis (HDCA) has been proposed in [146], where this method estimates EEG signatures of target detection events using multiple linear discriminators, each is trained at a different time window relative to the image onset. Since EEG signals contain both spatial and temporal information, a spatio-temporal representation for RSVP-EEG data has been proposed by Alpert [143]. This representation is divided into two steps: (1) Linear discriminant analysis (LDA) is applied at each timestamp to produce the spatial weights and a spatial weight matrix is then used for mapping original epoch to a new space; and (2) PCA is then used for dimensionality reduction based on the temporal domain i.e., for each independent channel.

3.3.4 Summary

Feature extraction is an essential step when designing a BCI system because pertinent features can significantly improve performance of the resulting classifier and additionally it can significantly reduce the computational cost. In CCCV, discriminative ERP-related activity often occurs in both early and late time region e.g., N200 and P300. Feature extraction for CCCV is best designed by considering these ERPs' properties.

3.4 Survey of Classifiers Used in CCCV Research

This section surveys the classifiers used for recognition of target images and standard images in CCCV systems. Due to the fact that the non-linear problem has not been well explored in RSVP-EEG data in the literature so far, this section is divided into linear classifiers and neural network classifiers. Since deep learning technology is very popular currently in other application domains such as computer vision and natural language processing, we introduce some deep learning methods in the neural network section.

3.4.1 Linear Classifiers

Linear classifiers are widely used for designing BCIs applications due to their good performance, often simple implementation and low computational complexity. Four main linear classifiers will be introduced in this section, namely, linear discriminant analysis (LDA), Bayesian linear regression (BLR), logistic regression (LR) and support vector machine (SVM). In this section, we consider our model as

$$y = \mathbf{w}^\top \mathbf{x} + b \quad (3.12)$$

where y is classifier output, \mathbf{x} is the feature vector and b is the threshold.

Linear Discriminant Analysis

LDA is a supervised subspace learning method which is based on the Fisher criterion and it is equivalent to least squares regression (LSR) if the regression targets are set to $\frac{n}{n_1}$ for samples from class 1 and $-\frac{n}{n_2}$ for samples from class 2 (where n is total number of training samples, n_1 is the number of samples from class 1 and n_2 is the number of samples from class 2) [73]. It aims to find an optimal linear transformation \mathbf{w} that maps \mathbf{x} to a subspace in which the between-class scatter is maximized while the within-class scatter is minimized in that subspace. The optimization problem for LDA is to maximize the cost function as below

$$J = \frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_W \mathbf{w}} \quad (3.13)$$

where \mathbf{S}_B is the between-class scatter and \mathbf{S}_W is the within-class scatter. Regularization is often applied in order to avoid the singular matrix problem of \mathbf{S}_W [147]. Figure 3.7 shows the LDA implementation on two different classes (red and black dots) with equal covariance and different mean values. The solid line is the projected subspace \mathbf{w} where these two classes will be projected by LDA. This transformation enables the best separation between two classes on the subspace \mathbf{w} . Details of the method can be found in Duda's book [148].

LDA has very low computational complexity which makes it suitable for online BCI systems. As mentioned earlier, classification of RSVP-EEG data suffers from the imbalanced dataset problem. In Xue's work [149], he showed that there is no reliable empirical evidence to support that an imbalanced dataset has a negative effect on the performance of LDA for gener-

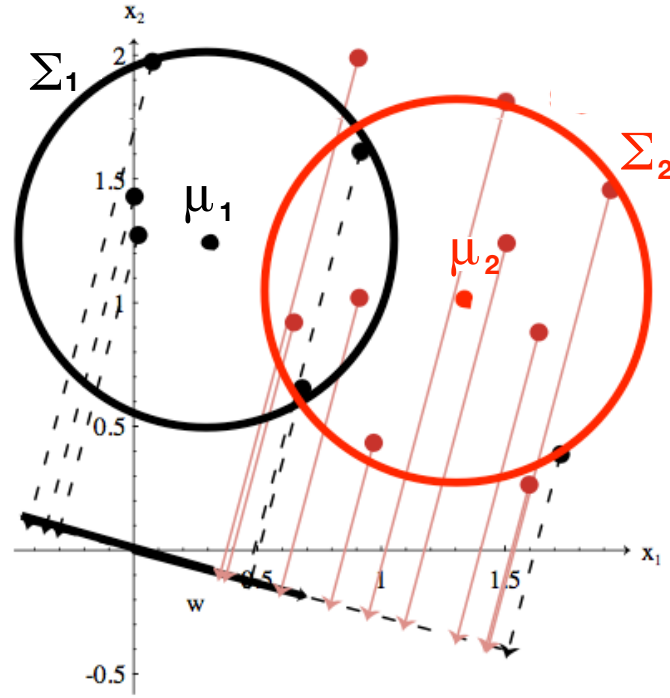


Figure 3.7: Projection of two different classes onto a line by LDA. Σ_1 and Σ_2 are the covariances of two classes while μ_1 and μ_2 are the mean values of two classes. [148].

ating the linear transformation vector. Consequently, LDA is suitable and has been successfully used in CCCV [60, 61]. LDA can suffer in terms of performance in the face of outliers in the training data. For this reason, regularization strategies are typically employed but are not covered here [147].

Bayesian Linear Regression

BLR, also named Bayesian linear discriminant analysis (BLDA), can be seen as an extension of LDA or LSR. In BLR, regularization for parameters is used for preventing overfitting caused by high dimensional and noisy data. BLR assumes the parameter distribution and target distribution are both Gaussian [73]. We introduce LSR as a starting point for the description of BLR. Given the linear model in equation (3.12), the input $\mathbf{X} \in \mathbb{R}^{m \times n}$ and the output $\mathbf{y} \in \mathbb{R}^{n \times 1}$ (m and n are number of parameters and number of samples), the solution of LSR can be stated as

$$\mathbf{w} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{y} \quad (3.14)$$

Note that $y = \frac{n}{n_1}$ for class 1 and $y = -\frac{n}{n_2}$ for class 2 here (threshold can be determined by adding a column with all one as the first column in \mathbf{X}) [73]. LSR does not consider the

parameter distribution in this case and it maximizes the likelihood. For BLR, it considers the parameter distribution and maximizes the posterior. Given the prior target distribution $p(\mathbf{y}) \sim \mathcal{N}(\mu, \beta^{-1})$ and parameter distribution $p(\mathbf{w}) \sim \mathcal{N}(0, \alpha^{-1}\mathbf{I})$ (where β and α are the inverse variance), BLR gives the optimal estimation for the parameter

$$\mathbf{w} = \beta(\beta\mathbf{X}\mathbf{X}^\top + \alpha\mathbf{I})^{-1}\mathbf{X}\mathbf{y} \quad (3.15)$$

It can be seen that the optimization of BLR is added with the prior information of parameter and data. Hence, the optimization depends on the hyperparameters β and α . In real-world applications, the hyperparameters can be tuned using cross validation or the maximum likelihood solution with an iterative algorithm [73, 150]. BLR has been proven to have very good and robust performance in BCI research [151, 152].

Logistic Regression

LR models the conditional probability as a linear regression of feature inputs. Considering the linear regression model of RSVP-EEG data in equation (3.12), the logistic model can be constructed as

$$p(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x} + b}} \quad (3.16)$$

The optimization problem of the LR can be constructed by minimizing the cost function as below

$$J(\mathbf{w}, b) = -\frac{1}{n} \sum_{i=1}^n [y_i \log(p(\mathbf{x}_i)) + (1 - y_i) \log(1 - p(\mathbf{x}_i))] \quad (3.17)$$

where $y \in \{0, 1\}$ and n is the sample number of two classes. LR can be modified by penalizing different cost terms to each class and the cost function can be modified as below

$$J(\mathbf{w}, b) = -\frac{1}{n_0 + n_1} \sum_{i=1}^{n_0+n_1} [n_0 y_i \log(p(\mathbf{x}_i)) + n_1 (1 - y_i) \log(1 - p(\mathbf{x}_i))] \quad (3.18)$$

where n_0 and n_1 are the numbers of standard and target image clips respectively.

LR is part of a broader family of generalized linear models (GLMs), where the conditional distribution of the response falls in some parametric family, and the parameters are set by the linear predictor. LR is the case where the response is binomial and it can give the prediction of the conditional probability estimation. LR is easily implemented and has been successfully applied for CCCV research [64, 153].

Linear Support Vector Machines

A linear support vector machine (linear SVM) aims to select the hyperplane which maximizes the margins (i.e., the distance from the nearest training samples). In order to overcome the imbalanced classification problem, a linear weighted support vector machine (WSVM) is proposed [154]. Linear SVMs can be used for linear and non-linear classification by using the “kernel trick”. This consists of mapping data to other spaces using a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$. For linear classification, a kernel function can be chosen as $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$. For nonlinear classification, Gaussian kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\delta^2})$, is widely used in the classification area. Compared to the previous three approaches, the computational time for a SVM (with kernel version) increases dramatically with increasing training samples i.e., it has $\mathcal{O}(n^3)$ computational complexity [155].

Linear SVMs have a small number of hyperparameters which need to be tuned manually and this is often done by using cross validation. There is already considerable use of linear SVM in CCCV research [108, 120, 156].

Other Machine Learning Algorithms

Here we have introduced four linear classification algorithms that are widely used in the CCCV research area. There are numerous machine learning algorithms available currently and we encourage readers to experiment with them. Scikit-learn [157] is a machine learning library in Python and it provides lots of machine learning algorithm implementations. Linear classification methods can be found in the GLMs list.

Note: For the four classifiers above, every method involves hyperparameters except LDA (unless using a regularized version that relies on some parameter selection). We suggest using random search or maximum likelihood estimation to determine these hyperparameters. K-fold validation can be used for the evaluation of each set of tuned hyperparameters. Importantly, classifiers’ performance should be evaluated on a withheld validation set such as an experimental block that does not overlap with the data used for model training or hyperparameter selection.

Summary

To summarize, we have introduced four types of linear classifier from the viewpoints of different optimization objectives. LDA aims to find the subspace which gives the best separation between two classes after projection. BLR uses prior information about data distribution and weights, constraining the estimated weights close to zeros, which helps to stop overfitting. SVM finds the optimal hyperplane maximizing the margins and it can be applied to non-linear cases by using the “kernel trick”. LR can predict the conditional probability. We recommend to use LDA (a regularized form) and BLR for CCCV because of their low computational cost and good performance. Recent work in CCCV research has shown that BLR outperforms LDA and SVM [152].

3.4.2 Artificial Neural Networks

Artificial neural networks (ANNs) are yet another category of classifiers that are increasingly used in BCIs research. An ANN comprises several artificial neurons that can enable non-linear decision boundaries.

This section is divided into two parts. The first describes the multi-layer perception (MLP), the most widely used ANN and then some deep learning techniques are introduced.

Multi-layer Perceptron

A MLP is minimally comprised of three layers of neurons, namely an input layer, one or several hidden layers and an output layer [158] as seen in Fig. 3.8. In each hidden layer, each neuron is connected to the output of each neuron in the previous layer and its output is the input of each neuron in the next layer. Figure 3.8 illustrates a MLP with four inputs, one hidden layer with 10 neurons and output layer (bias terms is not covered in this case). There are 50 connections totally in this case ($4 \times 10 + 10$). Considering a CCCV system, there is only one output in the output layer. Parameters in MLP can be updated in the direction of the negative gradient, where the gradient can be efficiently calculated by using backpropagation [159]. A number of what are called gradient descent optimization algorithms exist for this purpose [160].

ANNs and MLP are very flexible classifiers that can be applied to a great variety of problems because they are universal approximators [23]. Hence, MLP can be applied for almost all machine learning problems including binary classification or multi-class classification or mod-

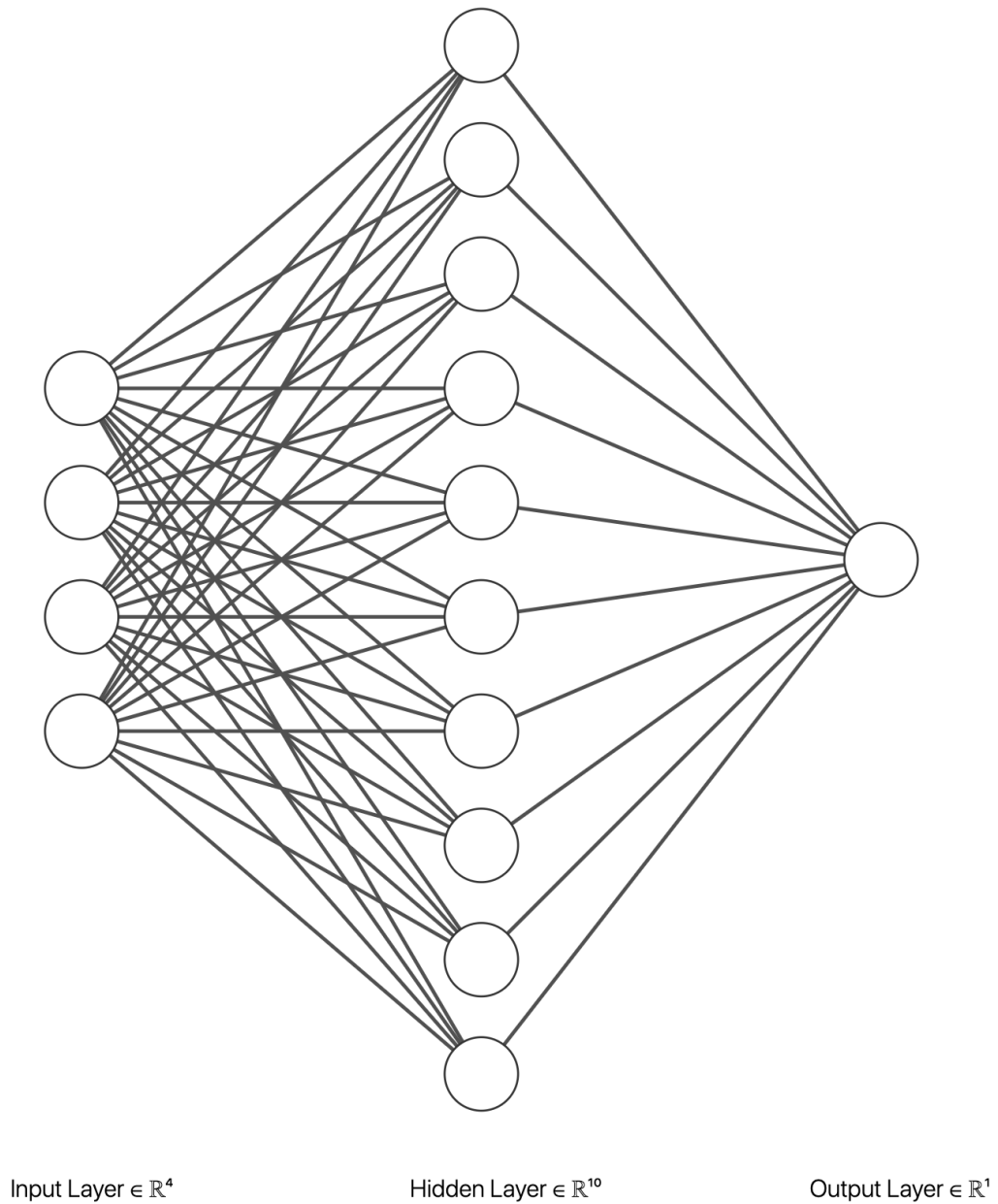


Figure 3.8: An example of a MLP architecture.

eling. The main disadvantage of MLP is overfitting [161] due to the limited training samples especially for targets in RSVP-EEG data. Therefore, one has to be careful when designing a MLP architecture and regularization is often required [162].

Some Deep Learning Techniques

Modern deep learning provides a powerful framework for supervised learning [74]. With more layers and more neurons in a layer, a DNN can represent increasingly complex non-linear patterns and it has been successfully applied to many fields including computer vision [163], nat-

ural language processing [164], etc.

Since deep learning implementations in the area of EEG are still rare, we will introduce three representative methods in the deep learning field, namely convolutional neural networks (CNNs), recurrent neural networks (RNNs) and deep belief nets (DBNs).

Convolutional Neural Networks A CNN is a type of DNN that employs a mathematical operation called convolution specialized for processing a grid of values where the arrangement of the values is not arbitrary such as is the case with an image where pixels near to any one pixel tend to be correlated in a meaningful way [165]. CNNs are simply neural networks that use convolutions in place of general matrix multiplications in at least one of their layers [74]. CNNs have been tremendously successful in many practical applications [166–168]. CNNs leverage three properties that improve learning, namely sparse interactions, parameter sharing and equivariant representations [74]. Sparse interactions are accomplished by the convolution operation while choosing the kernel smaller than the input size. This property enables meaningful features to be extracted from input data. Parameter sharing refers to the fact that each member of the kernel is used at every position of the input with the same parameters. Equivariant representations means that if the input changes, the output changes in the same way [74].

Figure 3.9 demonstrates a basic CNN architecture for EEG data classification. Since a CNN

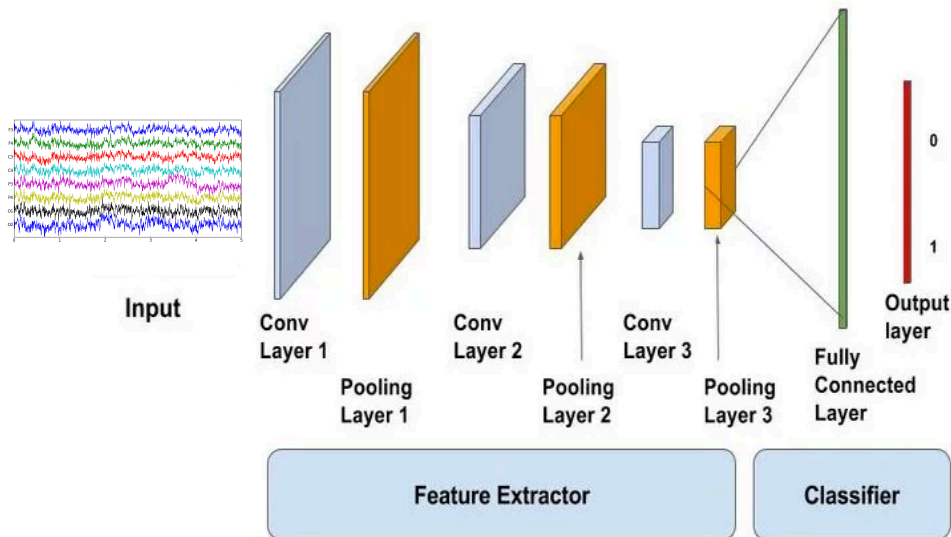


Figure 3.9: An example of CNN architecture for EEG classification.

is capable of extracting features from the input data automatically, CNNs have become the most widely used deep learning architecture in RSVP-EEG research [99, 100, 103, 134]. All of these

works have shown that CNNs are effective in combining the spatial filtering and the classification steps in a unified way. CNNs were also the winning solutions to the NAILS competition in the 13th NTCIR conference [127] which show better performance than traditional methods. The reason that CNNs perform effectively on the EEG classification task is that the property of EEG data is similar to image data at some extent. Both EEG and image data share a common characteristic in that they have a spatial arrangement e.g., signals tend to correlate when near each other in an image or on the scalp. The spatial information of the two dimensional EEG (number of channels \times number of time points) contains both channel location and time region. For example, P300 in RSVP-EEG is normally detected at parietal cortex and appears in 300 ms – 600 ms time region, which is very similar to an object appearing at specific location in an image. Thus CNNs are good at extracting such a type of spatial information from the presented EEG data. We list details of four CNNs architectures that have been proposed in the literature in Table 3.1. The architecture of DeepConvNet and ShallowConvNet in work [101] has been explained in detail in [103]. The architecture of EEGNet2 [127] is similar to the EEGNet [103]. So we have not included details of those CNNs architectures here.

Recurrent Neural Networks Different from CNNs specialized for processing a grid of values, a RNN is a family of neural networks which is designed for processing sequential data such as language data [86]. RNNs process the sequence which contains vectors $\mathbf{x}(t)$ with the time step index t ranging from 1 to τ [74].

There are several implementations of RNNs in the EEG signals analysis and classification [170–172]. However, RNNs implementations for a CCCV system have not been stated in the literature so far. Since EEG is sequential and the P300 has a temporal property, it is an open question if RNNs can be used effectively for a CCCV system.

Deep Belief Nets A DBN is a generative graphical model which comprises multiple layers of latent variables (“hidden units”), with connections between the layers but not between units within each layer [173]. It provides an efficient way to learn a multiple-layered restricted Boltzmann machine (RBM) [174].

A DBN has shown efficacy for a CCCV system from Ahmed’s work and it can extract discriminant features from RSVP-EEG data as well [175].

3.4. Survey of Classifiers Used in CCCV Research

Conv nets	Layer	Input	Operation	Filter	Stride	Output	Total parameters
Frontiers-15 [100]	1	(63, 32, 1)	Conv1D (Activation: ReLU)	(1, 32, 1, 96)	(1, 1)	(63, 1, 96)	11621601
		(63, 1, 96)	Maxpool1D	(3, 1)	(2, 1)	(31, 1, 96)	
	2	(31, 1, 96)	Conv1D (Activation: ReLU)	(6, 1, 96, 128)	(1, 1)	(26, 1, 128)	
		(26, 1, 128)	Maxpool1D	(3, 1)	(2, 1)	(12, 1, 128)	
	3	1536	Fully connected (Activation: ReLU)	N/A	N/A	2048	
	4	2048	Dropout	N/A	N/A	2048	
	5	2048	Fully connected (Activation: ReLU)	N/A	N/A	4096	
SPIE-16 [99]	6	4096	Dropout	N/A	N/A	4096	3076717
	7	4096	Fully connected (Activation: Sigmoid)	N/A	N/A	1	
	1	(250, 32, 1)	Conv1D (Activation: ReLU)	(64, 1, 1, 128)	(4, 1)	(250, 1, 96)	
		(47, 32, 128)	BatchNorm	N/A	N/A	(47, 32, 128)	
		(47, 32, 128)	Maxpool1D	(3, 1)	(2, 1)	(23, 32, 128)	
	2	(23, 32, 128)	Conv1D (Activation: ReLU)	(4, 1, 128, 128)	(2, 1)	(10, 32, 128)	
		(10, 32, 128)	BatchNorm	N/A	N/A	(10, 32, 128)	
		(10, 32, 128)	Conv1D (Activation: ReLU)	(1, 32, 128, 512)	(1, 1)	(10, 1, 512)	
	3	(10, 1, 512)	BatchNorm	N/A	N/A	(10, 1, 512)	
		(10, 1, 512)	Maxpool1D	(3, 1)	(2, 1)	(4, 1, 512)	
EMBC-17 [169]	4	(4, 1, 512)	Conv1D (Activation: ReLU)	(4, 1, 512, 256)	(1, 1)	(1, 1, 256)	6681
		(1, 1, 256)	BatchNorm	N/A	N/A	(1, 1, 256)	
	5	256	Fully connected (Activation: ReLU)	N/A	N/A	500	
	7	500	Dropout	N/A	N/A	500	
	8	500	Fully connected (Activation: ReLU)	N/A	N/A	500	
	9	500	Dropout	N/A	N/A	500	
	10	500	Fully connected (Activation: Sigmoid)	N/A	N/A	1	
EEGNet [103]	1	(63, 32, 1)	Conv1D (Activation: ReLU)	(1, 32, 1, 8)	(1, 1)	(63, 1, 8)	1596
	2	(63, 1, 8)	Conv2D (Activation: ReLU)	(48, 1, 8, 16)	(1, 1)	(16, 1, 16)	
	3	256	Fully connected (Activation: Sigmoid)	N/A	N/A	1	
	1	(250, 32, 1)	Conv1D	(1, 32, 1, 16)	(1, 1)	(250, 1, 16)	1596
		(250, 1, 16)	BatchNorm	N/A	N/A	(250, 1, 16)	
		(250, 1, 16)	ELU	N/A	N/A	(250, 1, 16)	
		(250, 1, 16)	Transpose(2, 0, 1)	N/A	N/A	(16, 250, 1)	
		(16, 250, 1)	Dropout	N/A	N/A	(16, 250, 1)	
		(16, 250, 1)	Conv2D (padding: same *)	(2, 32, 1, 4)	(1, 1)	(16, 250, 4)	
	2	(16, 250, 4)	BatchNorm	N/A	N/A	(16, 250, 4)	
		(16, 250, 4)	ELU	N/A	N/A	(16, 250, 4)	
		(16, 250, 4)	Maxpool2D	(2, 4)	(2, 4)	(8, 62, 4)	
		(8, 62, 4)	Dropout	N/A	N/A	(8, 62, 4)	
		(8, 62, 4)	Conv2D (padding: same)	(8, 4, 4, 4)	(1, 1)	(8, 62, 4)	
	3	(8, 62, 4)	BatchNorm	N/A	N/A	(8, 62, 4)	
		(8, 62, 4)	ELU	N/A	N/A	(8, 62, 4)	
		(8, 62, 4)	Maxpool2D	(2, 4)	(2, 4)	(4, 15, 4)	
	4	(4, 15, 4)	Dropout	N/A	N/A	(4, 15, 4)	
		240	Fully connected (Activation: Sigmoid)	N/A	N/A	1	

* For Convolutional layer implementation in Tensorflow, the default value for zero padding is set to "valid". Here we use the "same" choice from the original paper.

Table 3.1: CNNs architectures in the literature. The architecture of DeepConvNet and ShallowConvNet in work [101] has been explained detailly in [103]. The architecture of EEGNet2 [127] is similar to the EEGNet [103].

Summary

In this section, we have introduced the use of ANNs techniques for classification. The first part introduced a MLP framework which is the classic ANN framework. The MLP has a simple implementation and it is flexible but easily suffers from issues related to overfitting.

Deep learning is very popular and has achieved great success in many real-world applications but often requires very large volumes of data. With better data acquisition and more advanced generative models [65], it is possible to train better deep network models for RSVP-EEG leveraging very large datasets.

The main difficulty with EEG signals is the potentially very large feature dimensionality when considering all combinations of channel, frequency and time features. This makes feature

extraction a very complex step that must cater to inter-subject and inter-task variability. In traditional CCCV systems, feature extraction and classification are always separated. Deep learning provides a potentially unified way to accomplish this.

3.5 Conclusion

In this chapter, we have introduced the main components of a typical CCCV including RSVP-EEG data acquisition, data pre-processing, feature extraction and classification. We focused on the machine learning architecture (feature extraction and classification) as it is the most important part of a typical BCI system.

We have shown that discriminative patterns in RSVP-EEG data can appear in different time regions (both early and late) i.e., the P300 is not the only ERP being elicited in the RSVP paradigm. Therefore, we suggest to design a feature extraction method that takes into account the properties of the discriminative ERPs for a given task. A good feature extraction method can not only improve a classifier's performance in the later stage but also reduce the computational cost. In this chapter we introduced existing feature extraction methods used in the literature so far. We stated the objective of each method and a direct comparison between those methods will be part of future work.

The other part of the machine learning architecture in a CCCV system is the classifier. We divided our discussion on classifiers into both linear classifiers and ANNs. The choice of the classifier remains difficult and depends mainly on the number of available trials and feature vector dimensionality. Linear classifiers remain popular as they have low computational complexity, are easy to implement and have good performance on classification accuracy. ANNs possibly outperform linear classifiers with a large number of trials as DNNs are able to capture high level features related to the variability of the EEG signals across participants and over time. However, acquiring EEG data is time consuming and the variability in the EEG of a specific participant can change over time, which indicates that the number of trials for a CCCV system is limited.

Therefore, the choice of classification method should be capable of training a model with a limited amount of available data. In this aspect, linear classifiers are more preferable than ANNs due to fewer parameters in the model which in turn can help to prevent overfitting on the noisy and limited RSVP-EEG data. Here we suggest to use LDA and BLR for CCCV research

as they are easy to implement, efficient and have good performance.

The area of RSVP-EEG stretches back well over a decade and there has been significant progress in this time. With the emergence of deep learning approaches, computer vision recognition applications are able to perform at or even above a human level, which raises questions about whether people are still needed to perform image labeling tasks. We believe that when labeled image datasets are limited, these computer vision systems may not perform very well as a typical component to their success is the availability of very large labeled image datasets. In this way, RSVP-EEG may assist in more efficiently labeling large datasets of image content to support this process. Similarly, many image labeling tasks may require subjective (or expert) knowledge about the image that cannot be easily learned by a deep learning architecture but that may be readily detected when using a CCCV system. We see these systems as being able to work in a synergistic manner rather than competitively.

Chapter 4

Spatial Filtering Pipelines for Cortically Coupled Image Classification

Abstract: *This chapter evaluates spatial filtering pipelines for a cortically coupled computer visual (CCCV) system. We propose a novel spatial filtering method called multiple time window LDA beamformers (MTWLB). Then we provide a comprehensive comparison of nine spatial filtering pipelines using three spatial filtering schemes, namely, MTWLB, xDAWN, common spatial pattern (CSP) and three linear classification methods linear discriminant analysis (LDA), Bayesian linear regression (BLR) and logistic regression (LR). Finally, we compare the performance of time-course source signal reconstruction between xDAWN and MTWLB. The area under the curve (AUC) is used as measurement for the performance of classification in this chapter. Results reveal that MTWLB (92.2%) and xDAWN (92.4%) spatial filtering techniques enhance the classification performance of the pipeline but CSP (88.2%) does not. Results also support the conclusion that LR (92.7%) can be effective for a CCCV system if discriminative features are available. We also demonstrate the efficacy of using MTWLB for reconstructing the time-course source signal compared to xDAWN regarding signal-to-noise ratio (SNR) and classification performance, which suggests MTWLB is a better fit to the event-related potentials (ERPs) study for rapid serial visual presentation (RSVP) based EEG data. Part of this chapter has been published in the Journal of Brain-Computer Interfaces [93].*

4.1 Introduction

We have established already that there is an interest in using EEG to help in the development of alternative methods for image search. This can be done by examining participants' neural signals in response to image presentation [59, 61, 176]. Using modern signal processing and machine learning techniques, RSVP can be coupled with single-trial ERP detection to enable image search brain-computer interface (BCI) applications [103, 111], which is known as cortically coupled computer vision (CCCV). Single-trial ERP detection for a RSVP paradigm presents the following challenges which have been addressed in the previous chapter:

Challenge 1. Low signal-to-noise ratio (SNR): Amplitudes of ERP components are often much smaller than those of spontaneous EEG components and task-related ERP components are typically overwhelmed by strong ongoing EEG background activity in single trials and so cannot be normally visually recognized in the raw EEG trace [18]. Traditional methods analyze ERPs by averaging across several task-related trials in order to reduce or eliminate spontaneous EEG components [69].

Challenge 2. Curse of dimensionality: RSVP-EEG data can have high dimensionality spanning both space and time. Moreover, ERPs vary greatly across participants and experimental tasks [69]. In order to capture relevant ERPs, it is necessary to choose a time window large enough for epoching which involves the time region in which ERPs might appear. Moreover, the training sets available for machine learning purposes are typically modest in size and normally contain relatively few instances of the responses evoked by the infrequent (by design) target image class.

Challenge 3. Overlapping epochs: The strength of the RSVP paradigm is that the rate of the stimulus sequence increases the upper limit of potential information transfer rates in BCI applications. However, a relatively large time window has to be set for epoching in order to capture the ERPs. Therefore, there is substantial overlap between adjacent target epochs and standard epochs because of the short inter-stimulus interval (ISI) used in the RSVP paradigm.

We do not address the imbalanced datasets challenge here (mentioned in Chapter 3) as it only affects the threshold of a classifier (i.e., it requires additional efforts to overcome this problem such as cost-sensitive learning [125]). We focus on the performance of spatial filtering approaches and the performance of classifiers with respect to features extracted by spatial filters). The overall performance of a CCCV pipeline can be evaluated by using AUC score (not sensitive to the imbalanced datasets), which has been widely applied in the literature of

CCCV [59–61]. In this chapter, we consider a pipeline combining spatial filtering and linear classification as this is the most widely used pipeline configuration in a CCCV system.

There are several potential signal pre-processing techniques that may increase the detection of relevant single-trial ERPs including time-frequency feature extraction and hierarchical discriminant component analysis (HDCA) [60, 146]. However, spatial filtering techniques are more efficient when a high-density EEG dataset is available. In this chapter, we focus on spatial filtering for signal pre-processing as this is the predominant approach used in the CCCV research. Similarly, we are using a high-density EEG dataset (32 channels). Spatial filtering focuses on enhancing task-related information contained in EEG signals. It plays an important role in BCI research because it can enhance the discriminant information present in EEG signals whilst reducing the overall data dimensionality. This property in turn mitigates the curse of dimensionality problem when applying machine learning strategies [23]. Spatial filtering has been shown to enhance detection accuracy with a P300 speller paradigm [130]. However, classification pipelines without spatial filtering have been proposed for single-trial ERPs detection. These methods include widely used linear classifiers such as linear discriminant analysis (LDA) [60], logistic regression (LR) [153] and Bayesian linear regression (BLR) [151].

Investigation of spatial filtering in CCCV systems has been explored previously in the literature [92, 134, 152]. What is unclear from these studies is how to determine the optimal number of spatial filters i.e., this detail has been omitted in previous studies and yet this is an important consideration so is included in this investigation. The primary objective of this chapter is to explore the performance of pipelines that combine different spatial filtering approaches and classifiers where their respective hyperparameters are explored through (cross-validated) random search [157]. A pipeline in this chapter comprises spatial filtering, feature dimensionality reduction and classification steps. Three spatial filtering approaches are explored in this work, namely xDAWN [130], multiple time window LDA beamformers (MTWLB) which is an extension of LDA beamformer [135] and common spatial pattern (CSP) [132]. Principal component analysis (PCA) is utilized for feature dimensionality reduction. Three linear classification methods were explored, namely LDA, BLR and LR respectively. There are nine pipelines in total including spatial filtering and classification combinations. Three pipelines that only apply the three classification methods without applying any spatial filtering are used as baseline performance comparison. This chapter should provide neurotechnologists, who seek to apply a RSVP paradigm to EEG, with a comprehensive assessment of the comparative performance of both

commonly used spatial filtering pipelines and a new method that are all assessed using a new publicly available benchmark dataset [102, 104].

Time-course source signal reconstruction can be of benefit for ERP studies. Traditional ERP study is carried out by analyzing the EEG signal at single electrode sites, which has low signal quality and may be affected by some types of noise e.g., electrode noise. RSVP-EEG data is more noisy compared to EEG measured under other experimental paradigms e.g., one type of ERP called steady state visual evoked potential (SSVEP) [123] is also activated in the RSVP paradigm, which can be considered as a type of artifact. It is advised to use the reconstructed time-course source signal instead of the signal measured at the single electrode site for ERP analysis in a RSVP paradigm. Previous work [135] has demonstrated the efficacy of using the time-course reconstructed signal compared to the signal measured at the single electrode sites in ERP study.

Recently, several approaches have been introduced for the reconstruction and localization of neural sources from EEG. However, these studies focused on the performance for source localization rather than time-course source signal reconstruction. In this chapter, we explore the performance of using xDAWN and MTWLB for the time-course source signal reconstruction for a CCCV system because these two approaches are well performed for the image classification task compared to CSP.

This chapter is organized as follows. Firstly, we describe the methodology which includes pipeline construction and the performance evaluation metrics used in this chapter. Secondly, we clarify the experimental RSVP-EEG dataset used in this chapter. Finally, results and discussion are presented in the last two sections.

4.2 Methodology

4.2.1 Pipeline Description

This chapter explores nine pipelines comprising spatial filtering, feature dimensionality reduction and classification respectively along with three pipelines containing feature dimensionality reduction and classification as comparisons. Figure 4.1 illustrates the two pipeline architectures under consideration in this study. With spatial filtering applied, a n channel EEG epoch is transformed to m source components ($m \leq n$). Before the feature generation step, we applied PCA for each individual channel (pipelines without spatial filtering) and individual component

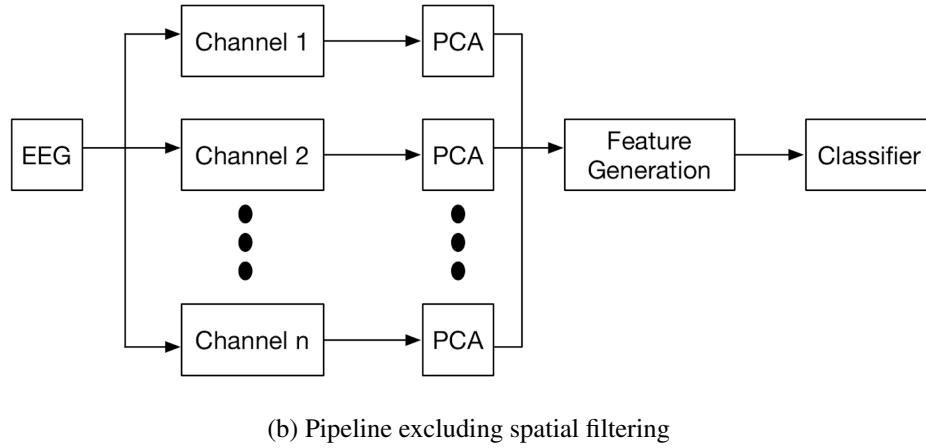
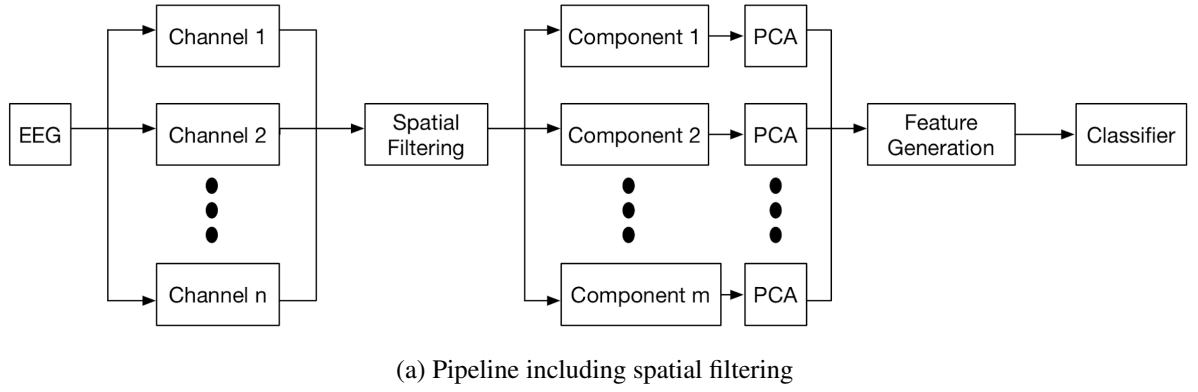


Figure 4.1: Two pipeline architectures for RSVP-based EEG discussed in this chapter.

(pipelines with spatial filtering) on the temporal axis for dimensionality reduction following the work [143]. The reason why we apply PCA individually is because EEG power in each channel and each component is not consistent and this step ensures that discriminant information is not lost. We left out the PCA components which contain less than 1% ratio of the variance. The feature generation step concatenated m components or n channel EEG to a feature vector before inputting it to the classifier. Details of the feature generation can be referred to the section 4.2.3.

4.2.2 Supervised Spatial Filtering

We have briefly introduced supervised spatial filtering approaches in Chapter 3. Here we give a recap and provide more configuration details such as selection of spatial filters which can affect the performance of spatial filtering pipeline for a CCCV system. Before introducing the supervised spatial filtering approaches, we clarify the notations to be used in this chapter. N_c is the number of channels, N_t is the number of time samples in an epoch, N_f is the number of selected spatial filters and n is the number of epochs.

Spatial filtering creates a weighted combination of each EEG channel input in order to enhance a particular subset of information which is contained in the original EEG epoch. Spatial filtering reduces the number of features because the number of spatial filter output N_f is smaller than the number of channels N_c . The problem of spatial filtering is to find projection vectors (spatial weights for each channel) $\mathbf{w} \in \mathbb{R}^{N_c \times N_f}$ to project $\mathbf{X} \in \mathbb{R}^{N_c \times N_t}$ to a subspace, where \mathbf{w} is calculated by different optimization criterion.

$$\mathbf{X}_{\text{sub}} = \mathbf{w}^\top \mathbf{X} \quad (4.1)$$

Several approaches have been presented in the literature for generating spatial filters \mathbf{w} in equation (4.1) in the area of BCIs research. Independent component analysis (ICA) is a blind source separation technique which can be used to find a linear representation of non-Gaussian data so that the components are statistically independent, or as independent as possible [137]. Such a representation is capable of capturing the inherent structure of data in many applications, and hence has applications to feature extraction [61, 136] and removing artifacts from EEG signals [139]. Specifically, ICA finds a component “unmixing” matrix (\mathbf{w}) that, when multiplied by the original data (\mathbf{X}), yields the matrix (\mathbf{X}_{sub}) of independent component (IC) time courses [177]. Generally, the main purpose of ICA is blind source separation instead of being specifically intended to discriminate EEG in two experimental tasks. PCA is another statistical technique that uses eigenvalue decomposition to convert a set of correlated variables into a set of linearly uncorrelated variables. PCA has been applied to EEG signals for dimensionality reduction [141] and generating spatial filters [142]. Similar to ICA, PCA operates without knowledge of stimulus types hence it is an unsupervised approach. In this work, we consider supervised spatial filtering methods that aim to enhance the difference between target and standard image stimuli. Three spatial filtering methods are considered in this chapter, namely MTWLB, xDAWN and CSP. MTWLB is an extension of the LDA beamformer method which aims to maximize the signal-to-noise ratio (SNR) in each individual time window. xDAWN and CSP are based on Rayleigh quotients where xDAWN maximizes the signal-to-signal-plus-noise ratio (SSNR) whereas CSP maximizes the difference of the variance between two classes. In the following paragraphs, we describe the operation of these three spatial filters generation techniques, i.e., the LDA beamformer with our window extensions, xDAWN and CSP.

LDA Beamformer

The LDA beamformer has been successfully applied for recovering N200 and P300 sources in an auditory experiment [135]. Considering a target epoch $\mathbf{X}_i \in \mathbb{R}^{N_c \times N_t}$ and a standard epoch $\mathbf{K}_i \in \mathbb{R}^{N_c \times N_t}$, let column vectors $\mathbf{p}_1 \in \mathbb{R}^{N_c \times 1}$ and $\mathbf{p}_2 \in \mathbb{R}^{N_c \times 1}$ be the spatial pattern of a specific component in two different experimental conditions. We denote the difference pattern as $\mathbf{p} := \mathbf{p}_1 - \mathbf{p}_2$ and the covariance matrix $\Sigma \in \mathbb{R}^{N_c \times N_c}$. The optimization problem of the LDA beamformer can be referred to equation (3.9) and equation (3.10) in previous Chapter 3.

The spatial pattern for the LDA beamformer was directly estimated from the difference between ERP peaks in an oddball experiment [135]. As seen in Fig. 4.2, the bold red line is

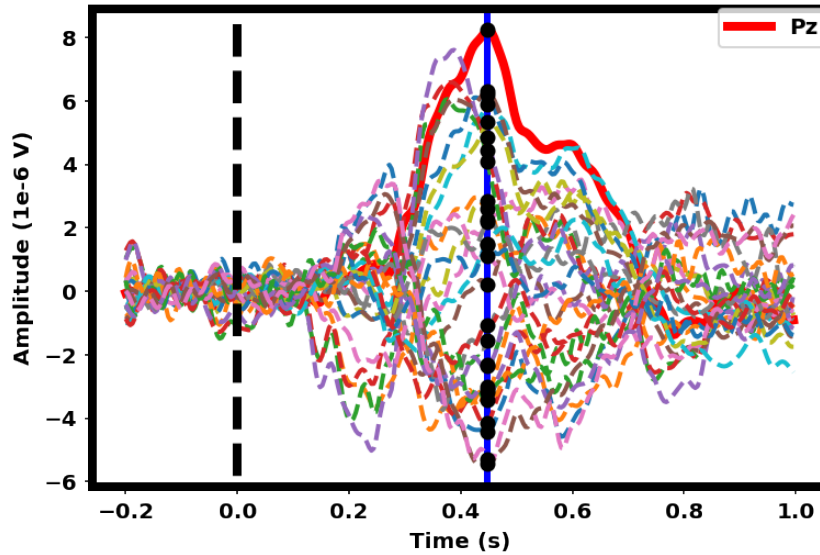


Figure 4.2: Spatial pattern estimation for LDA beamformer using whole EEG epoch via training data using CAR: *Participant 2*.

the ERP difference at the Pz channel and the blue line represents the peak value timestamp of difference ERPs at the Pz channel. The different ERP values across all channels at that timestamp can then be used to estimate a spatial pattern.

Due to the substantial overlap between adjacent target epochs and non-target epochs, along with inherent variability in ERP latencies and topographies between participants, we extended the LDA beamformer to MTWLB. The key idea of MTWLB is to train multiple LDA beamformer models over non-overlapping successive time windows i.e., to train a single LDA beamformer for each time window that is adaptive to the local spatio-temporal features characterizing target-related ERPs activity at that time point. *In order to be easier to discuss and compare MTWLB with xDAWN and CSP, we still use N_f here to represent the number of spatial filter out-*

put. However, the physical meaning of N_f here is **the number of divided time windows**, which is different from traditional spatial filtering approaches. More details can refer to Algorithm 1.

Spatial Pattern Estimation for MTWLB In contrast with the LDA beamformer method, we estimated the spatial pattern and calculate the equation (3.10) separately for each time window rather than whole time series. Therefore, there were N_f estimated spatial patterns each derived from the 32 channel data corresponding to that time point. The reason why we do this is the substantial overlap between adjacent target epochs and non-target epochs, along with inherent variability in ERP latencies and topographies between participants in RSVP-EEG.

Covariance Matrix Estimation for MTWLB MTWLB uses whole EEG epochs to estimate the covariance matrix as stationarity reasons have been stated in [135]. Given target epochs $\mathbf{X} \in \mathbb{R}^{n \times N_c \times N_t}$ and standard epochs $\mathbf{K} \in \mathbb{R}^{m \times N_c \times N_t}$, the covariance matrix can be calculated as: $\Sigma = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_i \mathbf{X}_i^\top + \frac{1}{m} \sum_{i=1}^m \mathbf{K}_i \mathbf{K}_i^\top$, where n and m are numbers for target and standard stimuli respectively. After using artifact rejection or other EEG pre-processing methods, the covariance matrix is singular. We used shrinkage algorithms to regularize the covariance matrix in order to make it invertible [178]. The MTWLB implementation is included in Algorithm 1.

xDAWN

The xDAWN algorithm has been applied in BCIs for ERP detection in the P300 speller paradigm [130, 179] and the RSVP paradigm [134]. The goal of xDAWN is to apply spatial filters \mathbf{w} to enhance the SSNR of the ERP responses corresponding to the target stimuli. The optimization problem for xDAWN has been introduced as equation (3.6) and equation (3.8) in Chapter 3.

Common Spatial Pattern

CSP is one of the most popular spatial filtering approaches for motor imagery based BCIs, where the task involves two different states of brain activity (e.g., imagery of the movement of the left or right hand) [132, 133]. CSP aims to maximize the variance of one class and minimize the variance of another class. The optimization problem for CSP can also be estimated and interpreted as Rayleigh quotient [180].

First, let $\mathbf{X}_1(i)$ and $\mathbf{X}_0(i)$ be the i th event locked ERP epoch $\in \mathbb{R}^{N_c \times N_t}$ and two covariance matrices Σ_1 and Σ_0 are calculated as follows (subscript “0” for standard condition and “1” for

4.2. Methodology

Algorithm 1 Implementation of MTWLB

Input:

- $\mathbf{X} \in \mathbb{R}^{n \times N_c \times N_t}$ is the EEG signal corresponding to the target stimulus, where n is the number of target trials, N_c is the number of channels, and N_t is the number of time points.
- $\mathbf{K} \in \mathbb{R}^{m \times N_c \times N_t}$ is the EEG signal corresponding to the standard stimulus, m is number of standard trials, N_c is number of channels, N_t is number of time points.

Output: Spatial filters \mathbf{W}

- 1: Set N_f time windows for MTWLB. $\triangleright N_f$ is the number of divided time windows.
 - 2: $M = \frac{N_t}{N_f}$ \triangleright This is the number of temporal points in each time window.
 - 3: $\Sigma = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_i \mathbf{X}_i^\top + \frac{1}{m} \sum_{i=1}^m \mathbf{K}_i \mathbf{K}_i^\top$
 - 4: $\mathbf{S} = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_i - \frac{1}{m} \sum_{i=1}^m \mathbf{K}_i$ \triangleright This is ERP difference between target epoch and standard standard.
 - 5: $\mathbf{W} \leftarrow$ empty list
 - 6: **for** $i = 1 : N_f$ **do**
 - 7: $\mathbf{J} \leftarrow$ empty list
 - 8: $\mathbf{w} \leftarrow$ empty list
 - 9: **for** \mathbf{p} in $\mathbf{S}[:, (i-1) \times M : i \times M]$ **do**
 - 10: $\mathbf{w} \leftarrow \Sigma^{-1} \mathbf{p} (\mathbf{p}^\top \Sigma^{-1} \mathbf{p})^{-1}$
 - 11: $\mathbf{J} \leftarrow \mathbf{w}^\top \Sigma \mathbf{w}$
 - 12: **end for**
 - 13: $\mathbf{W} \leftarrow \mathbf{W}_{\arg \min \mathbf{J}}$ \triangleright Here we get N_f spatial filters
 - 14: **end for**
-

target condition)

$$\Sigma_c = \frac{1}{n} \sum_{i=1}^n \frac{\mathbf{X}_c(i) \mathbf{X}_c^\top(i)}{\text{Trace}(\mathbf{X}_c(i) \mathbf{X}_c^\top(i))} \quad (4.2)$$

The solution for CSP can be determined through Raleigh quotients by solving a generalized eigenvalue problem

$$\arg \max_{\mathbf{w}} \frac{\mathbf{w}^\top \Sigma_1 \mathbf{w}}{\mathbf{w}^\top \Sigma_0 \mathbf{w}} \quad (4.3)$$

Similar to the previous two approaches, CSP is able to generate a set of spatial filters. However, spatial filters in CSP appear pair-by-pair because CSP maximizes variance in one class and minimizes variance in the other class. From Cecotti's work, four spatial filters were chosen as $[\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_{N_c-1}, \mathbf{w}_{N_c}]$ (N_c is the number of electrodes) [134]. We followed previous Cecotti's work [134], which uses each CSP spatial filter to filter the raw EEG epoch i.e., as seen in equation (4.1). This work chose a pair of spatial filters via cross-validation.

At this point, we have highlighted how the three methods under consideration here can generate spatial filters \mathbf{w} . All three spatial filters generation strategies serve the same objective of reducing computation complexity but the optimization aims are different. MTWLB generates

spatial filters based on maximizing the SNR in individual time windows. xDAWN, in contrast generates spatial filters based on maximizing the SSNR for the whole EEG epoch. Finally the method of CSP generates spatial filters through maximizing the variance difference between two classes.

4.2.3 Feature Generation

The spatial filter $\mathbf{w} \in \mathbb{R}^{N_c \times N_f}$ generated serves to transform the original EEG epoch $\mathbf{X} \in \mathbb{R}^{N_c \times N_t}$ to the feature space as mentioned before in equation (4.1), where $\mathbf{X}_{sub} \in \mathbb{R}^{N_f \times N_t}$. The projected subspace \mathbf{X}_{sub} can be represented as spatial-filtered EEG signals involving different discriminant information corresponding to the criteria used in their filter generation. In order to reduce the computational complexity, PCA is applied to each row in \mathbf{X}_{sub} for feature reduction (see section 4.2.1). In this work, principal components, whose explained variance ratios are greater than 1%, were *selected and concatenated as the feature vector* Ψ which would be used as inputs to the classification step.

For details of calculating the produced feature vector size, let us denote the number of features in each row of \mathbf{X}_{sub} after PCA processing as m_i . The number of final features Ψ can be calculated as $m_\Psi = \sum_{i=1}^{N_f} m_i$, where $\Psi \in \mathbb{R}^{1 \times m_\Psi}$.

4.2.4 Linear Classifiers

Linear classifiers are widely used for CCCV systems due to their good performance, often simple implementation and low computational complexity [59–61, 146]. In this chapter, we focus on three widely used linear classifiers in CCCV research, namely linear discriminant analysis (LDA), logistic regression (LR) and Bayesian linear regression (BLR), where these three methods have already been introduced in Chapter 3. So we do not go through the details of these classifiers in this chapter.

4.2.5 Evaluation

The evaluation described in this work seeks to assess relative performance when combining three spatial filtering approaches with three linear classification methods, thus there are nine pipelines (spatial filtering \times feature generation \times classification) in total that are discussed in this chapter. For comparison, the original EEG epochs without spatial filtering and only using

PCA, were used as inputs to three linear classifiers. Performance of the different pipelines were evaluated through the area under the curve (AUC) of the ROC (receiver operating characteristic) curve that is based on true positive rate (TPR) and false positive rate (FPR).

It should be noted that the pipelines described in this chapter contain the number of hyperparameters. Three spatial filtering approaches contain a number of spatial filters N_f as the hyperparameter. BLR contains data distribution variance (β) and parameter distribution variance (α) while LR has the regularization term (λ) as a hyperparameter. Only LDA does not require a hyperparameter. Table 4.1 summarizes the hyperparameters used in each pipeline. We

Pipeline	Hyperparameter
MTWLB _{LDA}	N_f
MTWLB _{BLR}	N_f, β, α
MTWLB _{LR}	N_f, λ
xDAWN _{LDA}	N_f
xDAWN _{BLR}	N_f, β, α
xDAWN _{LR}	N_f, λ
CSP _{LDA}	N_f
CSP _{BLR}	N_f, β, α
CSP _{LR}	N_f, λ
ALL _{LDA}	None
ALL _{BLR}	β, α
ALL _{LR}	λ

* Pipeline name comprises the used spatial filtering approach and classifier e.g., MTWLB_{LDA} refers to the use of the MTWLB spatial filter and the LDA classifier. "ALL" refers to the raw feature used without any spatial filtering processing.

Table 4.1: Hyperparameter summary for each pipeline discussed in this chapter. For each hyperparameter, $N_f \in [1, 16]$ is searched from 1 to 16 spatial filters, $\alpha \in [1e - 6, 1e + 2]$ is sampled from uniform distribution, $\beta \in [1e - 6, 1e + 2]$ is sampled from uniform distribution, $\lambda \in [1e - 2, 1e + 2]$ is sampled from uniform distribution.

applied a random search [181] for 100 hyperparameter combinations for each pipeline and selected these by evaluating on a validation set using 10-fold cross validation. The optimal model was then applied to the testing data to calculate the AUC score.

This chapter used the NAILS EEG dataset, which has been introduced in Chapter 2, for the investigation of the classification performance. The dataset was split into a training/testing set of 66%/33% respectively, by selecting 3 blocks (the 3rd, 6th, 9th) from each search task to act as a withheld test set in the evaluation.

4.3 Results

4.3.1 Impact of Number of Spatial Filters

The P300 is not the only ERP component that is commonly encountered when using a RSVP target search paradigm. Early ERPs (notably the N200) are often present alongside the P300 [182] and can be useful in providing discriminant information for classification. Figure 4.3 shows the discriminant ERPs for 9 participants in two different time regions and it can be seen

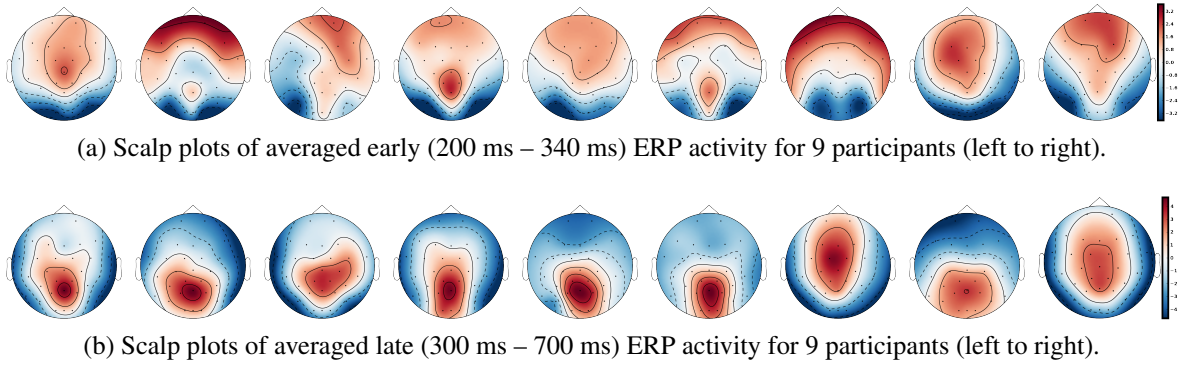


Figure 4.3: ERPs for each participant corresponding to time regions of (a) 200 ms – 340 ms and (b) 370 ms – 700 ms. These were selected for presentation to emphasize the presence of discriminant ERP-related activity in these time regions across participants, namely time regions coinciding with P200, N200 and P300 ERP activity.

that both early time regions and later time regions display discriminant ERP-related activity across participants. It can be also noted that the specific latencies and topographies vary across participants, hence N_f (MTWLB) may vary across participants for capturing target-related ERP phenomena in the CCCV system.

From previous work [134], N_f has been set to 4 for both xDAWN and CSP methods. It is difficult to determine the optimal N_f , thus we left it as a searchable hyperparameter in our pipeline. Even though selection of the optimal number of spatial filters has been recognized as an appropriate strategy in the area of the motor imagery BCIs [183], we require further hyperparameters in this work. In this case, we searched for the optimal number of spatial filters along with other parameters, together in each model, which has been specified in the Evaluation section. It is worth reiterating again that this has not been explicitly reported upon previously in the area of CCCV. Figure 4.4 shows an example of 10 spatial patterns and filters estimated with three spatial filtering approaches. It can be seen that spatial patterns estimated with the same approach are different from each other, which indicates target-related ERPs span broadly

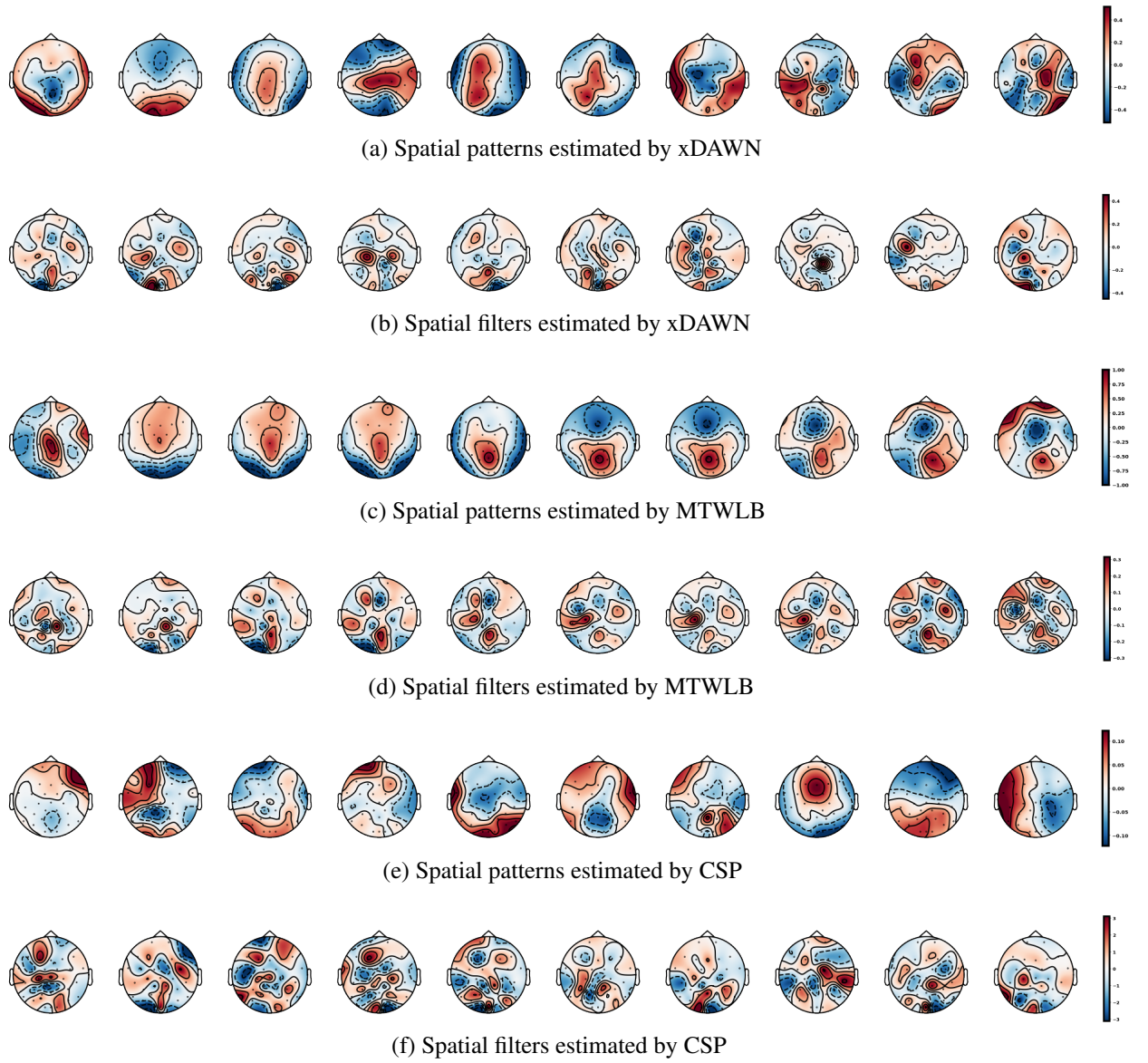


Figure 4.4: Example of estimated spatial patterns/filters for three spatial filtering approaches from 0 ms to 1000 ms for *Participant 1*.

4.3. Results

over both time and space in a RSVP paradigm. Hence, a search for an optimal value of N_f is required for the best performance.

4.3.2 Performance Evaluation

This work evaluated 9 different pipelines, composed of three classifiers exploring three spatial filtering methods. We used whole original EEG epochs (i.e., not spatially filtered) for training three classifiers as measuring metrics for comparison. The AUC scores for each pipeline across nine participants are presented in Table 4.2.

Pipeline	Participant									Mean
	1	2	3	4	5	6	7	8	9	
MTWLB _{LDA}	88.0	93.8	92.5	96.8	91.4	93.7	93.5	90.1	89.9	92.2
MTWLB _{BLR}	88.0	93.8	92.5	96.8	92.2	93.7	93.2	90.8	93.3	92.3
MTWLB _{LR}	88.5	93.0	89.8	97.4	91.7	94.1	93.3	91.3	90.9	92.2
Mean	88.2	93.5	91.6	97.0	91.8	93.8	93.3	90.7	91.4	92.2
xDAWN _{LDA}	88.0	93.4	92.7	97.3	91.6	94.3	92.9	90.6	90.1	92.3
xDAWN _{BLR}	88.5	93.4	92.8	96.6	91.6	94.3	92.8	90.6	90.1	92.3
xDAWN _{LR}	87.4	94.1	92.9	97.2	91.9	95.3	93.8	91.3	90.7	92.7
Mean	88.0	93.6	92.8	97.0	91.7	94.6	93.2	90.8	90.3	92.4
CSP _{LDA}	84.3	92.8	88.8	96.0	87.7	90.9	89.8	76.9	89.7	88.5
CSP _{BLR}	84.9	92.7	88.3	96.0	87.9	90.9	89.6	77.5	89.7	88.6
CSP _{LR}	83.8	92.7	85.7	93.1	86.6	90.3	89.5	76.8	89.0	87.5
Mean	84.3	92.7	87.6	95.0	87.4	90.7	89.6	77.1	89.5	88.2
ALL _{LDA}	86.9	93.4	90.8	96.7	91.9	94.0	95.0	89.4	90.3	92.0
ALL _{BLR}	88.0	93.1	91.4	96.0	91.6	93.4	93.8	89.4	90.7	91.9
ALL _{LR}	84.4	92.4	86.0	92.3	87.6	92.1	91.5	87.3	82.8	88.5
Mean	86.4	93.0	89.4	95.0	90.4	93.2	93.4	88.7	87.9	90.8

* Pipeline name comprises the used spatial filtering approach and classifier e.g., MTWLB_{LDA} refers to the use of the MTWLB spatial filter and the LDA classifier. "ALL" refers to the raw feature used without any spatial filtering processing.

Table 4.2: AUC score (%) for different pipelines across nine participants in testing session.

Performance of Spatial Filtering

Examining the performance of the three spatial filtering methods, all three classifiers with CSP pre-processing generate lower AUC score compared to those which do not use spatial filtering. This indicates that CSP (without modification) is a wholly unsuitable spatial filtering approach for a CCCV system. Unlike CSP, all three classifiers with MTWLB and xDAWN spatial filtering pre-processing perform better than those without spatial filtering, which show the efficacy of MTWLB and xDAWN pre-processing. This result demonstrates that it is critical to carefully

select the precise spatial filtering method in a CCCV system and an “improper” spatial filtering method may have deleterious effects and degenerate performance to the level of not using any spatial filtering (or worse).

CSP aims to maximize the EEG power variance difference between two classes. However, the single-trial ERP variance difference is very small in a RSVP paradigm between two classes (i.e., as mentioned before, this is caused by the large variance of SSVEP, which is elicited in both target and standard conditions) and the challenge for single-trial ERPs detection is its low SNR. Here we define the “proper” spatial filtering approach for RSVP-EEG as those methods which improve the SNR for the EEG signals. Both MTWLB and xDAWN optimize for a maximized SNR and as a result both perform better than the inappropriately applied CSP method. A proper spatial filtering method can not only improve the quality of the EEG data but also reduce the computational complexity since spatial filtering can reduce the EEG dimensionality.

Performance of Classifier

As mentioned before, CSP is not able to extract particularly discriminant features for ERPs generated via a RSVP paradigm. Therefore, the features generated through CSP have negative effects on all three classifiers compared to the use of EEG data without spatial filtering. From results generated by LR across MTWLB and xDAWN and without spatial filtering, it can be seen that the performance of LR is improved significantly when using non-CSP spatial filtering methods i.e., 92.2% for MTWLB (one-way ANOVA compared with no spatial filtering: $F(1, 16) = 6.12, p = 0.02$) and 92.7% for xDAWN (one-way ANOVA compared with no spatial filtering: $F(1, 16) = 7.44, p = 0.01$) versus 87.5% and 88.5% without spatial filtering. This indicates that the quality of features has a large impact on the performance of LR. With respect to the other two classifiers, spatial filtering improves the performance of LDA and BLR slightly. This indicates that LDA and BLR are more robust to the quality of features compared to LR. However, LR shows good performance if good features can be extracted by pre-processing (i.e., highest AUC score for LR with xDAWN).

4.3.3 Source Reconstruction

It has been shown that xDAWN and MTWLB are more effective in extracting discriminant information of ERPs for a CCCV system compared to CSP. And the classification performance between xDAWN and MTWLB are close to each other. We are going to compare the perfor-

4.3. Results

mance of reconstructing time-course source signal for xDAWN and MTWLB respectively.

To carry out more robust analysis, the NIFPA dataset was used for this aspect because two ERPs components have been elicited, which are the N170 and P300. We are going to explore the performance of using xDAWN and MTWLB for reconstructing the N170 and the P300 source signals.

Figure 4.5 demonstrates the spatial patterns topographical plots for the N170 and the P300

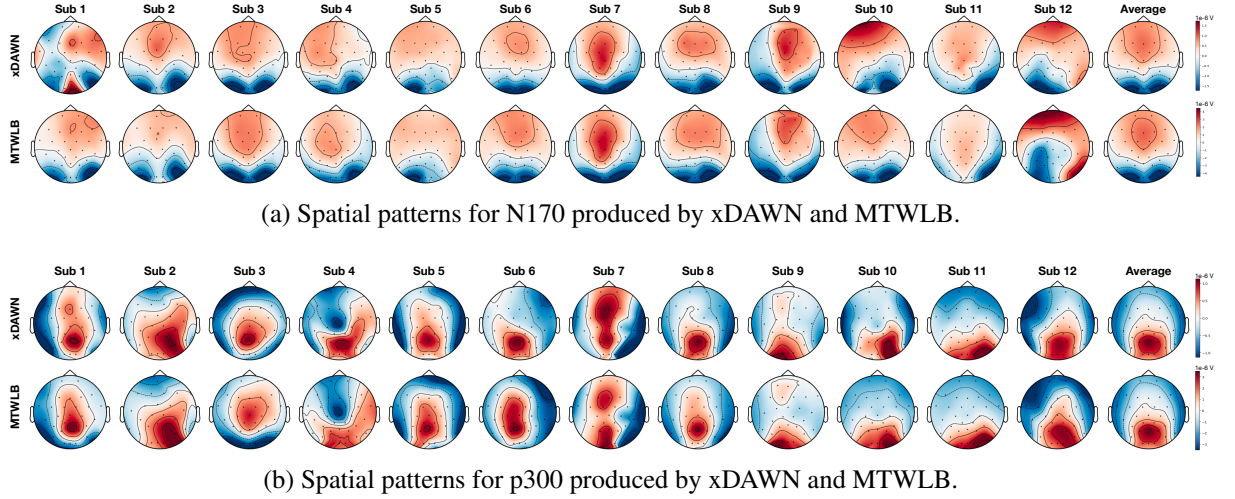
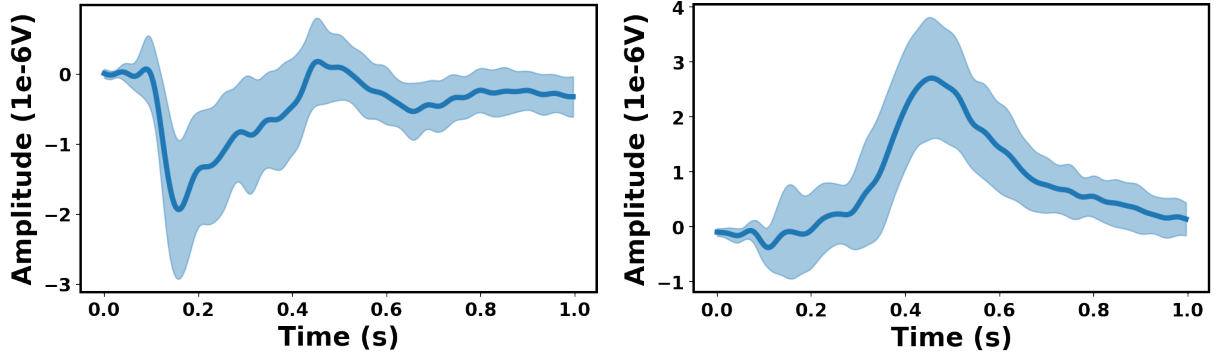
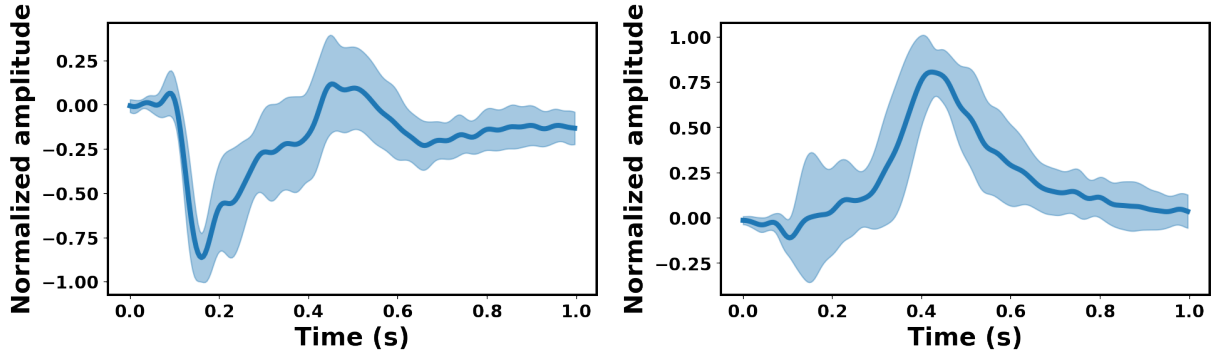


Figure 4.5: Spatial patterns topographical plots for the N170 and P300 produced by xDAWN and MTWLB across 12 participants (1 – 12 from left to right and last column was the averaged topography of 12 participants). Spatial filtering was applied in time window 100 ms – 200 ms for the N170 while time window 400 ms – 600 ms was used for the P300. Red color indicates the positive EEG amplitude while blue color indicates the negative EEG amplitude.

produced by xDAWN and MTWLB. Red color in the topography indicates the positive EEG amplitude while blue color indicates the negative EEG amplitude. It can be seen that the averaged spatial patterns for the N170 and P300 across 12 participants produced by xDAWN and MTWLB are close to each other, which indicates that these two spatial filtering approaches extract similar spatial information from the raw EEG epochs. Figure 4.6 demonstrates the reconstructed N170 and P300 signals (calculated via equation (4.1)) across 12 participants by using these two spatial filtering approaches. The solid lines in the graph are the mean values of 12 reconstructed source signals for 12 participants while the shadow area represents the standard deviation of the reconstructed source signals across the participants. Based on the spatial properties demonstrated in Fig. 4.5 and the temporal properties demonstrated in Fig. 4.6, we show that two ERPs (the N170 and P300) used here can be successfully reconstructed by using xDAWN and MTWLB.



(a) N170 (left) and P300 (right) reconstructed by xDAWN.



(b) N170 (left) and P300 (right) reconstructed by MTWLB.

Figure 4.6: Time-course source N170 and P300 reconstructed by xDAWN (top) and MTWLB (bottom) across 12 participants. Solid lines demonstrate the averaged value of 12 reconstructed source signals for 12 participants while shadow areas are standard deviations across participants. *MTWLB normalizes reconstructed source signals inside algorithm while xDAWN does not.*

In terms of validating the quality of the reconstructed source signals, we used SNR and classification accuracy (i.e., AUC score used here) between target condition and standard condition.

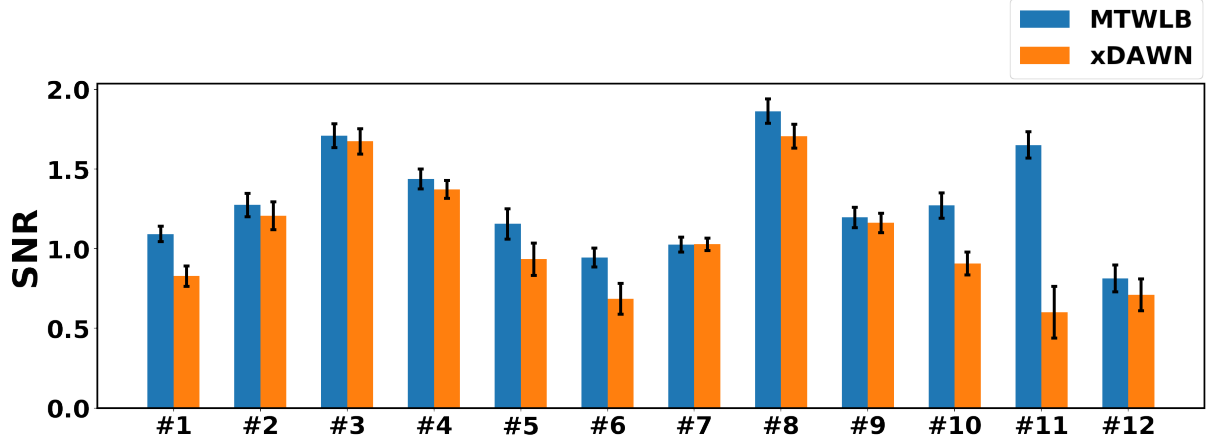
Suppose we have the reconstructed source signal $\mathbf{X} \in \mathbb{R}^{n \times N_t}$ for target stimuli and the reconstructed source signal $\mathbf{K} \in \mathbb{R}^{m \times N_t}$ for standard stimuli, where n is the number of target trials, m is the number of standard trials and N_t is the time length. The SNR is calculated in two parts: (1) Peak amplitude of the reconstructed source signal difference between target stimuli and standard stimuli in the ERP time region (100 ms – 200 ms for N170 and 400 ms – 600 ms for P300); and (2) Background noise estimation. So SNR for ERPs can be defined as $\text{SNR} := \frac{|\text{peak}|}{\text{noise}}$ [184], where the background noise can be characterized by the standard deviation of standard EEG epochs [185, 186]. Equation (4.4) summarizes these two aspects for estimating SNR. The numerator in the equation (4.4) calculates the aspect (1) in the selected time window and the denominator estimates the EEG background noise via the standard deviation of

4.3. Results

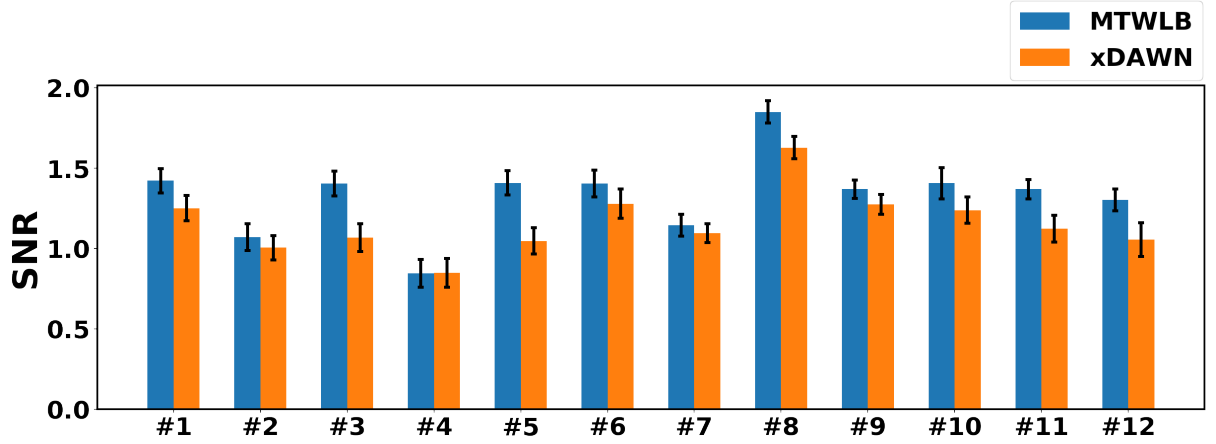
reconstructed signal corresponding to the standard stimuli.

$$\text{SNR} = \frac{\max \left| \frac{1}{n} \sum_{i=1}^n \mathbf{X}_{i,\text{win}} - \frac{1}{m} \sum_{i=1}^m \mathbf{K}_{i,\text{win}} \right|}{\frac{1}{m} \sum_{i=1}^m \sqrt{(\mathbf{K}_i - \frac{1}{m} \sum_{i=1}^m \mathbf{K}_i)^2}} \quad (4.4)$$

Figure 4.7 summarizes the SNR of the reconstructed N170 and P300 via xDAWN and MTWLB



(a) SNR for N170 reconstructed by xDAWN and MTWLB.



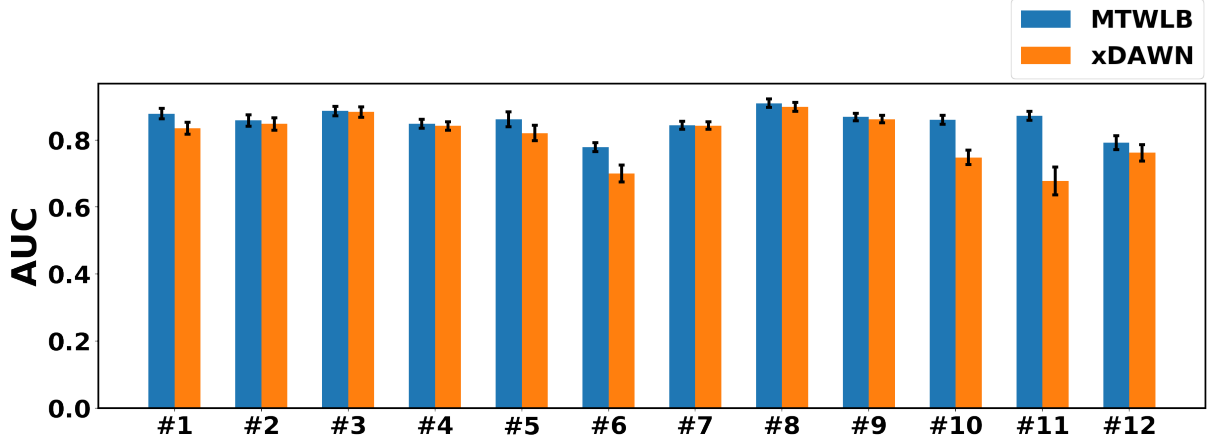
(b) SNR for P300 reconstructed by xDAWN and MTWLB.

Figure 4.7: SNR for reconstructed N170 (top) and P300 (bottom) across 12 participants. SNR was calculated as shown in equation (4.4). The averaged values across participants for N170 are: $\text{SNR}_{\text{MTWLB}} = 1.29$ and $\text{SNR}_{\text{xDAWN}} = 1.08$. The averaged values across participants for P300 are: $\text{SNR}_{\text{MTWLB}} = 1.33$ and $\text{SNR}_{\text{xDAWN}} = 1.16$. Details of the figure summary can be referred to Table B.1 in Appendix B.

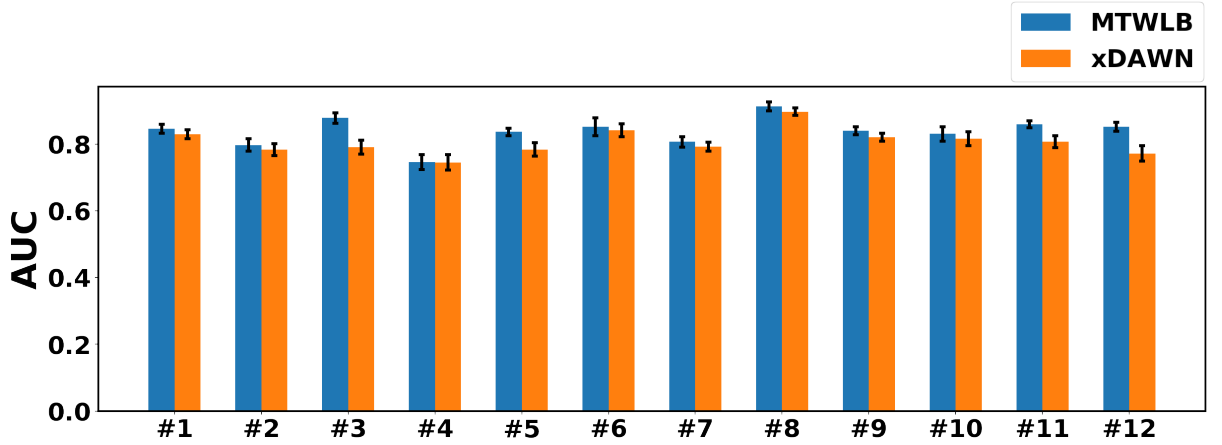
respectively for all participants. It can be seen that the SNR of both N170 and P300 reconstructed by MTWLB is higher than those reconstructed by MTWLB for almost every case, which indicates that the time-course source signal reconstructed by MTWLB has better signal quality.

4.3. Results

We also explored the classification performance between these two types of source signals reconstructed by xDAWN and MTWLB to see if the signal quality difference has any effect on the classification performance. Figure 4.8 demonstrates the difference of the classification



(a) AUC score of using the reconstructed N170 for the classification between target images and standard images.



(b) AUC score of using the reconstructed P300 for the classification between target images and standard images.

Figure 4.8: AUC score of using the source signal (N170 top and P300 bottom) reconstructed by xDAWN and MTWLB across 12 participants. The BLR was used as the classifier in this case. The averaged values across participants for N170 are: $AUC_{MTWLB} = 0.855$ and $AUC_{xDAWN} = 0.810$. The averaged values across participants for P300 are: $AUC_{MTWLB} = 0.838$ and $AUC_{xDAWN} = 0.806$. Details of the figure summary can be referred to Table B.2 in Appendix B.

performance between these two types of reconstructed source signals. It can be seen that the classification performance of using the source signal reconstructed by MTWLB is better than that produced by xDAWN, which indicates that the enhanced SNR has positive effect on the classification performance.

We have demonstrated that MTWLB is able to provide higher SNR for reconstructed source signal and better classification performance compared to xDAWN. These results demonstrate

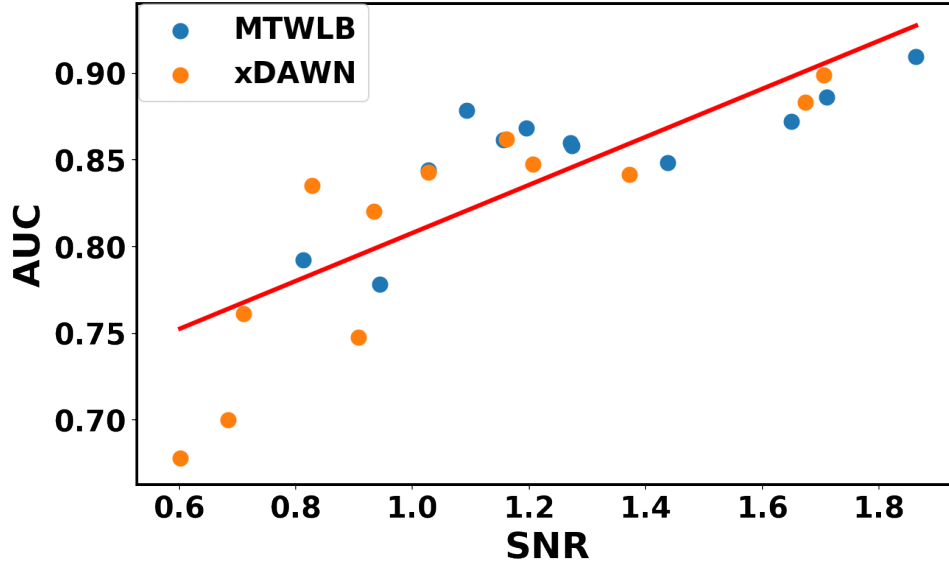
that MTWLB is not only suitable for the cortically coupled image classification (comparable classification performance in Table 4.2 to xDAWN) but also has good performance on reconstructing time-course source signal for a CCCV system. MTWLB is able to interpret the neurophysiology of ERPs over time from both spatial and temporal perspectives (as seen in Fig. 4.5 and Fig. 4.6), which provides an insight of ERP activity for a CCCV system over time.

To get a deeper insight on the relationship between signal quality (SNR) and classification performance (AUC), we conducted a correlation analysis between SNR and AUC of two reconstructed source components the N170 and the P300 using MTWLB and xDAWN (see Fig. 4.9). It can be seen that correlation between AUC and SNR of the P300 is linear while the N170 is not. So we use nonlinear and linear correlation statistics for the N170 and the P300 respectively i.e., Spearman correlation statistics for the N170 and Pearson correlation statistics for the P300. It can be seen that there is a strong positive correlation between AUC and SNR (N170: $p = 9.094 \times 10^{-7}$, P300: $p = 8.810 \times 10^{-14}$), which indicates that SNR is an important factor for considering the performance of ERP source reconstruction.

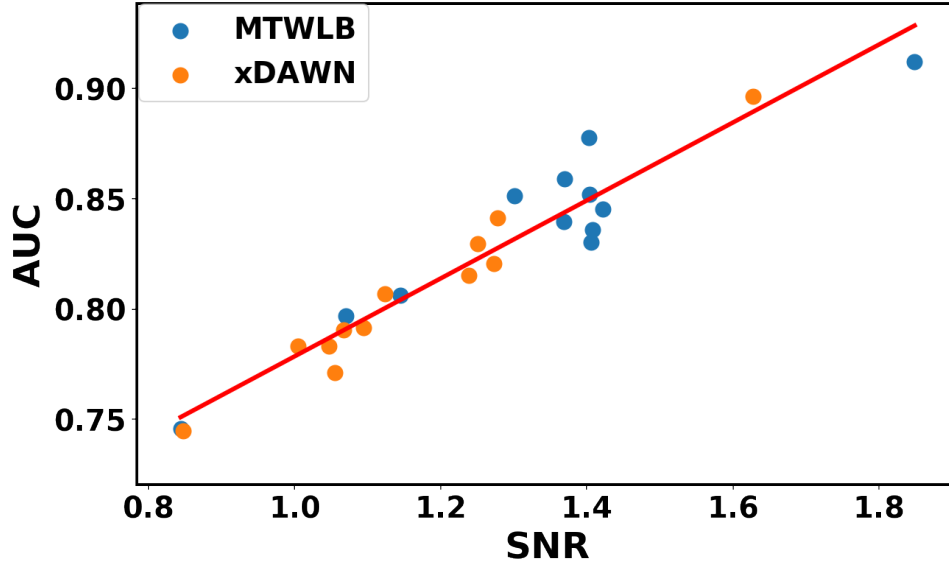
4.4 Discussion

In this chapter, we addressed three main issues: (1) The impact of choice of spatial filtering method on the performance of a CCCV system for a single-trial classification task; (2) The sensitivity of different classifiers' performance to the feature types produced in a typical CCCV system pipeline; and (3) The performance of reconstructing time-course ERP source signals that are elicited in a RSVP paradigm.

Regarding the first issue, we have shown that the performance of our novel MTWLB method and the popular xDAWN method both improve classification performance. Moreover, the performance generated by pipelines involving CSP is worse than those which do not use spatial filtering. This indicates that the choice of spatial filtering method is critical for single-trial detection of ERPs for a CCCV system. In the literature, we find some work that used CSP (and variants) for RSVP-EEG [120, 187]. These papers highlighted that CSP suffers from a number of issues and suggested extensions to the approach such as CSTP [120] which are a more suitable adaptation of the approach to CCCV systems. By comparing the optimization criterion of each spatial filtering approach, it should be noted that MTWLB and xDAWN both use ERPs for calculating the spatial filters (i.e., ERPs difference \mathbf{p} and the estimated ERPs responses $\mathbf{D}\hat{\mathbf{A}}$ are



(a) Correlation between AUC and SNR in terms of N170.



(b) Correlation between AUC and SNR in terms of P300.

Figure 4.9: Correlation analysis between SNR and AUC via using reconstructed source signals (the N170 and P300) by MTWLB and xDAWN across 12 participants. Pearson correlation statistics are used for the N170 and the P300 respectively. Pearson correlation statistics for the N170: $r(24) = 0.821, p = 9.094 \times 10^{-7}$; Pearson correlation statistics for the P300: $r(24) = 0.961, p = 8.810 \times 10^{-14}$.

used for calculating w in equation (3.10) and equation (3.8)).

The main difference between MTWLB and xDAWN is in the selection of the number of spatial filters. In MTWLB, the number of spatial filters is selected based on the divided individual time windows, which means each spatial filter maximizes SNR for ERPs in the selected time window. xDAWN uses a generalized eigenvalue decomposition for the whole EEG epoch and eigenvectors that correspond to high eigenvalues will be selected. Therefore, the number

of spatial filters N_f refers to the number of individual time windows in MTWLB while it refers to the number of eigenvectors corresponding to the highest eigenvalues in xDAWN. From the classification results, the proposed MTWLB approach gives similar performance compared to xDAWN. However, MTWLB can be well-suited for generating spatial filters for those tasks that elicit multiple ERPs with differing time and spatial characteristics. CSP also uses generalized eigenvalue decomposition for optimizing the spatial filter but it implements the optimization based on the covariance matrices corresponding to target trials and standard trials respectively, where the whole single-trial EEG epoch are used to calculate the covariance matrices for target trials and standard trials via using equation (4.3). Because RSVP-EEG has low SNR, this optimization formulation can be affected significantly by this low SNR in this case. The CSP approach originates from efforts to improve the motor imagery BCIs paradigm in which sensor motor rhythm (SMR), a periodic EEG, is elicited [188]. The optimization criterion for CSP is maximizing the ratio of variances (respective powers) between two classes which coincides with the properties of SMR. This approach typically includes application of a bandpass filter (8 Hz to 14 Hz) to attenuate power in non-relevant frequencies. In a RSVP paradigm, ERPs are elicited alongside the steady-state visual evoked potentials (SSVEP) and there is a very small difference in the variance between target and standard classes. The challenge for single-trial ERP detection is low SNR. MTWLB and xDAWN aim to improve the SNR and the SSNR respectively for the reconstructed signal which overcomes the low SNR problem. Therefore, MTWLB and xDAWN are very suitable for single-trial ERP detection in a RSVP paradigm. As a further aid to understanding the performance difference between MTWLB and xDAWN, we used one-way ANOVA on the mean value of three classification methods applied with MTWLB (92.2%) and xDAWN (92.4%) i.e., $[F(1, 16) = 0.004, p = 0.95]$. This indicates there is insignificant performance difference between these two spatial filtering methods. This suggests that the performance of these two methods are similar for this dataset at least. Despite this, MTWLB still has some advantages. Firstly, the proposed method MTWLB produces more intuitive outputs i.e., producing the spatial filter corresponding to the timeline. From the generated spatial filters w or the projected subspace Ψ , we can see both of them change over time. For example, spatial filters and spatial patterns change over time in MTWLB from left to right in Fig. 4.4. In this way, it provides a more physiologically correct view of the spatial patterns and the spatial filters changing with time, where a conventional spatial filtering approach is not able to represent. Secondly, we searched for the appropriate time window for MTWLB due to the inherent vari-

ability in ERPs latencies between participants in our case. However, MTWLB can be effective for those cases in which the time region for the ERPs are known in advance. Hence, there is no need to search for the time window and the computational complexity is reduced significantly. Thirdly, the performance of xDAWN can be affected by the selected epoch length because the optimization of xDAWN is based on the whole epoch. On the contrary, MTWLB estimates the spatial pattern in the specific time window instead of the whole EEG epoch. So changing the epoch length will have no effect on the performance of the MTWLB algorithm.

Regarding the second issue of the effect of features on classification performance, we have demonstrated the performance of three classifiers across the different pipelines. It can be noted that LDA and BLR outperform LR in the CSP pipeline and the pipeline without spatial filtering. This may indicate that LR is more sensitive to the quality of features compared to LDA and BLR. LR is used for modeling the relationship between independent and categorical dependent variables and variable colinearities may have negative effects on its estimation [189]. From the pipelines with appropriate spatial filtering (xDAWN and MTWLB), three classifiers perform closely to each other with MTWLB and LR outperforming the other two methods in the pipeline when using xDAWN. This indicates that the LR classifier performs well with informative features as input. LDA and BLR have been used more widely compared to LR in the literature since LDA performs well even without feature extraction and is simpler to implement [60, 61, 134, 152]. Here we have shown that LR is able to generate very good performance when informative features are extracted from RSVP- EEG.

Results in this chapter partly supports the result in [134] that spatial filtering can improve the overall performance. In this work, we performed a more comprehensive comparison of the spatial filtering pipeline. Firstly, we included a random search [181] for the set of hyperparameters listed in the Table 4.1 in order to attain optimal performance. During experimentation, we found that the number of spatial filters can have critical impact on the final classification performance and this varies across participants. Instead of using a predetermined number [134], we suggest to search for it as a hyperparameter as part of the pipeline in order to optimize the performance. Secondly, the type of spatial filtering used is critical to the classification performance for different BCI paradigms due to different task-related EEG phenomena. For example, CSP has been widely used for motor imagery based BCIs [133, 190, 191], where oscillatory EEG activity is elicited in the experiment [144, 192]. In our study, however, the classification performance of pipelines involving CSP spatial filtering is even worse than pipelines that do not use spatial fil-

tering, which supports the results in [134]. We understand that CSP is used to generate features differently for motor imagery BCI [133, 190]. We clarify that features produced by CSP at least in this study are not helpful for RSVP-EEG. These results suggest that improper use of spatial filtering in a CCCV system can have a negative impact on the classification performance and this conclusion can be extended to other types of EEG-based BCIs systems that utilize ERPs. On the contrary, applying the appropriate spatial filtering technique (e.g., MTWLB or xDAWN for RSVP-EEG in this work) results in reduced computational complexity and improvement in classification performance. This work demonstrates that it is critical to choose the appropriate type of spatial filtering for the signal processing pipeline in CCCV systems.

Furthermore, we compared the time-course source reconstruction performance for the N170 and the P300 by using xDAWN and MTWLB, where these two approaches have the comparable performance for cortically coupled image classification in RSVP paradigms. We have demonstrated the effectiveness of using MTWLB, where higher SNR and better classification performance are achieved for both N170 and P300, compared to xDAWN. This result suggests that MTWLB is able to be used to reconstruct time-course source signals in a RSVP paradigm for neurophysiological research.

The work presented in this chapter addresses **research question 1**: *Can we improve on the extraction of discriminative ERP components while preserving neurophysiological interpretability for a CCCV system?* We showed the comparable classification performance between MTWLB and the current advanced methods (xDAWN and CSP used in this case). We also explained the neurophysiological interpretability and demonstrated the effectiveness of the source signal reconstruction via MTWLB, which indicates MTWLB is able to handle the classification and the neurophysiological interpretability for a CCCV system.

4.5 Conclusion

In this work, we presented a novel spatial filtering approach (MTWLB) for RSVP-EEG. Our results demonstrated comparable performance with the leading method xDAWN although our approach is significantly different i.e., we applied spatial filtering in each independent time window instead of whole epoch. Consequently, the method presents a different set of optimization parameters which may make it suitable for CCCV implementations in particular. Even though there is no statistically significant difference between our proposed method and

xDAWN, MTWLB presents useful properties that lend themselves to certain CCCV performance optimizations not available via xDAWN. First, the method is more robust to EEG epoch length compared to the conventional spatial filtering approaches (e.g., xDAWN and CSP) because its optimization relies on the time window instead of whole epochs. Second, MTWLB can be more effective when the ERP time region is known in advance because there is no need then to search for the most appropriate time window and the computational complexity is reduced significantly. Furthermore this work included a thorough evaluation of single-trial classification pipelines with a number of spatial filters and classifiers in a comprehensive way using a publicly available dataset. We have shown that the selection of a spatial filtering method should correspond to the nature of the ERPs elicited in the task paradigm and that naive application of the approach may not produce good performance. We also demonstrated that even though LDA and BLR are the most prevalent classification approaches used in CCCV research, the LR method can be even more effective for single-trial ERP detection when good quality features are made available, for example, through spatial filtering methods. Finally, we showed the effectiveness of MTWLB for time-course ERP source signals reconstruction compared to xDAWN in a RSVP paradigm with respect to both SNR and classification sides. This result demonstrates that MTWLB is potentially able to be applied for RSVP-EEG data for the neurophysiological research. In summary, this work should help inform designers of CCCV systems of an appropriate spatial filtering and classifier choice at design time based on results generated using a publicly available dataset (allowing for later comparative benchmarks).

Chapter 5

Generative Adversarial Networks: A Survey and Taxonomy

Abstract: *Generative adversarial networks (GANs) have been extensively studied in the past few years. Arguably the revolutionary techniques are in the area of computer vision such as plausible image generation, image to image translation, facial attribute manipulation and similar domains. Despite the significant success achieved in the computer vision field, applying GANs to real-world problems still poses significant challenges, three of which we focus on here: (1) High quality image generation; (2) Diverse image generation; and (3) Stable training. Through an in-depth review of GAN-related research in the literature, we provide an account of the architecture-variants and loss-variants, which have been proposed to handle these three challenges from two perspectives. We propose loss-variants and architecture-variants for classifying the most popular GANs, and discuss the potential improvements with focusing on these two aspects. While several reviews for GANs have been presented to date, none have focused on the review of GAN-variants based on their handling the challenges mentioned above. In this chapter, we review and critically discuss 7 architecture-variant GANs and 9 loss-variant GANs for remedying those three challenges. The objective of this review is to present the current status of GANs and to establish open problems in evaluation which will necessitate as a part solution our contributions via Neuro-AI to come in later chapters. Some notation has been explained in Appendix C. Code related to GAN-variants studied in this work is summarized on https://github.com/sheqi/GAN_Review.*

5.1 Introduction

Generative adversarial networks (GANs) are attracting growing interests in the deep learning community [65, 193–197]. GANs have been applied to various domains such as computer vision [198–205], natural language processing [206–209], time series synthesis [68, 210–213] and semantic segmentation [214–218]. GANs belong to the family of deep generative models (DGMs). Compared to other DGMs e.g., variational autoencoders, GANs offer advantages such as an ability to handle sharp estimated density functions, efficiently generating desired samples, eliminating deterministic bias and good compatibility with the internal neural architecture. These properties have allowed GANs to enjoy success especially in the computer vision field e.g., plausible image generation [219–223], image to image translation [193, 224–230], image super-resolution [216, 231–234] and image completion [235–239].

However, GANs suffer challenges from two aspects: (1) Hard to train — It is non-trivial for discriminator and generator to achieve Nash equilibrium during the training and the generator cannot learn the distribution of the full datasets well, which is known as mode collapse. Lots of work has been carried out in this area [240–243]; and (2) Hard to evaluate — the evaluation of GANs can be considered as an effort to measure the dissimilarity between the real distribution p_r and the generated distribution p_g . Unfortunately, the accurate estimation of p_r is not possible. Thus, it is challenging to produce good estimations of the correspondence between p_r and p_g . Previous work has introduced evaluation metrics for GANs [194, 244–251]. The first aspect concerns the performance for GANs directly e.g., image quality, image diversity and stable training. In this work, we are going to study existing GAN-variants that handle this aspect in the area of computer vision while those readers who are interested in the second aspect can consult [244, 251].

Current GANs research focuses on two directions: (1) Improving the training for GANs; and (2) Deployment of GANs to real-world applications. The former seeks to improve GANs performance and is therefore a foundation for the latter aspect. Considering numerous research work in the literature, we give a brief review on the GAN-variants that focus on improving training in this chapter. The improvement of the training process provides benefits in terms of GANs performance as follows: (1) Improvements in generated image diversity (also known as mode diversity); (2) Increases in generated image quality; and (3) More stable training such as remedying the vanishing gradient for the generator. In order to improve the performance as mentioned above, modification for GANs can be done from either the architectural side or

the loss perspective. We will study the GAN-variants coming from both sides that improve the performance for GANs.

The rest of the chapter is organized as follows: (1) We introduce the search strategy and part of the results for the existing GANs papers in the area of computer vision; (2) We introduce related review work for GANs and illustrate the difference between those reviews and this work; (3) We give a brief introduction to GANs; (4) We review the architecture-variant GANs in the literature; (5) We review the loss-variant GANs in the literature; (6) We summarize the GAN-variants in this study and illustrate their difference and relationships; and (7) We conclude this review and preview likely future research work in the area of GANs.

Many GAN-variants have been proposed in the literature to improve performance. These can be divided into two types: (1) Architecture-variants. The first proposed GAN used fully-connected neural networks [65] so specific types of architecture may be beneficial for specific applications e.g., convolutional neural networks (CNNs) for images and recurrent neural networks (RNNs) for time series data; and (2) Loss-variants. Here different variations of the loss function are explored regarding the equation (1.1) to enable more stable learning of G .

5.2 Search Strategy and Results

A review of the literature was performed to identify research work describing GANs. Papers were first identified through manual search of the online datasets (Google Scholar and IEEE Xplore) through use of the keyword “generative adversarial networks”. Secondly papers related to computer vision were manually selected. This search concluded on 17th May 2019.

A total of 322 papers describing GANs related to computer vision were identified. The earliest paper was in 2014 [65]. Papers were classified into three categories regarding their repository, namely conference, arXiv and journal. More than half of the papers were presented at conferences (204, 63.3%). The rest were journal articles (58, 18.0%) and arXiv pre-prints (60, 18.6%).

Details of searched papers are included in Fig. 5.1 and Fig. 5.2. Figure 5.1 illustrates the number of papers in each year from 2014 to 2019. It can be seen that the number of papers increases each year from 2014 to 2018. As our search ends up on 17th May 2019, this number can not represent the overall number of papers in 2019. Especially there are several upcoming top-tier conferences e.g., CVPR, ICCV, NeurIPS, and ICML, where much more papers may

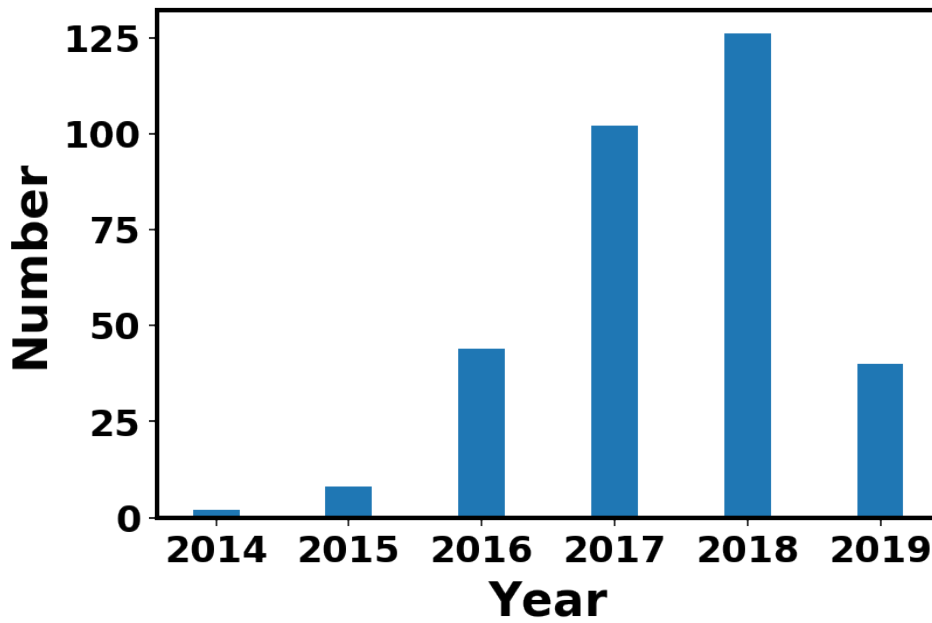


Figure 5.1: Number of papers in each year from 2014 to 17th May 2019.

come out later this year. Even given this situation, the number of papers in 2019 is close to that in 2016. It can be noticed that there is significant rise of papers in 2016 and 2017. Indeed we see lots of exciting research in these two years e.g., CoGAN, f-GAN in 2016 and WGAN, PROGAN in 2017, which pushes the GANs research and exposes GANs to the public. In 2018, GANs still attracts lots of attention and the number of papers is more than that in previous years.

Figure 5.2 illustrates the number of papers published on three repositories, namely confer-

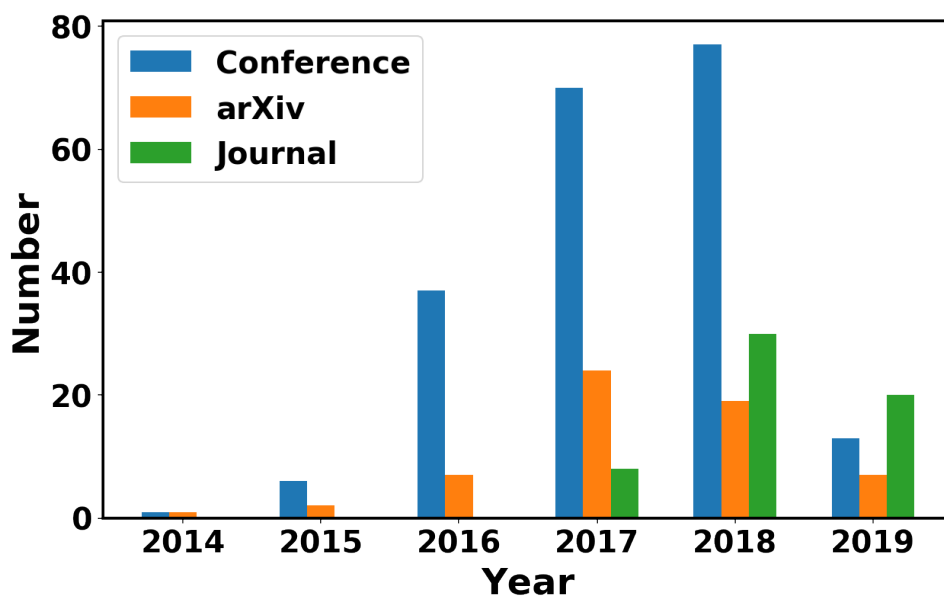


Figure 5.2: Categories of papers from 2014 to 17th May 2019. Papers are categorized as conference, arXiv and journal.

ence, arXiv and journal. Conferences take the largest amount from 2015 to 2018 and dramatic increase appears in 2016 and 2017. As mentioned before, there are several top-tier upcoming conferences later this year, conference supposes to take the lead on the number of papers in 2019. Papers published in journals start to increase since 2017, which may be caused by the reviewing duration for a journal paper that is longer than a conference paper and of course much longer than an arXiv paper. As GANs are well-developed and well-known to researchers from different areas today, the number of journal papers related to GANs supposes to maintain the increasing tendency in 2019. It is interesting that number of arXiv pre-prints reaches a peak in 2017 and then starts to descend. We guess this is caused by more and more papers are accepted by conference and journal so arXiv pre-prints claim the publication details, which leads to the decreasing number of pre-prints on arXiv. This indicates higher quality of GANs research in recent years from the other side. Figure 5.3 gives an illustration on the percentage of each cat-

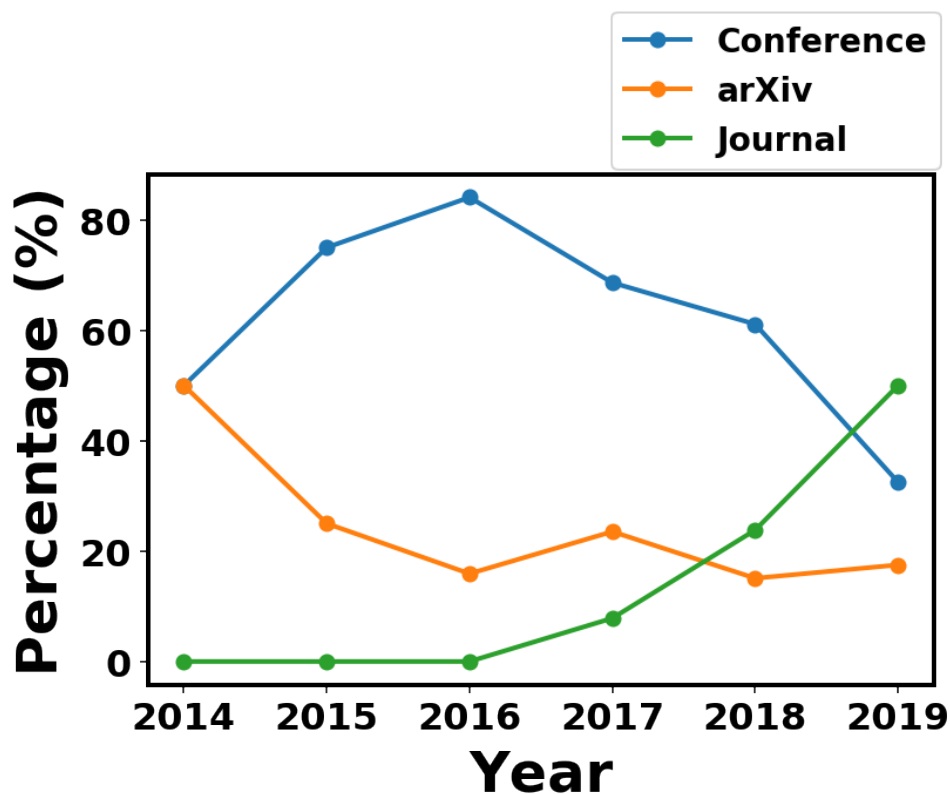


Figure 5.3: Percentages of each category take account the total number of papers in each year.

egory taking account the total number of papers in each year. Supporting results in Fig. 5.2, tendency of number of journal papers keeps going up. Percentage of number of conference papers reaches peak at 2016 then begins to descend. It should be noted that this does not mean the decrease of number of conference papers. This is due to other categories (i.e., arXiv and

journal papers) start to increase.

A detail of searched papers are listed on this link: <https://cutt.ly/GAN-CV-paper>.

5.3 Previous Related Literature Reviews on GANs

There have been previous GANs review papers for example in terms of reviewing GANs performance [94]. That work focuses on the experimental validation across different types of GANs benchmarking on LSUN-BEDROOM [252], CelebA-HQ-128 [253] and the CIFAR10 [254] image datasets. The results suggest that the original GAN [65] with spectral normalization [255] is a good starting choice when applying GANs to a new dataset. A limitation of that review is that the benchmark datasets do not consider diversity in a significant way. Thus the benchmark results tend to focus more on evaluation of the image quality, which may ignore GANs efficacy in producing diverse images. Work [96] surveys different GANs architectures and their evaluation metrics. A further comparison on different architecture-variants' performance, applications, complexity and so on needs to be explored. Papers [95, 97, 98] focus on the investigation of the newest development trends and the applications of GANs. They compare GAN-variants through different applications. Comparing this review to the current review literature, we emphasize an introduction to GAN-variants based on their performance including their ability to produce high quality and diverse images, stable training, ability for handling the vanishing gradient problem, etc. This is all done through the taking of a perspective based on architecture and loss function considerations. This work also provides the comparison and analysis in terms of pros and cons across GAN-variants present in this paper.

5.4 Generative Adversarial Networks

GANs, as a member of the DGMs family, have attracted exponentially growing interest in the deep learning community because of some advantages compared to the traditional DGMs: (1) GANs are able to produce better output than other DGMs. Compared to the most well-known DGMs—variational autoencoder (VAE), GANs are able to produce any type of probability density while VAE is not able to generate sharp images; (2) The GAN framework can train any type of generator network. Other DGMs may have pre-requirements for the generator e.g., the output layer of generator is Gaussian; (3) There is no restriction on the size of the latent variable. These

advantages have led GANs to achieve the state-of-the-art performance on producing synthetic data especially for image data.

5.5 Architecture-variant GANs

There are many types of architecture-variants proposed in the literature (see Fig. 5.4) [223, 224, 256–258]. Architecture-variant GANs are mainly proposed for the purpose of different applications e.g., image to image transfer [224], image super resolution [231], image completion [259], and text-to-image generation [260]. In this section, we provide a review on architecture-variants that helps improve the performance for GANs from three aspects mentioned before, namely improving image diversity, improving image quality and more stable training. Review for those architecture-variants for different applications can be referred to work [96, 97].

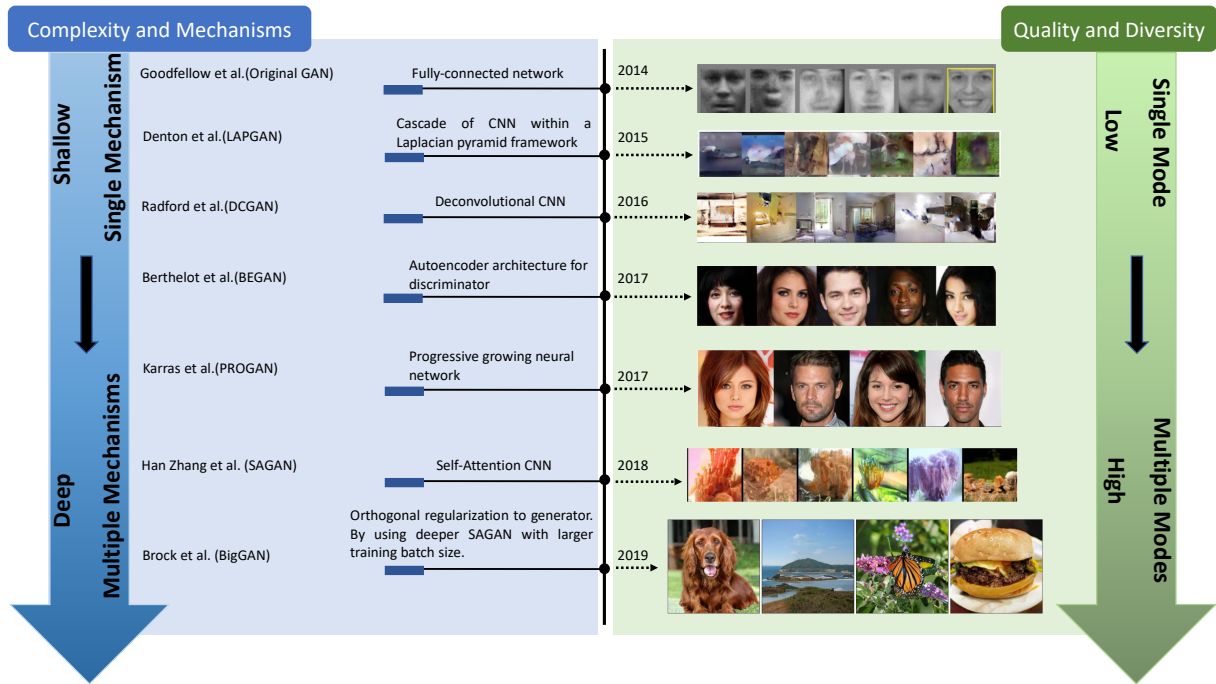


Figure 5.4: Timeline of architecture-variant GANs. Complexity in blue stream refers to size of the architecture and the computational cost such as batch size. Mechanisms refer to the number of types of models used in the architecture (e.g., BEGAN uses an autoencoder architecture for its discriminator while a deconvolutional neural network is used for the generator. In this case, two mechanisms are used).

5.5.1 Fully-connected GAN (FCGAN)

The original GAN paper [65] used fully-connected neural networks for both generator and discriminator. This architecture-variant was applied for some simple image datasets i.e., MNIST [80],

CIFAR-10 [254] and Toronto Face Dataset. It did not demonstrate good generalization performance for more complex image types. The FCGAN was trained by using the stochastic gradient descent optimizer.

5.5.2 Laplacian Pyramid of Adversarial Networks (LAPGAN)

LAPGAN was proposed for the production of higher resolution images from lower resolution input GAN [261]. Figure 5.5 demonstrates the up-sampling process of generator in LAPGAN

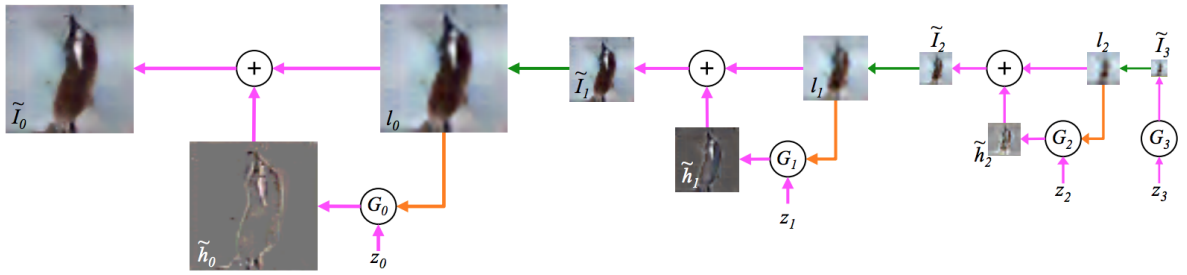


Figure 5.5: Up-sampling process of generator in LAPGAN (from right to left). The up-sampling process is marked using green arrow and conditioning process via CGAN [262] is marked using orange arrow. The process starts to use G_3 to generate image \tilde{I}_3 and then up-sample the image \tilde{I}_3 to l_2 . Together with another noise z_2 , G_2 generates a difference image \tilde{h}_2 and add \tilde{h}_2 to l_2 be the generated image \tilde{I}_2 . The rest can be done in the same manner. LAPGAN contains 3 generators in this work in order to up-sample the image. Figure from [261].

from right to left. LAPGAN utilizes a cascade of CNNs within a Laplacian pyramid framework [263] to generate high quality images.

5.5.3 Deep Convolutional GAN (DCGAN)

DCGAN is the first work that applied a deconvolutional neural networks architecture for G [256]. Figure 5.6 illustrates the proposed architecture for G . Deconvolution was proposed to visualize the features for a CNN and has shown good performance for CNNs visualization [264]. DCGAN deploys the spatial up-sampling ability of the deconvolution operation for G , which enables the generation of higher resolution images using GANs. The Adam optimizer was used for training the DCGAN in this work.

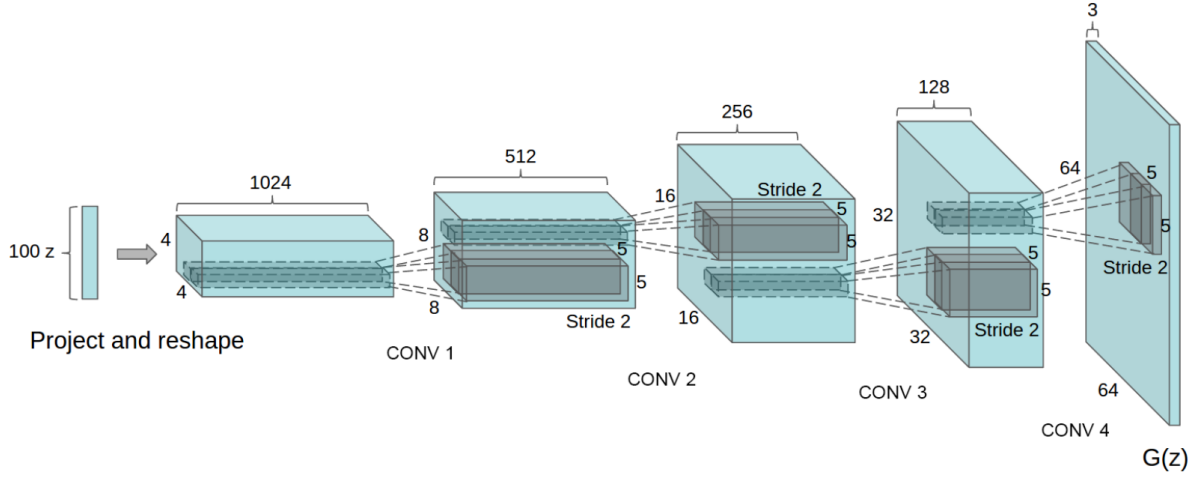


Figure 5.6: Detail of DCGAN architecture for generator. This generator successfully generates 64×64 pixel image for LSUN scene dataset, which is more complex than the datasets used in the original work. Figure from [256].

5.5.4 Boundary Equilibrium GAN (BEGAN)

BEGAN uses an autoencoder architecture for the discriminator which was first proposed in EBGAN [265] (see Fig. 5.7). Compared to traditional optimization, the BEGAN matches the

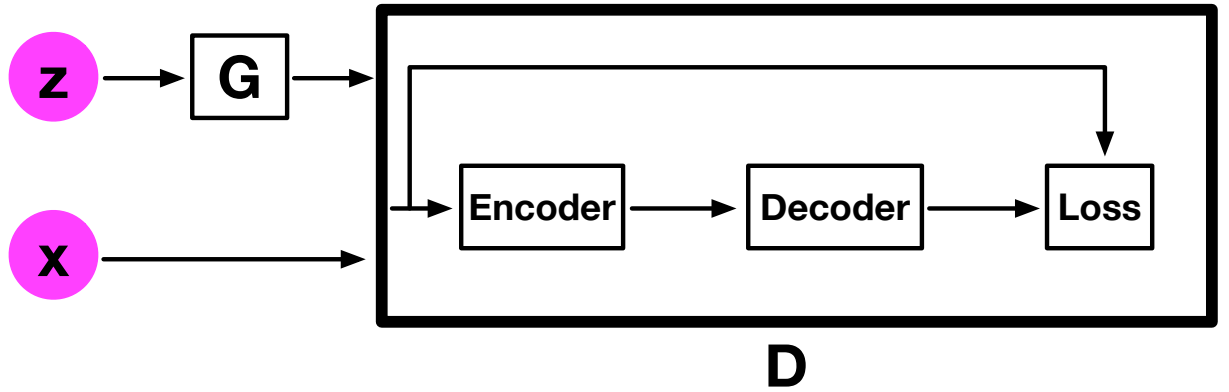


Figure 5.7: Illustration of BEGAN architecture. z is the latent variable for G and x is input image. BEGAN deploys autoencoder architecture for the discriminator. Loss is calculated using L_1 or L_2 norm at pixel level.

autoencoder loss distributions using a loss derived from the Wasserstein distance instead of matching data distributions directly. This modification helps G to generate easy-to-reconstruct data for the autoencoder at the beginning because the generated data is close to 0 and the real data distribution has not been learned accurately yet, which prevents D easily winning G at the early training stage. The Adam optimizer was used to train BEGAN in this work.

5.5.5 Progressive GAN (PROGAN)

PROGAN involves progressive steps toward the expansion of the network architecture [258]. This architecture uses the idea of progressive neural networks first proposed in [266]. This technology does not suffer from forgetting and can leverage prior knowledge via lateral connections to previously learned features. Consequently it is widely applied for learning complex task sequences. Figure 5.8 demonstrates the training process for PROGAN. Training starts with

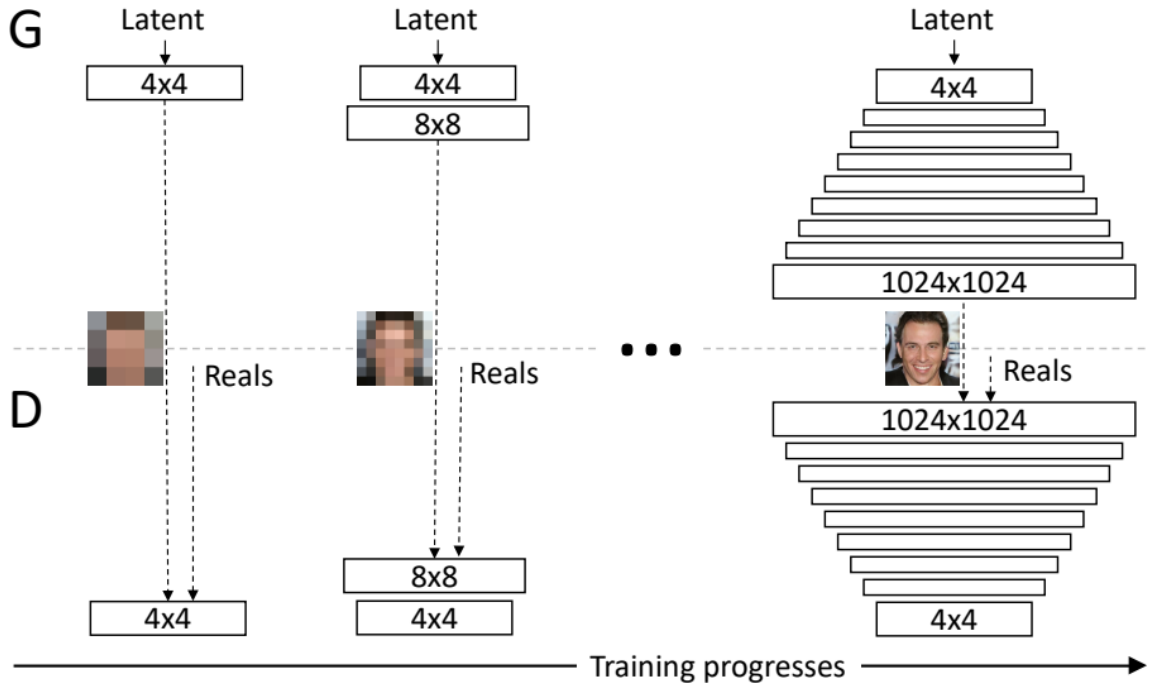


Figure 5.8: Progressive growing step for PROGAN during the training process. Training starts with 4×4 pixels image resolution. With the training step growing, layers are incrementally added to *G* and *D* which increases the resolution for the generated images. All existing layers are trainable throughout the training stage. Figure from [258].

low resolution 4×4 pixels image. Both *G* and *D* start to grow with the training progressing. Importantly, all variables remain trainable throughout this growing process. This progressive training strategy enables substantially more stable learning for both networks. By increasing the resolution little by little, the networks are continuously asked a much simpler question compared to the end goal of discovering a mapping from latent vectors. All current state-of-the-art GANs employ this type of training strategy and it has resulted in impressive, plausible images [219, 258, 267]. The Adam optimizer was used to train the PROGAN.

5.5.6 Self-attention GAN (SAGAN)

Traditional CNNs can only capture local spatial information and the receptive field may not cover enough structure, which causes CNN-based GANs to have difficulty in learning multi-class image datasets (e.g., ImageNet) and the key components in generated images may shift e.g., the nose in a face-generated image may not appear in right position. Self-attention mechanism have been proposed to ensure large receptive field and without sacrificing computational efficiency for CNNs [268]. SAGAN deploys a self-attention mechanism in the design of the discriminator and generator architectures for GANs [269] (see Fig. 5.9). Benefiting from the

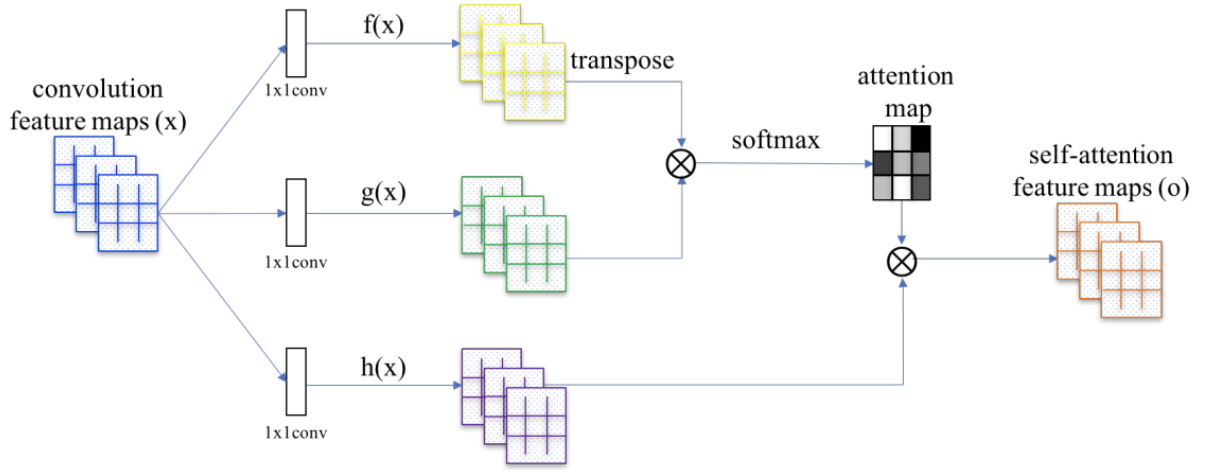


Figure 5.9: Self-attention mechanism architecture proposed in the paper. f , g and h are corresponding to query, key and value in the self-attention mechanism. The attention map indicates the long-range spatial dependencies. The \otimes is matrix multiplication. Figure from [269].

self-attention mechanism, SAGAN is able to learn global and long-range dependencies for generating images. It has achieved great performance on multi-class image generation based on the ImageNet datasets. The Adam optimizer was used for training the model in this work.

5.5.7 BigGAN

BigGAN [267] has also achieved state-of-the-art performance on the ImageNet datasets. Its design is based on SAGAN and it has been demonstrated that the increase in batch size and the model complexity can dramatically improve GANs performance with respect to complex image datasets. The Adam optimizer was used for training the BigGAN in this work.

5.5.8 Summary

We have provided an overview of architecture-variant GANs which aim to improve performance based on the three key challenges: (1) Image quality; (2) Mode diversity; and (3) Vanishing gradient. An illustration of relative performance can be found in Fig. 5.10.

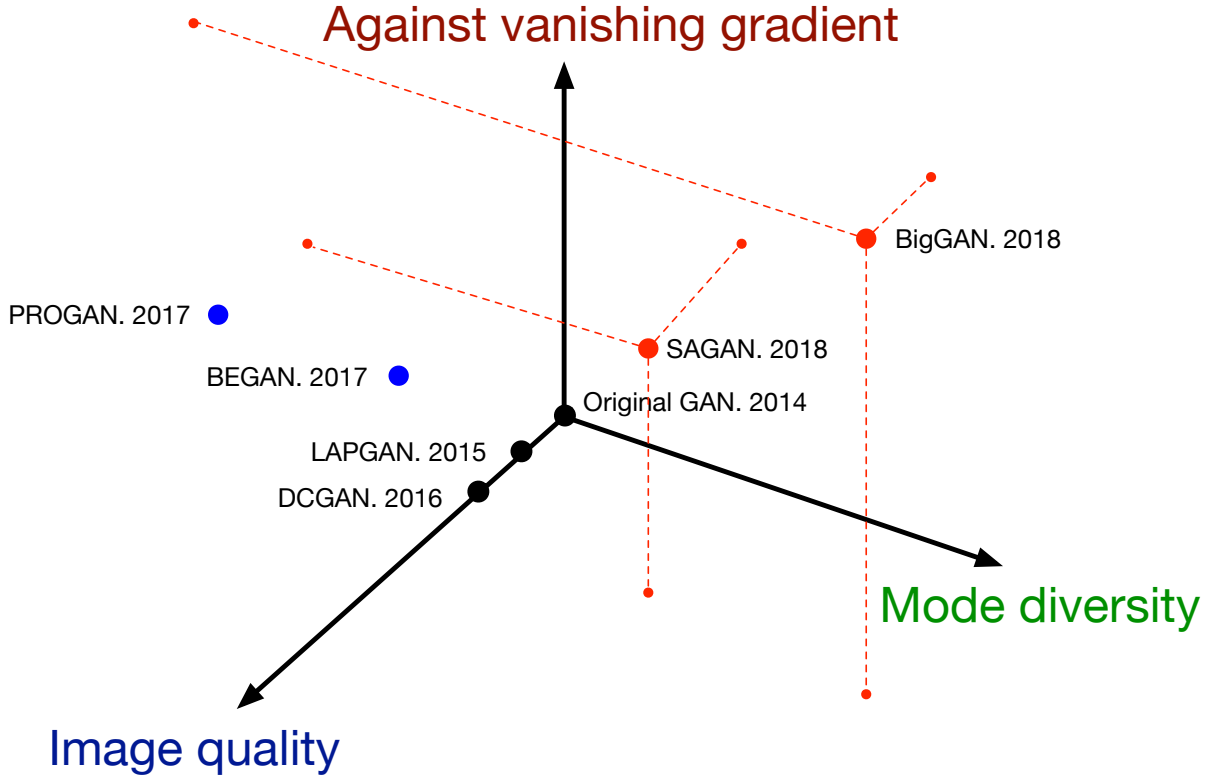


Figure 5.10: Summary of recent architecture-variant GANs for solving the three challenges. The challenges are categorized by three orthogonal axes. **A larger value for each axis indicates better performance.** Red points indicate GAN-variants which cover all three challenges, blue points cover two, and black points cover only one challenge.

All proposed architecture-variants are able to improve image quality. SAGAN was proposed for improving the capacity of multi-class learning in GANs, the goal of which is to produce more diverse images. Benefiting from the SAGAN architecture, BigGAN was designed for improving both image quality and image diversity. It should be noted that both PROGAN and BigGAN are able to produce high resolution images. BigGAN realizes this higher resolution by increasing the batch size and the authors mentioned that a progressive growing [258] operation is unnecessary when the batch size is large enough (2048 used in the original paper [267]). However, a progressive growing operation is still needed when GPU memory is limited (a large batch size requires significant GPU memory). Benefiting from spectrum normalization (SN), which will be discussed in loss-variant GANs part, both SAGAN and BigGAN are effective

for the vanishing gradient challenge. These milestone architecture-variants indicate a strong advantage of GANs — compatibility, where a GAN is open to any type of neural architecture. This property enables GANs to be applied to many different applications.

Regarding the improvements achieved by different architecture-variant GANs, we next present an analysis on the interconnections and comparisons between the architecture-variants presented here. Starting with the FCGAN described in the original GAN paper, this architecture-variant can only generate simple image datasets. Such a limitation is caused by the network architecture, where the capacity of FC networks is very limited. Research on improving the performance of GANs starts from designing more complex architectures for GANs. A more complex image datasets (e.g., ImageNet) has higher resolution and diversity comparing to simple image datasets (e.g., MNIST) and needs accordingly more sophisticated approaches.

In the context of producing higher resolution images, one obvious approach is to increase the size of generator. LAPGAN and DCGAN up-sample the generator based on such a perspective. Benefiting from the concise deconvolutional up-sampling process and easy generalization of DCGAN, the architecture in DCGAN is more widely used in the GANs literature. It should be noticed that most GANs in the computer vision area use the deconvolutional neural network as the generator, which was first used in DCGAN. Therefore, DCGAN is one of the classical GAN-variants in the literature.

The ability to produce high quality images is an important aspect of GANs clearly. This can be improved through judicious choice of architecture. BEGAN and PROGAN demonstrate approaches from this perspective. With the same architecture used for the generator in DCGAN, BEGAN redesigns the discriminator by including encoder and decoder, where the discriminator tries to distinguish the difference between the generated and autoencoded images in pixel space. Image quality has been improved in this case. Based on DCGAN, PROGAN demonstrates a progressive approach that incrementally trains an architecture similar to DCGAN. This novel approach cannot only improve image quality but also produce higher resolution images.

Producing diverse images is the most challenging task for GANs and it is very difficult for GANs to successfully produce images such as those represented in the ImageNet sets. It is difficult for traditional CNNs to learn global and long-range dependencies from images. Thanks to self-attention mechanism though, approaches such as those in SAGAN integrate the self-attention mechanism to both discriminator and generator, which helps GANs a lot in terms of learning multi-class images. Moreover, BigGAN, which can be considered an extension of

SAGAN, introduces a deeper GAN architecture with a very large batch size, which produces high quality and diverse images as in ImageNet and is the current state-of-the-art.

5.6 Loss-variant GANs

Another design decision in GANs which significantly impacts performance is the choice of loss function in equation (1.1), which is

$$\min_G \max_D \mathbb{E}_{\mathbf{x} \sim p_r} \log[D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z} \log[1 - D(G(\mathbf{z}))]$$

While the original GAN work [65] has already proved global optimality and the convergence of GANs training. It still highlighted the instability problem which can arise when training a GAN. The problem is caused by the global optimality criterion as stated in [65]. Global optimality is achieved when an optimal D is reached for any G . So the optimal D is achieved when the derivative of D for the loss in equation (1.1) equals 0. So we have

$$\begin{aligned} -\frac{p_r(\mathbf{x})}{D(\mathbf{x})} + \frac{p_g(\mathbf{x})}{1 - D(\mathbf{x})} &= 0 \\ D^*(\mathbf{x}) &= \frac{p_r(\mathbf{x})}{p_r(\mathbf{x}) + p_g(\mathbf{x})} \end{aligned} \tag{5.1}$$

where \mathbf{x} represents the real data and generated data, $D^*(\mathbf{x})$ is the optimal discriminator, $p_r(\mathbf{x})$ is the real data distribution and $p_g(\mathbf{x})$ is the generated data distribution. We have got the optimal discriminator D so far. When we have the optimal D , the loss for G can be visualized by substituting $D^*(\mathbf{x})$ into equation (1.1)

$$\mathcal{L}_G = \mathbb{E}_{\mathbf{x} \sim p_r} \log \frac{p_r(\mathbf{x})}{\frac{1}{2}[p_r(\mathbf{x}) + p_g(\mathbf{x})]} + \mathbb{E}_{\mathbf{x} \sim p_g} \log \frac{p_g(\mathbf{x})}{\frac{1}{2}[p_r(\mathbf{x}) + p_g(\mathbf{x})]} - 2 \cdot \log 2 \tag{5.2}$$

Equation (5.2) demonstrates the loss function for a GAN when discriminator is optimized and it is related to two important probability measurement metrics. One is Kullback–Leibler (KL) divergence which is defined as

$$KL(p_1 || p_2) = \mathbb{E}_{\mathbf{x} \sim p_1} \log \frac{p_1}{p_2} \tag{5.3}$$

and the other is Jensen-Shannon (JS) divergence which is stated as

$$JS(p_1||p_2) = \frac{1}{2}KL(p_1||\frac{p_1+p_2}{2}) + \frac{1}{2}KL(p_2||\frac{p_1+p_2}{2}) \quad (5.4)$$

Thus the loss for G regarding the optimal D in equation (5.2) can be reformulated as

$$\mathcal{L}_G = 2 \cdot JS(p_r||p_g) - 2 \cdot \log 2 \quad (5.5)$$

which indicates that the loss for G now equally becomes the minimization of the JS divergence between p_r and p_g . With the training D step by step, the optimization of G will be closer to the minimization of JS divergence between p_r and p_g . We now start to explain the unstable training problem, where D often easily wins G . This unstable training problem is actually caused by the JS divergence in equation (5.4). Give an optimal D , the objective of optimization for equation (5.5) is to move p_g toward p_r (see Fig. 5.11). JS divergence for the three plots

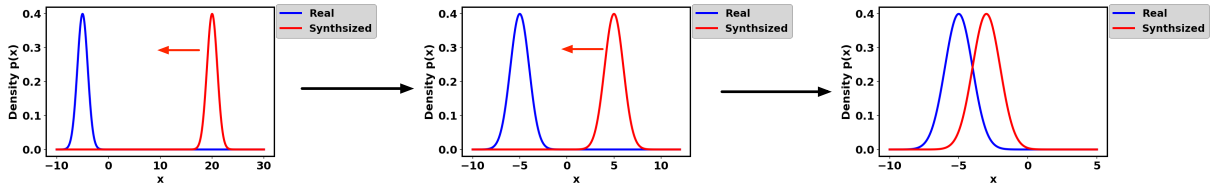


Figure 5.11: Illustration of training progress for a GAN. Two normal distributions are used here for visualization. Given an optimal D , the objective of GANs is to update G in order to move the generated distribution p_g (red) towards the real distribution p_r (blue) (G is updated from left to right in this figure). Left: initial state, middle: during training, right: training converging). However, JS divergence for the left two figures are both 0.693 and the figure on the right is 0.336, indicating that JS divergence does not provide sufficient gradient at the initial state.

from left to right are 0.693, 0.693 and 0.336, which indicates that JS divergence stays constant ($\log 2 = 0.693$) if there is no overlap between p_r and p_g . Figure 5.12 demonstrates the change of JS divergence and its gradient corresponding to the distance between p_r and p_g . It can be seen that JS divergence is constant and its gradient is almost 0 when the distance is greater than 5, which indicates that training process does not have any effect on G . The gradient of JS divergence for training the G is non-zero only when p_g and p_r have substantial overlap i.e., the vanishing gradient will arise for G when D is close to optimal. In practice, the possibility that p_r and p_g do not overlap or have negligible overlap is very high [270].

The original GANs work [65] also highlighted the minimization of $-\mathbb{E}_{\mathbf{x} \sim p_g} \log[D(\mathbf{x})]$ for training G to avoid a vanishing gradient. However, this training strategy will lead to another

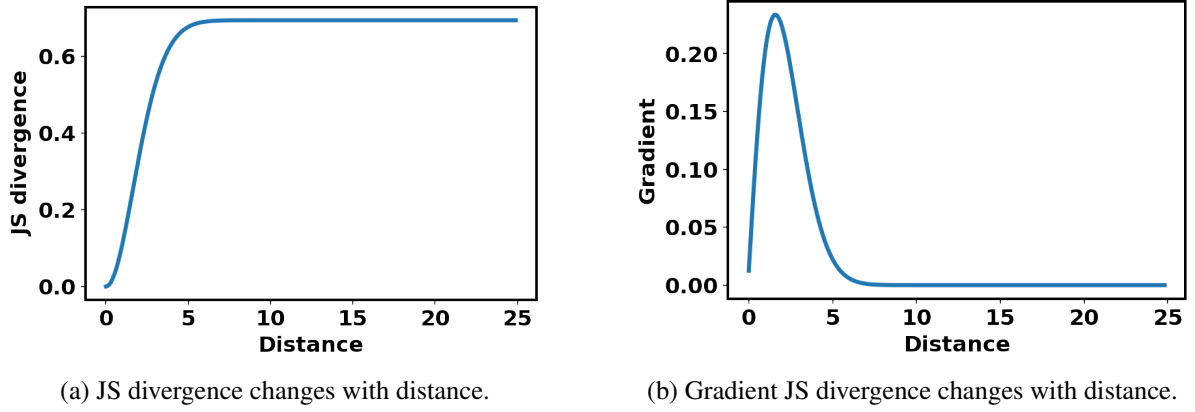


Figure 5.12: JS divergence and gradient change with the distance between p_r and p_g . The distance is the difference between two distribution means.

problem called mode dropping. First, let us examine $KL(p_g||p_r) = \mathbb{E}_{\mathbf{x} \sim p_g} \log \frac{p_g}{p_r}$. With an optimal discriminator D^* , $KL(p_g||p_r)$ can be reformulated as

$$\begin{aligned}
 KL(p_g||p_r) &= \mathbb{E}_{\mathbf{x} \sim p_g} \log \frac{p_g(\mathbf{x})/(p_r(\mathbf{x}) + p_g(\mathbf{x}))}{p_r(\mathbf{x})/(p_r(\mathbf{x}) + p_g(\mathbf{x}))} \\
 &= \mathbb{E}_{\mathbf{x} \sim p_g} \log \frac{1 - D^*(\mathbf{x})}{D^*(\mathbf{x})} \\
 &= \mathbb{E}_{\mathbf{x} \sim p_g} \log[1 - D^*(\mathbf{x})] - \mathbb{E}_{\mathbf{x} \sim p_g} \log[D^*(\mathbf{x})]
 \end{aligned} \tag{5.6}$$

The alternative loss form for G now can be stated by switching the order of the two sides in equation (5.6)

$$\begin{aligned}
 -\mathbb{E}_{\mathbf{x} \sim p_g} \log[D^*(\mathbf{x})] &= KL(p_g||p_r) - \mathbb{E}_{\mathbf{x} \sim p_g} \log[1 - D^*(\mathbf{x})] \\
 &= KL(p_g||p_r) - 2 \cdot JS(p_r||p_g) + 2 \cdot \log 2 + \mathbb{E}_{\mathbf{x} \sim p_r} \log[D^*(\mathbf{x})]
 \end{aligned} \tag{5.7}$$

where the alternative loss for G in equation (5.7) is only affected by the first two terms (the last two terms are constant). It can be noticed that the optimization in equation (5.7) is contradictory because the first term aims to push the generated distribution toward the real distribution while the second term aim to push in the opposite direction (the negative sign). This will cause an unstable numerical gradient for training G . More importantly, KL divergence is an asymmetrical distribution measurement highlighted below

- When $p_g(\mathbf{x}) \rightarrow 0$, $p_r(\mathbf{x}) \rightarrow 1$, $KL(p_g||p_r) \rightarrow 0$
- When $p_g(\mathbf{x}) \rightarrow 1$, $p_r(\mathbf{x}) \rightarrow 0$, $KL(p_g||p_r) \rightarrow +\infty$

The penalization for two instances of poor performance made by G are totally different. The first instance of poor performance is that G is not producing a reasonable range of samples and yet incurs a very small penalization. The second instance of poor performance concerns G producing implausible samples but has very large penalization. The first example concerns the fact that the generated samples lack diversity while the second concerns that fact that the generated samples are not accurate. Considering this first case, G generates repeated but “safe” samples instead of taking risk to generate diverse but “unsafe” samples, which leads to the mode collapse problem. In summary, using the original loss in equation (1.1) will result in the vanishing gradient for training G and using the alternative loss in equation (5.7) will incur the mode collapse problem. These kind of problems cannot be solved by changing the GANs architectures. Therefore, it could be argued that ultimate GANs problem stems from the design of the loss function and that innovative ideas for this redesign of the loss function may solve the problem.

Loss-variant GANs have been researched extensively to improve the stability of training GANs.

5.6.1 Wasserstein GAN (WGAN)

WGAN [271] has successfully solved the two problems for the original GAN by using the Earth mover (EM) or Wasserstein-1 [272] distance as the loss measure for optimization. The EM distance is defined as

$$W(p_r, p_g) = \inf_{\gamma \in \Pi(p_r, p_g)} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma} \|\mathbf{x} - \mathbf{y}\| \quad (5.8)$$

where $\Pi(p_r, p_g)$ denotes the set of all joint distributions and $\gamma(\mathbf{x}, \mathbf{y})$ whose marginals are p_r and p_g . Compared with KL divergence and JS divergence, EM is able to reflect distance even when p_r and p_g do not overlap and it is also continuous and thus able to provide meaningful gradient for training the generator. Figure 5.20 illustrates the gradient of WGAN compared to the original GAN. It is noticeable that WGAN has a smooth gradient for training the generator spanning the complete space. However, the infimum in equation (5.8) is intractable but the creators demonstrate that instead the Wasserstein distance can be estimated as

$$\max_{w \sim \mathcal{W}} \mathbb{E}_{\mathbf{x} \sim p_r} [f_w(\mathbf{x})] - \mathbb{E}_{\mathbf{z} \sim p_z} [f_w(G(\mathbf{z}))] \quad (5.9)$$

where f_w can be realized by D but has some constraints (for details the interested reader can refer to the original work [271]) and \mathbf{z} is the input noise for G . So w here is the parameters in D and D aims to maximize equation (5.9) in order to make the optimization distance equivalent to Wasserstein distance. When D is optimized, equation (5.8) will become the Wasserstein distance and G aims to minimize it. So the loss for G is

$$- \min_G \mathbb{E}_{\mathbf{z} \sim p_z} [f_w(G(\mathbf{z}))] \quad (5.10)$$

An important difference between WGAN and the original GAN is the function of D . The D in the original work is used as a binary classifier but D used in WGAN is to fit the Wasserstein distance, which is a regression task. Thus, the sigmoid in the last layer of D is removed in the WGAN. The RMSProp optimizer was used for training the WGAN as authors reported that momentum based optimizers such as the Adam optimizer will cause unstable issues.

5.6.2 WGAN-GP

Even though WGAN has been shown to be successful in improving the stability of GAN training, it is not well generalized for a deeper model. Experimentally it has been determined that most WGAN parameters are localized at -0.01 and 0.01 because of parameter clipping. This will dramatically reduce the modeling capacity of D . WGAN-GP has been proposed using gradient penalty for restricting $\|f\|_L \leq K$ for the discriminator [273] and the modified loss for discriminator now becomes

$$\mathcal{L}_D = \mathbb{E}_{\mathbf{x}_g \sim p_g} [D(\mathbf{x}_g)] - \mathbb{E}_{\mathbf{x}_r \sim p_r} [D(\mathbf{x}_r)] + \lambda \mathbb{E}_{\hat{\mathbf{x}} \sim p_{\hat{x}}} [(\|\nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}})\|_2 - 1)^2] \quad (5.11)$$

where \mathbf{x}_r is sample data drawn from the real data distribution p_r , \mathbf{x}_g is sample data drawn from the generated data distribution p_g and $p_{\hat{x}}$ sampling uniformly along the straight lines between pairs of points sampled from the real data distribution p_r and the generated data distribution p_g . The first two terms are original loss in WGAN and the last term is the gradient penalty. WGAN-GP demonstrates a better distribution of trained parameters compared to WGAN (Fig. 5.13) and better stability performance during training of GANs. The Adam optimizer was used in this work for training the model.

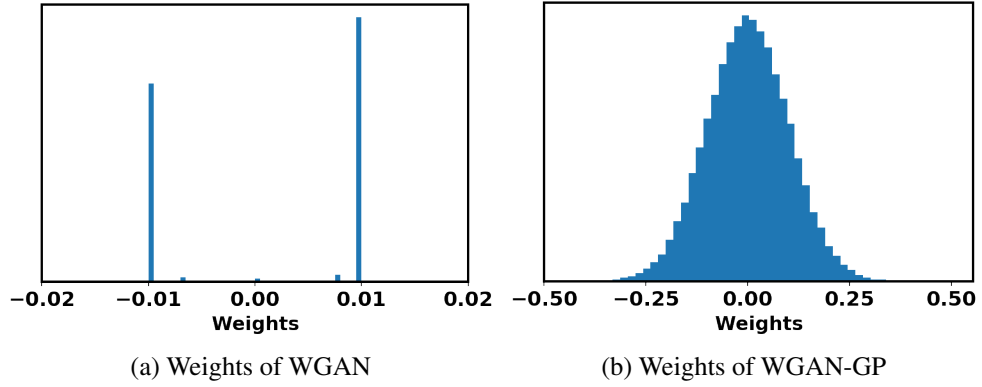


Figure 5.13: Comparison of parameter distribution between WGAN and WGAN-GP. Left is WGAN and right is WGAN-GP. Figure from [273].

5.6.3 Least Square GAN (LSGAN)

The LSGAN¹ is a new approach proposed in [274] to remedy the vanishing gradient problem for G from the perspective of the decision boundary determined by the discriminator. This work argued that the decision boundary for D of the original GAN penalizes very small error to update G for those generated samples that are far away from the decision boundary. The author proposed using a least square loss for D instead of sigmoid cross entropy loss stated in the original paper [65]. The proposed loss function is defined as

$$\begin{aligned} \min_D \mathcal{L}_D &= \frac{1}{2} \mathbb{E}_{\mathbf{x} \sim p_r} [(D(\mathbf{x}) - b)^2] + \frac{1}{2} \mathbb{E}_{\mathbf{z} \sim p_z} [(D(G(\mathbf{z})) - a)^2] \\ \min_G \mathcal{L}_G &= \frac{1}{2} \mathbb{E}_{\mathbf{z} \sim p_z} [(D(G(\mathbf{z})) - c)^2] \end{aligned} \quad (5.12)$$

where a is the label for the generated samples, b is the label for the real samples and c is the value that G wants D to believe for the generated samples. This modified change has two benefits: (1) The new decision boundary made by D penalizes large error arising from those generated samples that are far away from the decision boundary, which pushes those “bad” generated samples towards the decision boundary. This is beneficial in terms of generating improved image quality; (2) Penalizing the generated samples far away from the decision boundary can provide more gradient when updating the G , which remedies the vanishing gradient problems for training G . Figure 5.14 demonstrates the comparison of decision boundaries for LSGAN and the original GAN. The decision boundaries for D that have been trained by the original sigmoid cross entropy loss and the proposed least square loss are different.

¹It should be noted here LSGAN is different from LS-GAN(we will introduce later). The loss functions of these two GAN-variants are significantly different.

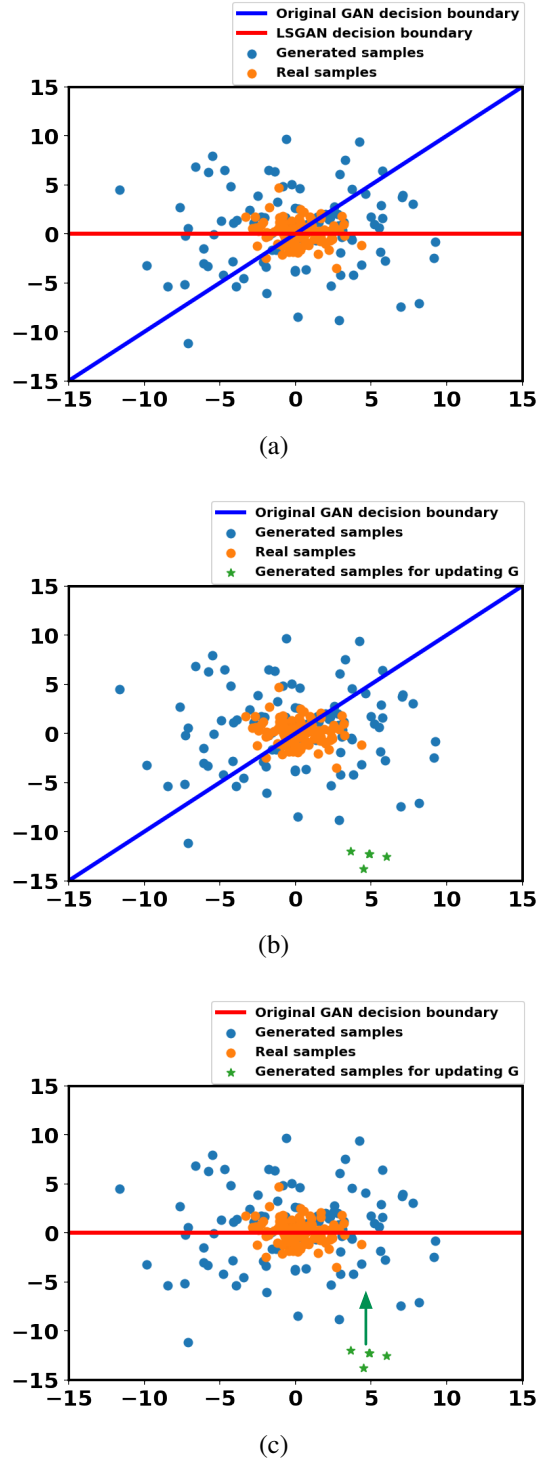


Figure 5.14: Decision boundary illustration of original GAN and LSGAN. (a). Decision boundaries for D of original GAN and LSGAN. (b). Decision boundary of D for the original GAN. It gets small errors for the generated samples, which is far away from the decision boundary (in green), for updating G . (c). Decision boundary for D of LSGAN. It penalizes the large error for generated sample that is far away from the boundary (in green). Thus it pushes generated samples (in green) toward the boundary [274].

The work [274] has proven that the optimization of LSGAN is equivalent to minimizing the Pearson χ^2 divergence between $p_r + p_g$ and $2p_g$ when a , b and c satisfy the condition of $b - c = 1$ and $b - a = 2$. Similar to WGAN, D here behaves as regression and the sigmoid is also removed. The Adam optimizer was used for training the LSGAN in this work.

5.6.4 f-GAN

f-GAN summarized that GANs can be trained by using an f-divergence [275]. f-divergence is a function $D_f(P||Q)$ that measures the difference between probability distributions P and Q . e.g., KL divergence, JS divergence and Pearson χ^2 as mentioned before. This work discusses the efficacy of various divergence function in terms of training complexity and the quality of the generative models.

5.6.5 Unrolled GAN (UGAN)

UGAN was a design proposed to solve the problem of mode collapse for GANs during training [276]. The core design innovation of UGAN is the addition of a gradient term for updating G that captures how the discriminator would react to a change in the generator. The optimal parameter for D can be expressed as the fixed point of an iterative optimization procedure

$$\begin{aligned}\theta_D^0 &= \theta_D \\ \theta_D^{k+1} &= \theta_D^k + \eta^k \frac{df(\theta_G, \theta_D^k)}{d\theta_D^k} \\ \theta_D^*(\theta_G) &= \lim_{k \rightarrow \infty} \theta_D^k\end{aligned}\tag{5.13}$$

where η^k is the learning rate, θ_D represents parameters for D and θ_G represents parameters for G . The surrogate loss by unrolling for K steps can be expressed as

$$f_K(\theta_G, \theta_D) = f(\theta_G, \theta_D^K(\theta_G, \theta_D))\tag{5.14}$$

This surrogate loss is then used for updating parameters for D and G

$$\begin{aligned}\theta_G &\leftarrow \theta_G - \eta \frac{df_K(\theta_G, \theta_D)}{d\theta_G} \\ \theta_D &\leftarrow \theta_D + \eta \frac{df(\theta_G, \theta_D)}{d\theta_D}\end{aligned}\tag{5.15}$$

Figure. 5.15 illustrates the computation graph for an unrolled GAN with 3 unrolling steps.

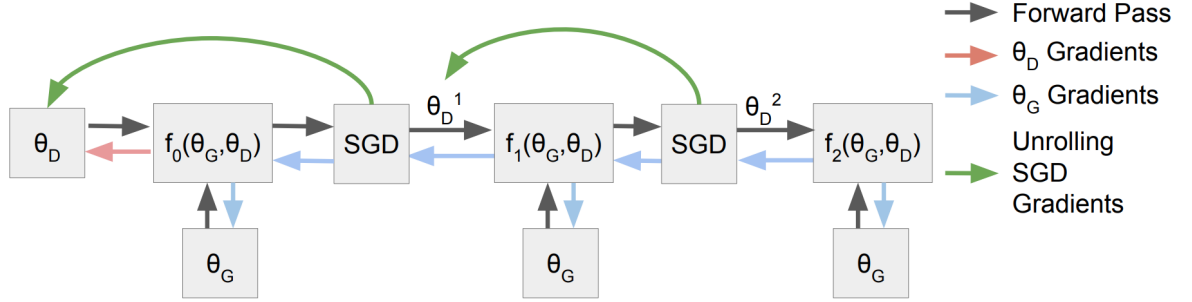


Figure 5.15: An example of computation for an unrolled GAN with 3 unrolling steps. G and D update using equation (5.15). Each step k uses the gradients of f_k regarding θ_D^k stated in the equation (5.13). Figure from [276].

Equation (5.16) illustrates the gradient for updating G

$$\frac{df_K(\theta_G, \theta_D)}{d\theta_G} = \frac{\partial f(\theta_G, \theta_D^K(\theta_G, \theta_D))}{\partial \theta_G} + \frac{\partial f(\theta_G, \theta_D^K(\theta_G, \theta_D))}{\partial \theta_D^K(\theta_G, \theta_D)} \frac{d\theta_D^K(\theta_G, \theta_D)}{d\theta_G} \quad (5.16)$$

It should be noted that the first term in equation (5.16) is the gradient for the original GAN. The second term here reflects how D reacts to changes in G . If G tends to collapse to one mode, D will increase the loss for G . Thus, this unrolled approach is able to prevent mode collapse in GANs. The Adam optimizer was unrolled for training the model in this work.

5.6.6 Loss Sensitive GAN (LS-GAN)

LS-GAN was introduced to train the generator to produce realistic samples by minimizing the designated margins between real and generated samples [277]. This work argued that the problems such as the vanishing gradient and mode collapse as appearing in the original GAN is caused by a non-parametric hypothesis that the discriminator is able to distinguish any type of probability distribution between real samples and generated samples. As mentioned before, it is very normal for the overlap between the real samples distribution and the generated samples distribution to be negligible. Moreover, D is also able to separate real samples and generated samples. The JS divergence will become a constant under this situation, where the vanishing gradient arises for G . In LS-GAN, the classification ability of D is restricted and is learned by a loss function $L_\theta(x)$ parameterized with θ , which assumed that a real sample ought to have smaller loss than a generated sample. The loss function can be trained as the following

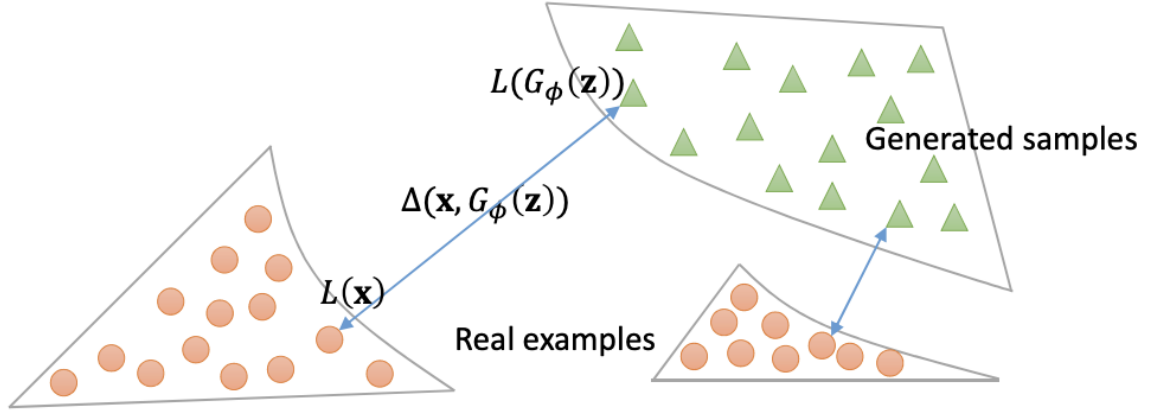


Figure 5.16: Demonstration of the loss in equation (5.18). $\Delta(\mathbf{x}, G(\mathbf{z}))$ is used to separate real samples and generated samples. If some generated samples are close to real samples enough, LS-GAN will work focus on other generated samples that are far away from the real samples. This optimization loss puts restriction on D to prevent it from separating generated and real samples perfectly. Thus, it solves the vanishing gradient problem arises in original GAN. ($G_\phi(\mathbf{z})$ here is equivalent to $G(\mathbf{z})$ where ϕ represents the parameters for generator). Figure from [277].

constraint

$$L_\theta(\mathbf{x}) \leq L_\theta(G(\mathbf{z})) - \Delta(\mathbf{x}, G(\mathbf{z})) \quad (5.17)$$

where $\Delta(\mathbf{x}, G(\mathbf{z}))$ is the margin measuring the difference between real samples and generated samples. This constraint indicates that a real sample is separated from a generated sample by at least a margin of $\Delta(\mathbf{x}, G(\mathbf{z}))$. The optimization for the LS-GAN is then stated as

$$\begin{aligned} \min_D \mathcal{L}_D &= \mathbb{E}_{\mathbf{x} \sim p_r} L_\theta(\mathbf{x}) + \lambda \mathbb{E}_{\mathbf{x} \sim p_r, \mathbf{z} \sim p_z} (\Delta(\mathbf{x}, G(\mathbf{z})) + L_\theta(\mathbf{x}) - L_\theta(G(\mathbf{z})))_+ \\ \min_G \mathcal{L}_G &= \mathbb{E}_{\mathbf{z} \sim p_z} L_\theta(G(\mathbf{z})) \end{aligned} \quad (5.18)$$

where λ is a positive balancing parameter, $(a)_+ = \max(a, 0)$ and θ are the parameters in D . From the second term in \mathcal{L}_D in the equation (5.18), $\Delta(\mathbf{x}, G(\mathbf{z}))$ is added as a regularization term for optimizing D in order to prevent D from overfitting the real samples and the generated samples. Figure 5.16 demonstrates the efficacy of equation (5.18). The loss for D puts a restriction on the ability of D i.e., it challenges the ability of D for to separate well generated samples from real samples, which is the original cause for the vanishing gradient. More formally, LS-GAN assumes that p_r lies in a set of Lipschitz densities with a compact support. The Adam optimizer was used for training the model in this work.

5.6.7 Mode Regularized GAN (MRGAN)

MRGAN proposed a metric regularization to penalize missing modes [278], which is then used to solve the mode collapse problem. The key idea behind this work is the use of an encoder $E(\mathbf{x}): \mathbf{x} \rightarrow \mathbf{z}$ to produce the latent variable \mathbf{z} for G instead of using noise. This procedure has two benefits: (1) The encoder reconstruction can add more information to G so that is not that easy for D to distinguish between generated samples and real samples; and (2) The encoder ensures correspondence between \mathbf{x} and \mathbf{z} ($E(\mathbf{x})$), which means G can cover different modes in the \mathbf{x} space. So it prevents the mode collapse problem. The loss function for this mode regularized GAN is

$$\begin{aligned}\mathcal{L}_G &= -\mathbb{E}_{\mathbf{z}}[\log[D(G(\mathbf{z}))]] + \mathbb{E}_{\mathbf{x} \sim p_r}[\lambda_1 d(\mathbf{x}, G \circ E(\mathbf{x})) + \lambda_2 \log[D(G(\mathbf{x}))]] \\ \mathcal{L}_E &= \mathbb{E}_{\mathbf{x} \sim p_r}[\lambda_1 d(\mathbf{x}, G \circ E(\mathbf{x})) + \lambda_2 \log[D(G(\mathbf{x}))]]\end{aligned}\tag{5.19}$$

where d is a geometric measurement which can be chosen from many options e.g., pixel-wise L^2 and distance of extracted features. The Adam optimizer was used for training the model in this work .

5.6.8 Geometric GAN

Geometric GAN [279] was proposed using SVM separating hyperplane, which has the maximal margins between the two classes. Figure 5.17 demonstrates the update rule for the discriminator and generator based on the SVM hyperplane. Geometric GAN has been successfully demonstrated to be more stable for training and less prone to mode collapse. The RMSprop optimizer was used for training the model in this work .

5.6.9 Relativistic GAN (RGAN)

RGAN [280] was proposed as a general approach to devising new cost functions from the existing one i.e., it can be generalized for all integral probability metric (IPM) [281,282] GANs. The discriminator in the original GAN measures *the probability for a given real sample or a generated sample*. The author argued that key relative discriminant information between real data and generated data is missing in original GAN. The discriminator in RGAN takes into account that *how a given real sample is more realistic than a randomly sampled generated*

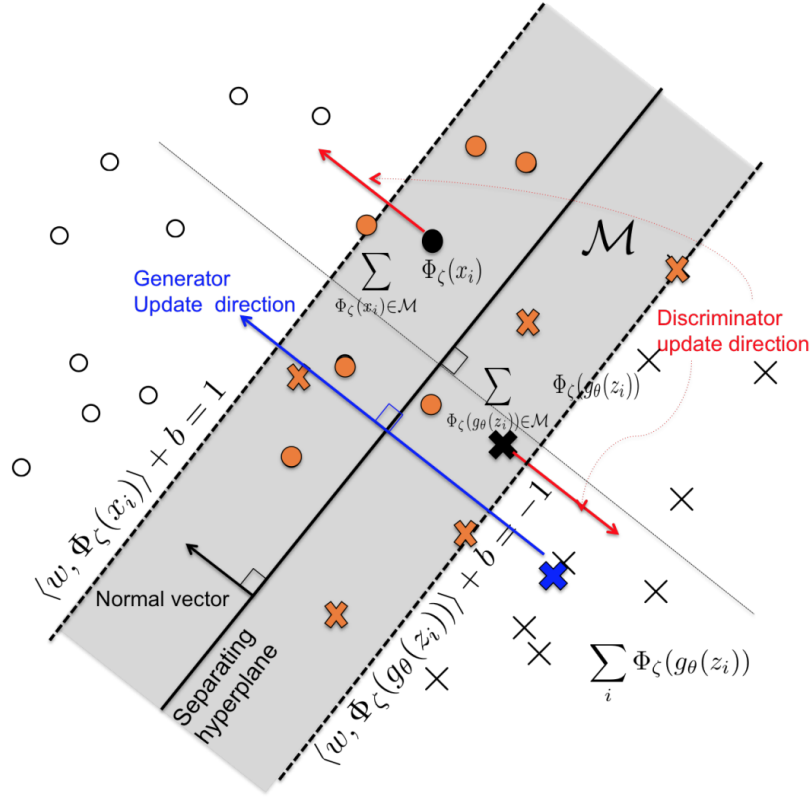


Figure 5.17: SVM hyperplane used in Geometric GAN. Discriminator updates by pushing real data samples and generated data samples away from the hyperplane while generator updates by pushing generated data samples towards the hyperplane. Figure from [279].

sample. Loss function of RGAN applied to original GAN is stated as

$$\begin{aligned} \max_D \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [\log(\text{sigmoid}(C(\mathbf{x}_r) - C(\mathbf{x}_g)))] \\ \max_G \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [\log(\text{sigmoid}(C(\mathbf{x}_g) - C(\mathbf{x}_r)))] \end{aligned} \quad (5.20)$$

where $C(\mathbf{x})$ is the non-transformed layer. Figure 5.18 demonstrates the effect on D of using the RGAN approach compared to the original GAN. In terms of the original GAN, the optimization aims to push the $D(\mathbf{x})$ to 1 (right one). For RGAN, the optimization aims to push $D(\mathbf{x})$ to 0.5 (left one), which is more stable compared to the original GAN. The author also claims that RGAN can be generalized to other types of loss-variant GANs if those loss functions belong to IPMs. The generalization loss is stated as

$$\begin{aligned} \mathcal{L}_D &= \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [f_1(C(x_r) - C(x_g))] + \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [f_2(C(x_g) - C(x_r))] \\ \mathcal{L}_G &= \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [g_1(C(x_r) - C(x_g))] + \mathbb{E}_{\substack{\mathbf{x}_r \sim p_r \\ \mathbf{x}_g \sim p_g}} [g_2(C(x_g) - C(x_r))] \end{aligned} \quad (5.21)$$

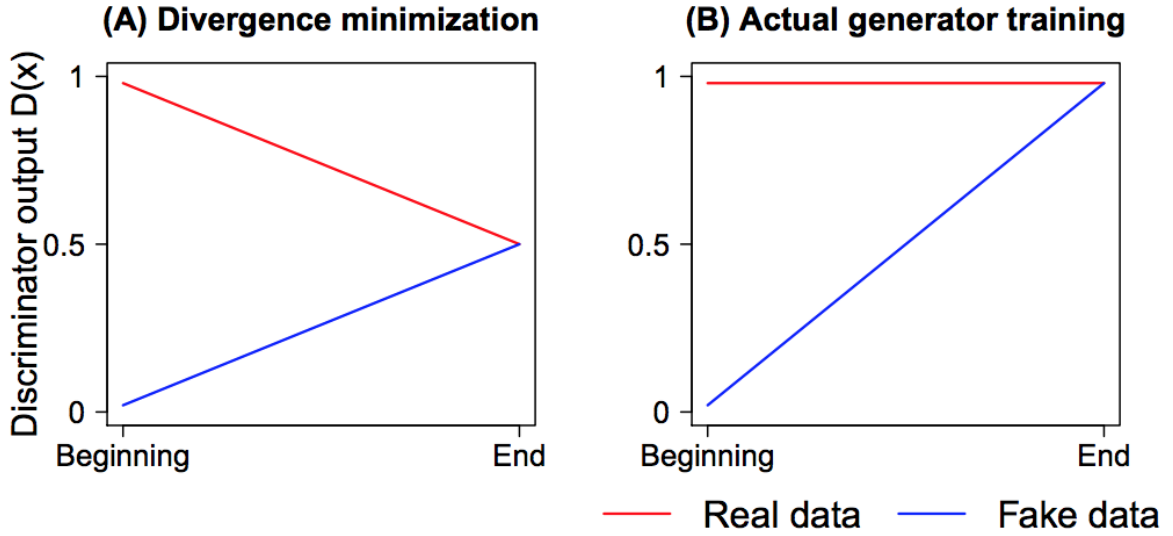


Figure 5.18: D output comparison between RGAN and original GAN. (a) D output in RGAN; (b) D output in original GAN when training the G . Figure from [280].

where $f_1(y) = g_2(y) = -y$ and $f_2(y) = g_1(y) = y$. Details of loss generalization for other GANs refers to the original paper [280]. The Adam optimizer was used for training in this work.

5.6.10 Spectral Normalization GAN (SN-GAN)

SN-GAN [283] proposed the use of weight normalization to stabilize the training of the discriminator. This technique is computationally light and easily applied to existing GANs. Previous work for stabilizing the training of GANs [271, 273, 277] emphasized the importance that D should be from the set of K -Lipshitz continuous functions. Popularly speaking, Lipschitz continuity [284–286] is more strict than the continuity, which describes that the function does not change rapidly (see Appendix C.1 for more details). This smooth D is of benefit in stabilizing the training of GANs. The work mentioned previously focused on the control of the Lipschitz constant of the discriminator function. This work demonstrated an alternative simpler way to control the Lipschitz constant through spectral normalization of each layer for D . Spectral normalization is performed as

$$\bar{\mathbf{W}}_{SN}(\mathbf{W}) = \frac{\mathbf{W}}{\sigma(\mathbf{W})} \quad (5.22)$$

where \mathbf{W} represents weights on each layer for D and $\sigma(\mathbf{W})$ is the L_2 matrix norm of \mathbf{W} (explanation of matrix norm has been presented in Appendix C.2). The paper proved this will

make $\|f\| \leq 1$. The fast approximation for the $\sigma(\mathbf{W})$ was also demonstrated in the original paper. The Adam optimizer was used for training the SN-GAN in this work.

5.6.11 Summary

We have explained the training problems (mode collapse and vanishing gradient for G) in the original GAN and we have introduced loss-variant GANs in the literature, which were mainly proposed for improving the performance of GANs in terms of three key aspects. Figure 5.19 summarizes the efficacy of loss-variant GANs for the challenges. Losses of LSGAN, RGAN

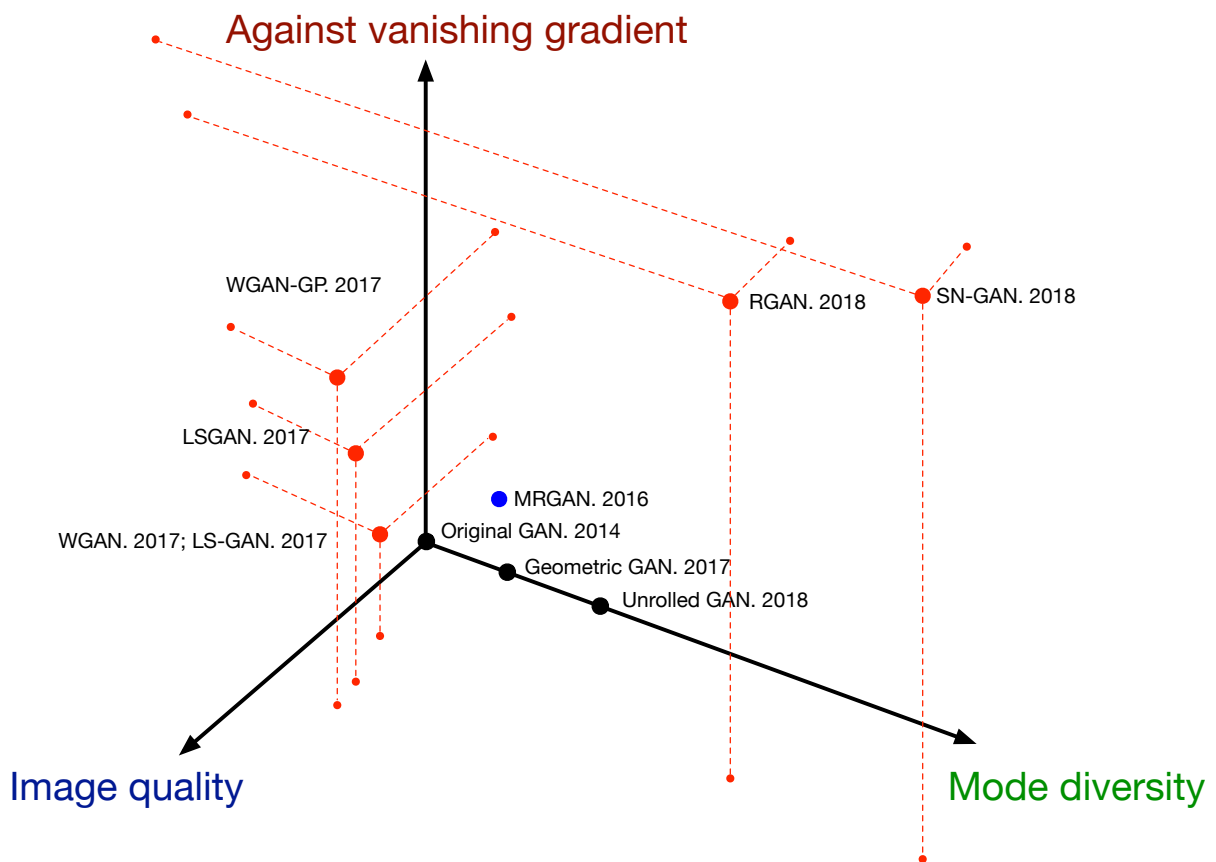


Figure 5.19: Current loss-variants for solving the challenges. Challenges are categorized by three orthogonal axes. A larger value for each axis indicates better performance. Red points indicate the GAN-variant covers all three challenges, blue points cover two, and black points cover only one challenge.

and WGAN are very similar to the original GAN loss. We used a toy example (i.e., the two distributions used in Fig. 5.11) to demonstrate the G loss in terms of the distance between the real data distribution and the generated data distribution in Fig. 5.20. It can be seen that RGAN and WGAN are able to inherently solve the vanishing gradient problems for the generator when

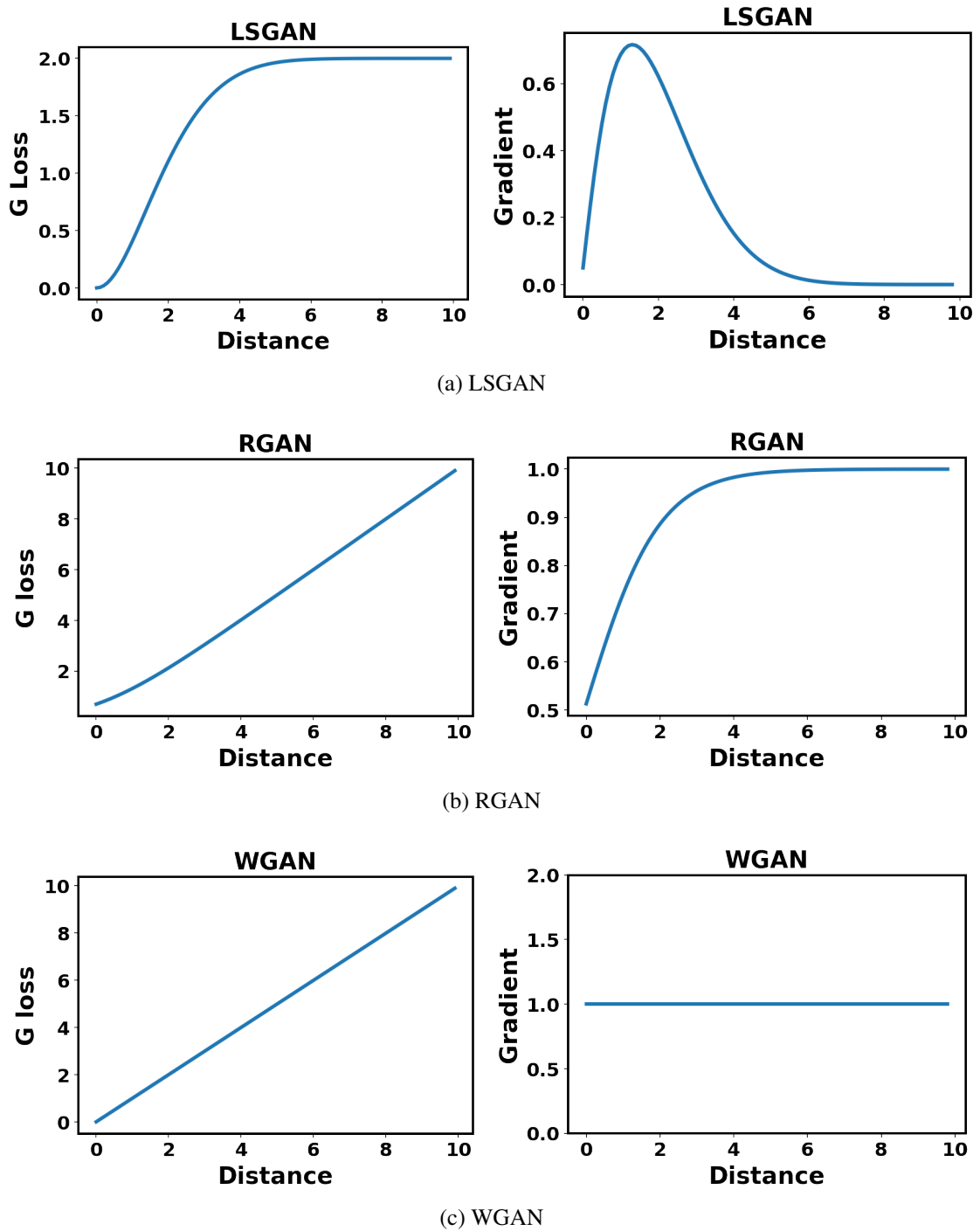


Figure 5.20: Loss and gradient for the generator of different loss-variant GANs.

the discriminator is optimized. LSGAN on the contrary still suffers from a vanishing gradient for the generator, however, it is able to provide a better gradient compared to the original GAN in Fig. 5.12 when the distance between the real data distribution and the generated data distribution is relatively small. This was demonstrated in the original paper [274] where LSGAN is shown to be easier to push generated samples to the boundary made by the discriminator.

Table 5.1 gives details of the properties of each loss-variant GAN. WGAN, LSGAN, LS-

5.6. Loss-variant GANs

Author & Year	GAN type	Pros	Cons
Goodfellow et al. 2014	Original GAN	1. Generate samples very fast. 2. Able to deal with sharp probability distribution.	1. Vanishing gradient for G . 2. Mode collapse. 3. Resolution of generated images is very low.
Chen et al. 2016	MRGAN	1. Improve the mode diversity. 2. Stabilize the training for the GAN.	1. Generated image quality is low. 2. Have not solved vanishing gradient problem for G . 3. Only tested on CelebA dataset. Have not tested on more diverse image datasets e.g. CIFAR or ImageNet.
Nowozin et al. 2016	f-GAN	1. Provide a unified framework based on f-divergence.	1. Have not specified the stability for different f-divergence functions.
Arjovsky et al. 2017	WGAN	1. Solve the vanishing gradient problem. 2. Improve the image quality. 3. Solve the mode collapse.	1. Large weight clipping causes long time to converge and small weight clipping causes vanishing gradient. 2. Weight clipping reduces the capacity of the model and limits the capability to model complex function. 3. Very deep WGAN is hard to converge.
Gulrajani et al. 2017	WGAN-GP	1. Converges much faster than WGAN. 2. Model is more stable during the training. 3. Able to use deeper GAN to model more complex function.	1. Cannot use batch normalization because gradient penalization is done for each sample in the batch.
Mao et al. 2017	LSGAN	1. Improve the image quality. 2. Improve the mode diversity for model. 3. Easy to implement.	1. Generated samples are pushed to decision boundary instead of real data, which may affect the generated image quality.
Qi. 2017	LS-GAN	1. Solve the vanishing gradient problem. 2. Solve the mode collapse problem.	1. Difficult to implement. Lots of tiny pieces have to be carefully designed for loss function. 2. The margin added between real samples and generated samples may affect the quality of generated images.
Lim et al. 2017	Geometric GAN	1. Less mode collapsing. 2. More stable for training. 3. Converges to the Nash equilibrium between the discriminator and generator.	1. Have not demonstrated the ability against the vanishing gradient. 2. Experiment tests have to be done on more complex datasets e.g. ImageNet.
Metz et al. 2018	Unrolled GAN	1. Solve the mode collapse problem. 2. Demonstrate that high order gradient information can help train a GAN. 3. Improve the training stability for GAN.	1. The quality of generated image is low.
Martineau. 2018	RGAN	1. Solve the vanishing gradient problem. 2. Unified framework for IPM-based GANs. 3. Solve the mode collapse problems.	1. Lack of mathematical implications of adding relativism to GANs. 2. Have not done a survey which IPM-based GAN will achieve the best performance with being added the relativism.
Miyato et al. 2018	SN-GAN	1. Computationally light and easy to implement on existing GANs. 2. Improve the image quality and solve the mode collapse. 3. Stabilize the training GANs and solve the vanishing gradient problem.	1. Need to test on more complex image datasets.

Table 5.1: Summary of loss-variant for GANs.

GAN, RGAN and SN-GAN are proposed to overcome the vanishing gradient for G . LSGAN argued that the vanishing gradient is mainly caused by the sigmoid function in the discriminator so it used a least squares loss to optimize the GAN. LSGAN turns out to be the optimization on Pearson χ^2 divergence and remedies the vanishing gradient problem. WGAN used Wasser-

stein (or Earth mover) distance as the loss. Compared to JS divergence, Wasserstein distance is smoother and there is no sudden change with respect to the distance between real samples and generated samples. To be able to use Wasserstein distance as the loss, the discriminator must be Lipschitz continuous, where WGAN deployed the parameter clipping to force discriminator satisfy the Lipschitz continuity. However, it causes problems such as most of parameters in the discriminator locates to the edges of clipping range, which leads to the low capacity of discriminator. WGAN-GP was proposed the use of gradient penalty to make discriminator is Lipschitz continuous, which successfully solves the problems in WGAN. LS-GAN proposed to use a margin that is enforced to separate real samples from generated samples, which restricts the modelling capability of discriminator. It solves the vanishing gradient problem for the generator because this problem arises when the discriminator is optimized. RGAN is a unified framework that is suitable for all IPM-based GANs e.g., WGAN. RGAN added discriminant information to GANs for better learning. SN-GAN proposed an elegant way for optimizing a GAN. As mentioned before, a Lipschitz continuous discriminator is important for stable learning, vanishing gradient and so on. SN-GAN proposed spectral normalization [255] to constrain the discriminator under the Lipschitz continuous requirement. SN-GAN is the first GAN (we do not consider AC-GANs [202] because an ensemble of 100 AC-GANs was used for ImageNet datasets [287].) that has been successfully applied to ImageNet datasets. In theory, spectral normalization as demonstrated in the SN-GAN could be applied to every GAN type. SAGAN and BigGAN [267, 269] both deployed spectral normalization and achieved good results with ImageNet.

Loss-variant GANs are able to be applied to architecture-variants. However, SN-GAN and RGAN show stronger generalization abilities compared to other loss-variants, where these two loss-variants can be deployed by other types of loss-variants. Spectral normalization can be applied to any GAN-variant while the RGAN concept can be applied to any IPM-based GAN. We strongly recommend the use of spectral normalization for all GANs applications as described here. There are a number of loss-variant GANs mentioned in this paper which is able to solve the mode collapse and unstable training problem. Details are given in Table 5.1.

5.7 Discussion

We have introduced the most significant problems present in the original GAN design, which are mode collapse and vanishing gradient for updating G . We have surveyed significant GAN-variants that remedy these problems through two design considerations: (1) Architecture-variants. This aspect focuses on architectural options for GANs. This approach enables GANs to be successfully applied to different applications, however, it is not able to fully solve the problems mentioned above; (2) Loss-variant. We have provided a detail explanation why these problems arise in the original GAN. These problems are essentially caused by the loss function in the original GAN. Thus, modifying this loss function can solve this problem. It should be noted that the loss function may change for some architecture-variants. However, this loss function is changed according to the architecture thus it is architecture-specific loss. It is not able to generalize to other architectures.

Through a comparison of the different architectural approaches surveyed in this work, it is clear that the modification of the GAN architecture has significant impact on the generated images quality and their diversity. Recent research shows that the capacity and performance of GANs are related to the network size and batch size [267], which indicates that a well designed architecture is critical for good GANs performance. However, modifications to the architecture only is not able to eliminate all the inherent training problems for GANs. Redesign of the loss function including regularization and normalization can help yield more stable training for GANs. This work introduced various approaches to the design of the loss function for GANs. Based on the comparison for each loss-variant, we find that spectral normalization as first demonstrated in the SN-GAN brings lots of benefits including ease of implementation, relatively light computational requirements and the ability to work well for almost all GANs. We suggest that researchers, who seek to apply GANs to real-world problems, include spectral normalization to the discriminator.

There is no answer to the question of which GAN is the best. The selection of a specific GAN type depends on the application. For instance, if an application requires the production of natural scenes images (this requires generation of images which are very diverse). DCGAN with spectrum normalization applied, SAGAN and BigGAN can be good choices here. BigGAN is able to produce the most realistic images compared to the other two. However, BigGAN is much more computationally intensive. Thus it depends on the actual computational requirements set by the real-world application.

5.7.1 Interconnections between Architecture and Loss

In this chapter, we highlighted the problems inherent in the original GAN design. In highlighting how subsequent researchers have remedied those problems, we explored architecture-variants and loss-variants in GAN designs separately. However, it should be noted that there are interconnections between these two types of GAN-variants. As mentioned before, loss functions are easily integrated to different architectures. Benefit from improved convergence and stabilization through a redesigned loss function, architecture-variants are able to achieve better performance and accomplish solutions to more difficult problems. For examples, BEGAN and PROGAN use Wasserstein distance instead of JS divergence. SAGAN and BigGAN deploy spectral normalization, where they achieved good performance based on multi-class image generation. These two types of variants equally contribute to the progress of GANs.

5.7.2 Future Directions

GANs were originally proposed to produce plausible synthetic images and have achieved exciting performance in the computer vision area. GANs have been applied to some other fields, (e.g., time series generation [68,211,288] and natural language processing [206,289–291]) with some success. Compared to computer vision, GANs research in other areas is still somewhat limited. The limitation is caused by the different properties inherent in image versus non-image data. For instance, GANs work to produce continuous value data but natural language are based on discrete values like words, characters, bytes, etc., so it is hard to apply GANs for natural language applications. Another limitation of current literature is that the lack of perceptual metrics to evaluate GANs performance. Future research of course is being carried out for applying GANs to other areas and investigates more perceptual metrics to evaluate GANs performance which we are going to cover in the next chapter.

5.8 Conclusion

In this chapter, we have reviewed GAN-variants based on performance improvement offered in terms of higher image quality, more diverse images and more stable training. We reviewed the current state of GAN-related research from an architecture and loss basis. Current state-of-art GANs models such as BigGAN and PROGAN are able to produce high quality images and diverse images in the computer vision field. However, research that applies GANs to video

is limited. Moreover, GAN-related research in other areas such as time series generation and natural language processing lags that for computer vision in terms of performance and capability. We conclude that there are clearly opportunities for future research and application in these fields in particular. With respect to metrics for evaluating GANs performance, we focus on this respect in the next chapter.

Chapter 6

Use of Neural Signals to Evaluate GANs

Abstract: *This chapter demonstrates a novel approach that deploys a cortically coupled computer vision (CCCV) system to generate a biologically neurally-produced metric named **Neuroscore**, which closely mirrors the behavioral ground truth measured from participants tasked with discerning real images from synthetic images produced by generative adversarial networks (GANs). This evaluation process is called a **neuro-AI interface**, as it provides an interface between human neural systems and artificial intelligent (AI) systems. In this chapter, we first compare the three most widely used metrics in the literature for evaluating GANs in terms of visual quality compared to human judgments. Second, we propose and demonstrate a novel approach using neural signals and rapid serial visual presentation (RSVP) that directly measures a human perceptual response to facial production quality, independent of a behavioral response measurement. Finally, we show that our Neuroscore is more consistent with human perceptual judgment compared to the conventional metrics we evaluated. The correlation between our proposed Neuroscore and human perceptual judgments has Pearson correlation statistics: $r(36) = -0.828, p = 4.766 \times 10^{-10}$. We also present the bootstrap result for the correlation i.e., $p \leq 0.0001$. We conclude that neural signals have potential application for high quality, rapid evaluation of GANs in the context of visual image synthesis. This work has been published in the Cognitive Computation [248].*

6.1 Introduction

Artificial intelligence (AI) has significant impact on society yet research into the interaction between humans and AI deserves further exploration and has only recently become a research

focus. Cognitive computation provides a way of using cognitively inspired techniques to solve a variety of real-world problems and these become especially useful when the interface between an AI system and a human is via a brain-computer interface (BCI). Abbass [292] recently explored the last 50 years of the human-AI relationship with a focus on how the development of trust between the parties has been essential. He also covered the emergence of direct BCIs based on EEG (electroencephalography).

As EEG can be the direct reflection of a human mental process, the use of EEG is widely studied and deployed in the cognitive computation literature, for example by [293, 294]. It has been demonstrated recently that EEG can be used effectively for reading emotion [294] and that a spiking neural network framework can be used to analyze a human attention to a task by using EEG [293]. In this chapter, we demonstrate a type of neuro-AI interface derived from cognitive computational perspective (as seen in Fig. 6.1), which uses neural signals, in this case the EEG signals, to score the performance of generative adversarial networks (GANs). The relevance between our work and the existing literature such as [293, 294] is that a processing pipeline has been developed and demonstrated for transforming EEG signals into a value (score or accuracy) and this value matches well a human cognitive response to a specific class of stimulus, in our case an artificially generated facial image. Moreover, our work contains experimental details and provides neuroscientific interpretation in the comparison of our EEG-based technique to existing approaches in the literature.

GANs [65] are attracting increasing interest across many different computer vision applications, for example the generation of plausible synthetic images [256–258, 271], image-to-image translation [224, 228] and simulated image refinement [295]. Despite the extensive work and the many different GAN models reported in the literature, evaluation of the performance of GANs is still challenging. Some comprehensive reviews for GANs evaluation are available including work in [244, 245, 251] and in summary the evaluation for GANs is divided into two main types, *qualitative* and *quantitative*. The most representative *qualitative* metric is to use human annotation to determine the visual quality of the generated images. *Quantitative* metrics compare statistical properties between generated image and real images. Both approaches have strengths and limitations.

Qualitative metrics generally focus on how convincing the image is from a human perceptual perspective rather than detecting overfitting, mode dropping and mode collapsing problems [276]. Human annotation approaches are also time-consuming because they require asking

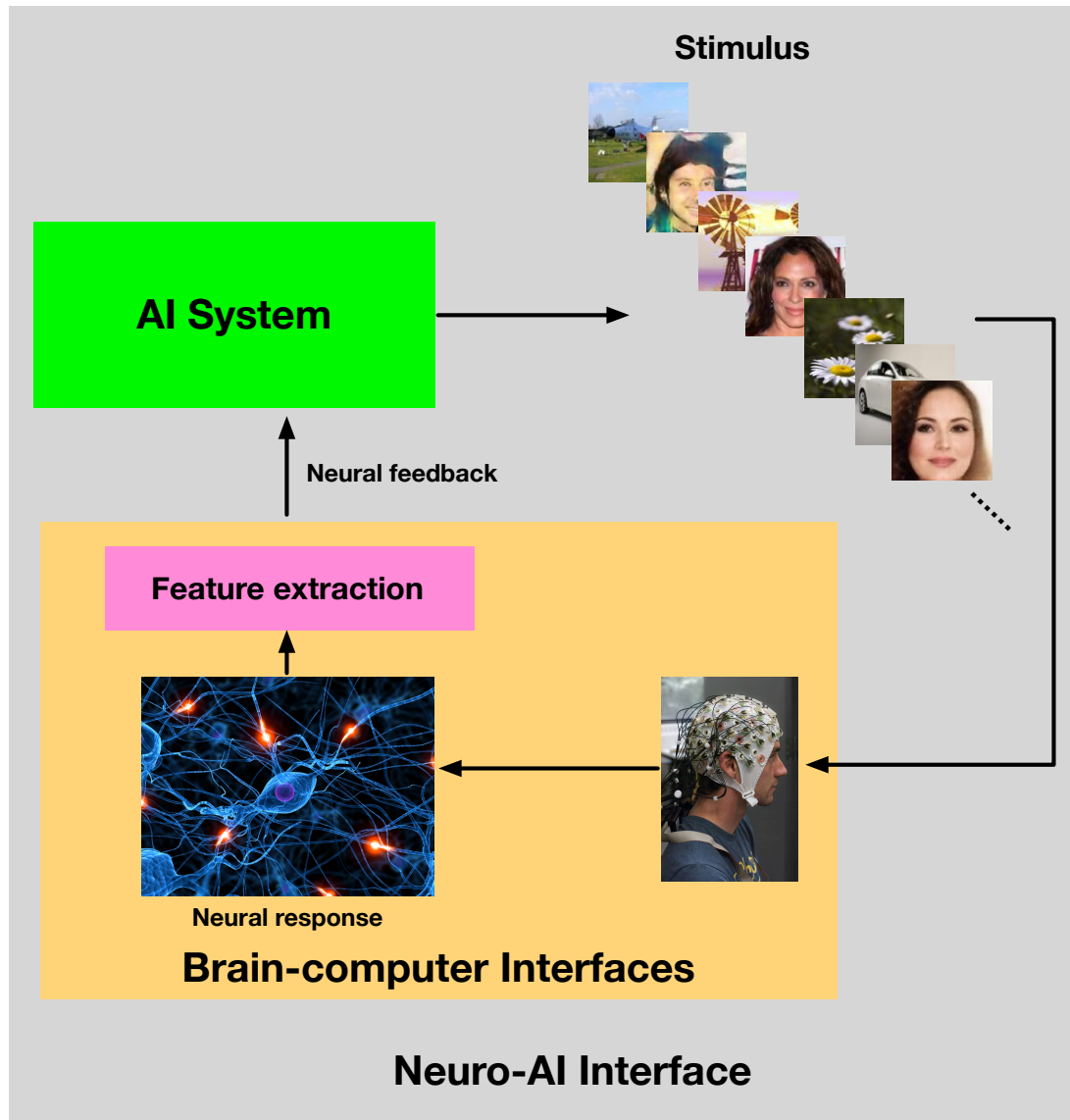


Figure 6.1: Schematic of the neuro-AI interface demonstrated in this study. A type of AI system (e.g., GANs used in this work) produces image stimulus to participants and the corresponding recorded neural response returns to scoring the performance of GANs.

evaluators to generate behavioral responses on an image-by-image basis.

Quantitative metrics in contrast, are less subjective but the psychoperceptual basis of image quality assessment is not well represented in such metrics hence the robustness of their performance is compromised. As a result, the field of research around evaluation methodologies for GANs is still developing and presents opportunities for new approaches. One such approach which we propose, is the introduction of a neuro-AI interface, that uses neural signals for image evaluation in the context of a BCI.

The P300 ERP can suffer from a low signal-to-noise ratio (SNR) and its appearance spans multiple electrodes on the scalp, which make the precise measurement of P300 activity in the

raw, unprocessed EEG epoch difficult. Our previous work in Chapter 3 and Chapter 4 [92, 93] have shown that the P300 can be spatially filtered to improve SNR and reduce dimensionality. The work here will demonstrate a pipeline that uses the LDA beamformer to reconstruct the P300 component for each type of GANs.

Although some work in the GANs evaluation literature has mentioned that quantitative metrics are correlated with human judgment [194, 246], there is no specifically designed work reported in the literature which compares quantitative metrics with those produced by human judgment. It should be noted that the use of human judgment through annotation to evaluate GANs in terms of visual quality is very effective. However, such approaches are very time-consuming and impractical in terms of scale, in real-world applications. Given the advantages of conventional human annotation approaches, we explore the area of BCIs as we know that neural signals can reflect human perception. In this work, we propose a type of neuro-AI interface for evaluating GANs outputs and we deploy an oddball task for eliciting the P300 component via a RSVP protocol, where human participants are rapidly evaluating images produced by GANs. A biologically neuro-produced evaluation metric called Neuroscore is proposed and the calculation of Neuroscore is demonstrated. Results show this neuro-AI interface is more efficient compared to conventional human annotation approaches and Neuroscore is highly correlated with behavioral human judgment. Given this, our work has two primary contributions:

- The design and evaluation of an experiment to compare human assessments with the leading quantitative metrics for GANs performance measurement in terms of image quality.
- The demonstration of a fast and efficient neuro-AI interface in which neural signals provide a superior metric for the evaluation of GANs.

6.2 Related Work

We have introduced the current status of GANs performance in Chapter 5 and have presented the limitation in GAN-related research which is the lack of more perceptual metrics for evaluating GANs performance. This chapter aims to bridge the gap in this aspect. Three well-known metrics are compared with Neuroscore in this chapter, which are Inception Score (IS), Kernel Maximum Mean Discrepancy (MMD) and Fréchet Inception Distance (FID).

Inception Score (IS) is the most widely used metric in the literature [194, 244, 245]. It uses a pre-trained Inception network [296] as an image classification model \mathcal{M} to compute

$$\text{IS} = \exp \left(\mathbb{E}_{\mathbf{x} \sim p_g} [\text{KL} (p_{\mathcal{M}} (y|\mathbf{x}) || p_{\mathcal{M}}(y))] \right) \quad (6.1)$$

Where $p_{\mathcal{M}}(y|\mathbf{x})$ is the label distribution of \mathbf{x} that is predicted by the model \mathcal{M} and $p_{\mathcal{M}}(y)$ is the marginal probability of $p_{\mathcal{M}}(y|\mathbf{x})$ over the probability p_g . A larger inception score will have $p_{\mathcal{M}}(y|\mathbf{x})$ close to a point mass and $p_{\mathcal{M}}(y)$ close to uniform, which indicates that the Inception network is very confident that the image belongs to a particular ImageNet category where all categories are equally represented. This suggests the generative model has both high quality and diversity.

Kernel Maximum Mean Discrepancy (MMD) is a method for comparing two distributions, in which the test statistic is the largest difference in expectations over functions in the unit ball of a reproducing kernel Hilbert space [247]. MMD is computed as

$$\begin{aligned} \text{MMD}^2(p_r, p_g) = \mathbb{E}_{\mathbf{x}_r, \mathbf{x}_r^\top \sim p_r, \mathbf{x}_g, \mathbf{x}_g^\top \sim p_g} \\ [k(\mathbf{x}_r, \mathbf{x}_r^\top) - 2k(\mathbf{x}_r, \mathbf{x}_g) + k(\mathbf{x}_g, \mathbf{x}_g^\top)] \end{aligned} \quad (6.2)$$

where \mathbf{x}_r and \mathbf{x}_g are pixel space or feature space sampled from real images and generated images respectively. It measures the dissimilarity between p_r and p_g for some fixed kernel function k , such as a Gaussian kernel [203]. A lower MMD indicates that p_g is closer to p_r , showing the GAN has better performance.

Fréchet Inception Distance (FID) uses a feature space extracted from a set of generated image samples by a specific layer of the Inception network [246]. The feature space is modelled via a multivariate Gaussian with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. FID is computed as

$$\text{FID}(p_r, p_g) = \|\boldsymbol{\mu}_r - \boldsymbol{\mu}_g\|_2^2 + \text{Tr} (\boldsymbol{\Sigma}_r + \boldsymbol{\Sigma}_g - 2(\boldsymbol{\Sigma}_r \boldsymbol{\Sigma}_g)^{\frac{1}{2}}) \quad (6.3)$$

FID is able to take care of both image quality and image diversity. If the image quality is not high e.g., blurry, it will lead to the increase of the first term. If a GAN falls to the mode collapsing issue, it will lead to the increase of the second term. Similar to MMD, lower FID is better, corresponding to more similar real and generated samples as measured by the distance

between their activation distributions.

In the case of the Inception Score, this is calculated through the Inception model [296]. It has been shown previously that Inception Score is very sensitive to the model parameters [250]. Even scores produced by the same model trained using different libraries (e.g., TensorFlow [297], Keras [298] and PyTorch [299]) differ a lot from each other. Inception Score also requires a large sample size for the accurate estimation of $p_{\mathcal{M}}(y)$. FID and MMD both measure the similarity between training images and generated images based on the feature space [245], since the pixel representations of images do not naturally support the computation of meaningful Euclidean distances [300]. The main concern about the FID and MMD methods is whether the distributional characteristics of the feature space exactly reflect the distribution for the images [300].

6.3 Methodology

6.3.1 P300 Reconstruction

This chapter used the NIFPA EEG dataset which has been introduced in Chapter 2. In Chapter 4, we have shown the efficacy of the LDA beamformer for reconstructing time-course source signal. In this chapter, we applied the LDA beamformer to the EEG epoch in 400 ms – 600 ms time window in order to reconstruct the source P300 signal and we used Algorithm 1 in Chapter 4 to search for the optimal time index for the P300 source signal. We used the LDA beamformer [135] to reconstruct the P300 in this work for the following reasons. Firstly, it is difficult to compare the P300 between participants across a number of channels as the location of the P300 varies across participants. Secondly, the P300 suffers from strong background brain activity so it has a very low signal-to-noise ratio (SNR) [69]. The LDA beamformer method allows us to reconstruct the P300 from a multi-dimensional set of EEG signals i.e., transform 32 channels of EEG to a one-channel time series facilitating within-subject comparisons (with the additional benefit of improving the SNR for the reconstructed P300 as well). A list of related neurophysiologically-relevant terminology and associated explanations used in this work is presented below

- *Single-trial P300 amplitude:* This is the amplitude of the P300 component corresponding to each individual image. The P300 amplitude is calculated by selecting the maximum voltage value between 400 ms and 600 ms for each EEG epoch.

- *Averaged P300 amplitude*: This is the averaged target (for example a face) trial P300 amplitudes.
- *Reconstructed single-trial P300 amplitude*: This is the P300 amplitude corresponding to each single target image. It is the LDA-beamformed single-trial P300 amplitude (the detail of the LDA beamformer method is introduced in Algorithm 2).
- *Reconstructed averaged P300 amplitude*: It is the averaged LDA-beamformed P300 amplitudes corresponding to target trials.

6.3.2 Neuroscore

The *reconstructed averaged P300 amplitude* is used as the basis for a novel metric for evaluating the GANs. Because the latency of the P300 varies across participants, our previous work (Chapter 4) [93] has been demonstrated use of LDA beamformer to search the optimal P300 time index in the RSVP experiment. We picked the maximum value in the 200 ms time window which is centered at the optimal time index to represent the *reconstructed single-trial P300 amplitude* and then average these across the trials to get the *reconstructed averaged P300 amplitude*. This *reconstructed averaged P300 amplitude* is the Neuroscore. In general, our Neuroscore is calculated via two steps: (1) Reconstruct P300 source signal from raw EEG; and (2) Average the P300 amplitude of each reconstructed single-trial source signal across target trials (see Algorithm 2). It should be noted that the **Neuroscore benefits from high SNR** compared to traditional single-trial P300 for the following reasons: (1) The LDA beamformer has been applied to raw EEG epochs in order to maximize the SNR; and (2) The Neuroscore is calculated by averaging trials which is able to mitigate the background EEG noise. Hence, our proposed Neuroscore is a relatively robust metric as defined for this work. It should be noted that higher Neuroscore indicates better GAN performance which is reversed to the traditional scores used in this work.

6.4 Results

6.4.1 Behavioral Task Performance

Figure 6.2 shows examples of four types of face images used in this study i.e., DCGAN, BEGAN, PROGAN and real face (RFACE). We included 12 participants in the behavioral (BE)

6.4. Results

Algorithm 2 Calculation of Neuroscore

Input:

- $\mathbf{X} \in \mathbb{R}^{N \times C \times T}$ is the EEG signal corresponding to the target stimulus, where N is the number of target trials, C is the number of channels, and T is the number of time points.
- $\mathbf{K} \in \mathbb{R}^{M \times C \times T}$ is the EEG signal corresponding to the standard stimulus, M is the number of standard trials, C is the number of channels, T is the number of time points.

Output: Neuroscore

```

1:  $\Sigma = \frac{1}{N} \sum_{i=1}^N \mathbf{X}_i \mathbf{X}_i^\top + \frac{1}{M} \sum_{i=1}^M \mathbf{K}_i \mathbf{K}_i^\top$ 
2: for  $t_i$  in [400 ms, 600 ms] do
3:    $\mathbf{p} = \frac{1}{N} \sum_{i=1}^N \mathbf{X}_{i,t_i} - \frac{1}{M} \sum_{i=1}^M \mathbf{K}_{i,t_i}$ 
4:    $\mathbf{w} = \Sigma^{-1} \mathbf{p} (\mathbf{p}^\top \Sigma^{-1} \mathbf{p})^{-1}$ 
5:    $\mathbf{J}_{t_i} \leftarrow \mathbf{w}^\top \Sigma \mathbf{w}$ 
6:    $\mathbf{W}_{t_i} \leftarrow \mathbf{w}$ 
7: end for
8:  $t_{optimal} = \text{argmin}_{t_i} \mathbf{J}$ 
9:  $\mathbf{w} = \mathbf{W}_{t_{optimal}}$ 
10:  $t_{P300} = [t_{optimal} - 100 \text{ ms}, t_{optimal} + 100 \text{ ms}]$   $\triangleright$  This is time window being detected for the P300.
11: for  $i = 1 : N$  do  $\triangleright$  This is for target trials.
12:    $\mathbf{s} = \mathbf{w}^\top \mathbf{X}_i$ 
13:    $a = \max(\mathbf{s}_{t_{P300}})$ 
14:    $A_i \leftarrow a$ 
15: end for
16: Neuroscore =  $\frac{1}{N} \sum_{i=1}^N A_i$ 

```

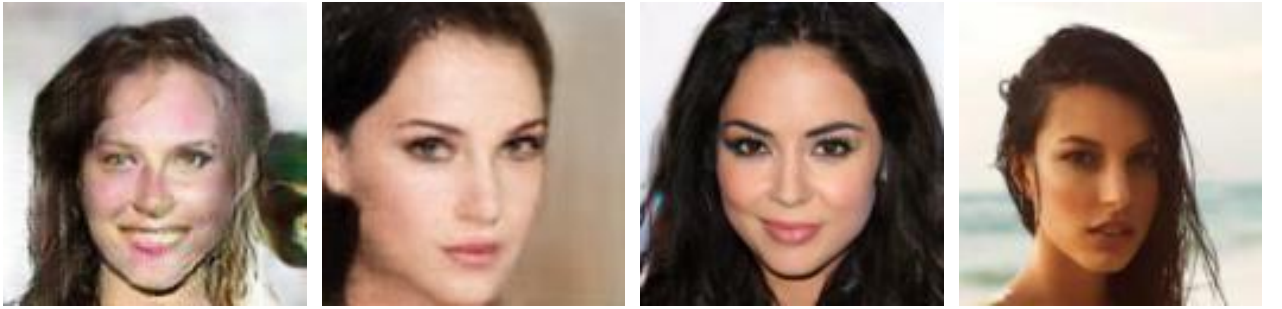


Figure 6.2: Face image examples used in the experiment. From left to right: DCGAN, BEGAN, PROGAN, real face (RFACE). Size of all presented images is 128×128 .

tasks and recorded the accuracy (which is calculated as the number of correctly labelled images divided by the total number of images) of their judgments for each face category. In Table 6.1, it can be seen that participants achieve the lowest accuracy of 0.705 for PROGAN and the highest accuracy 0.994 for DCGAN i.e., participants ranked PROGAN, BEGAN and DCGAN from high performance to low performance respectively. While learning effects may

<i>ID</i>	<i>DCGAN</i>	<i>BEGAN</i>	<i>PROGAN</i>	<i>RFACE</i>
1	1.000	0.759	0.704	0.759
2	0.981	0.741	0.537	0.537
3	1.000	0.796	0.778	0.537
4	0.981	0.889	0.704	0.667
5	1.000	0.667	0.648	0.759
6	1.000	0.926	0.704	0.759
7	1.000	0.815	0.611	0.759
8	0.981	0.815	0.870	0.759
9	1.000	0.796	0.685	0.704
10	1.000	0.815	0.759	0.722
11	1.000	0.907	0.759	0.685
12	1.000	0.963	0.704	0.796
Mean	0.995	0.824	0.705	0.695

Table 6.1: Accuracy (number of correctly labelled images divided by the total number of images) for face images generated from three GANs and real face images in the BE task. Lower accuracy for GAN-generated images indicates better image quality i.e., participants were often convinced that synthesized faces were in fact real.

be present, we find our result is robust regardless of the learning effects as we examined using different groups of RSVP blocks combined with different parts of the BE task, and the results remained consistent. It is interesting that human judgment accuracy for RFACE is 0.695 which is very low. This may be caused by participants being convinced by GAN-generated images and subsequently feeling less confident on the RFACE images, which indicates that GANs are able to convince participants in this case.

6.4.2 Rapid Serial Visual Presentation Task Performance

In order to employ neural signals to evaluate the performance of GANs, we used the RSVP paradigm to elicit the P300 ERP. Figure 6.3 shows the *reconstructed averaged P300 signal* across all participants (using LDA beamformer) in the RSVP experiment. It should be noted here that the *reconstructed averaged P300 signal* is calculated as the difference between averaged target trials and averaged standard trials after applying the LDA beamformer method i.e., $\frac{1}{N} \sum_{i=1}^N \mathbf{w}^\top \mathbf{X}_i - \frac{1}{M} \sum_{i=1}^M \mathbf{w}^\top \mathbf{K}_i$, where \mathbf{w} is the spatial filter calculated by the LDA beamformer, \mathbf{X} and \mathbf{K} are target EEG epochs and standard EEG epochs separately, N and M are the numbers of targets and standards respectively. The solid lines in Fig. 6.3 are the means of the *reconstructed averaged P300 signals* for each image category (across 12 participants) while the shaded areas represent the standard deviations (across participants). It can be seen that the *re-*

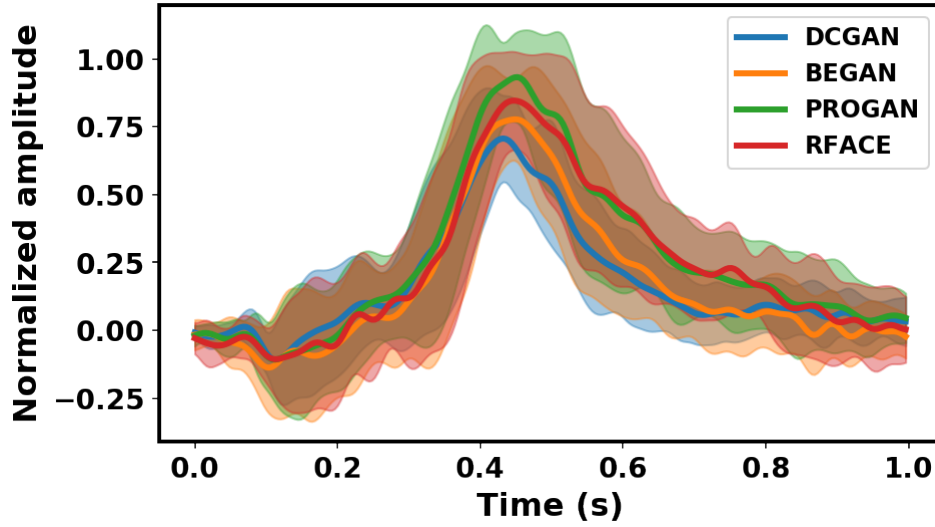


Figure 6.3: Reconstructed averaged (via LDA beamformer) P300 signal across 12 participants in this study. Solid lines are the mean values of the *reconstructed averaged P300 signal* across participants while the shaded areas are the corresponding standard deviations across participants.

constructed averaged P300 signals (across participants) clearly distinguishes between different image categories. Figure 6.4 shows topographical plots of averaged ERP activity (also known as spatial patterns) for the different image categories for each participant. It demonstrates that the spatial topography of P300-related activity varies across participants. It is for this reason that we use the LDA beamformer approach to reconstruct the source P300 for each participant in this study (so as to eliminate erroneous measurement of the P300 by using a specific common channel). We also show a topographic representation of F-values from an ANOVA test that assesses statistical differences between the means of the four categories (one ANOVA for each channel). Larger F-values indicate a larger statistical effect when examining reconstructed P300 values across the four categories for a participant. It can be seen that spatial locations with high F-values are closely aligned to the P300’s spatial topography.

We also show the Neuroscore for each participant (for each GAN) in Table 6.2 for this study. A higher Neuroscore indicates better performance of a GAN. Ranking the performance of GANs by Neuroscore we can see: $\text{PROGAN} > \text{BEGAN} > \text{DCGAN}$, which is consistent with the “human perceptual judgment” in the BE task. Figure 6.5 summarizes the details in Table 6.2. The median values of the Neuroscore for each category across participants give the same rank as the mean value in Table 6.2.

From the averaged subtracted values (on a per-participant basis) of the Neuroscore and BE accuracies (as reflection of human perceptual judgment), it can be seen that the Neuroscore is

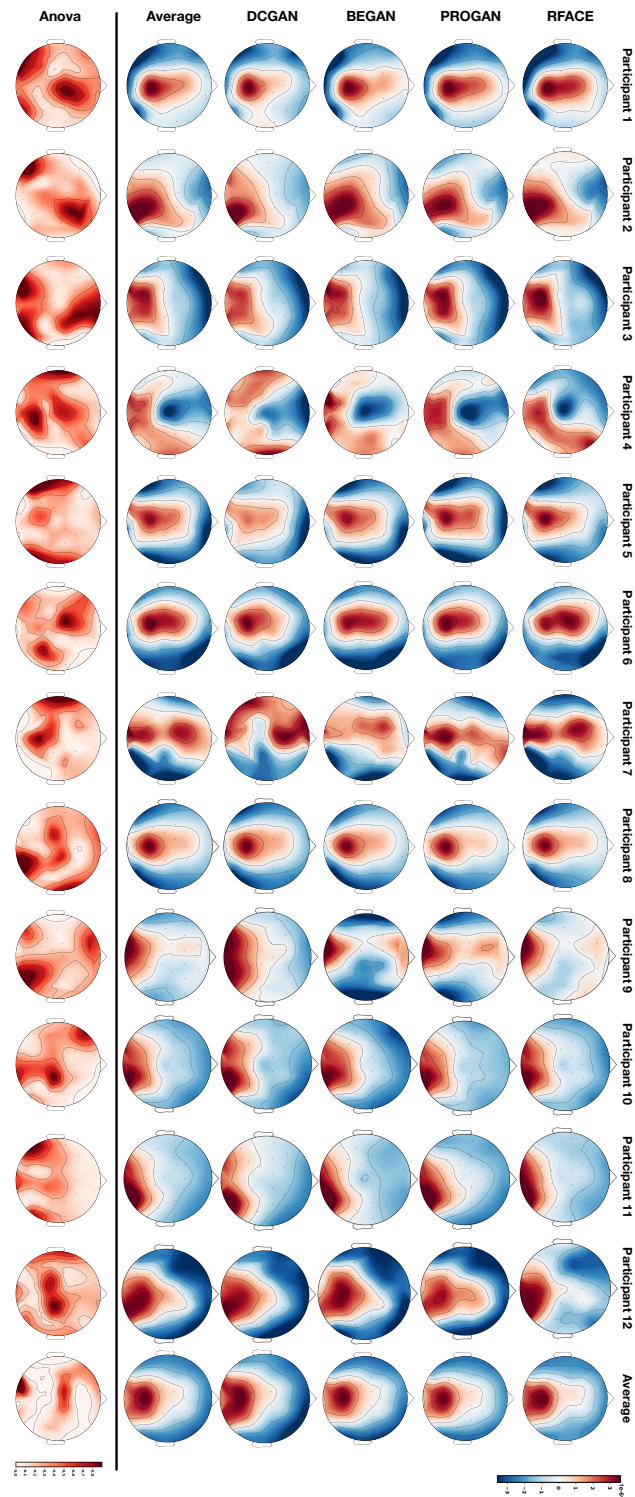


Figure 6.4: Averaged P300 topography of each participant for each category. F-values from an ANOVA test were computed for each channel across four categories. Topography is created at the optimal P300 time index for each participants which is demonstrated in Algorithm 2.

<i>ID</i>	<i>DCGAN</i>	<i>BEGAN</i>	<i>PROGAN</i>	<i>RFACE</i>
1	0.577	0.668	0.685	0.641
2	0.613	0.769	0.939	0.820
3	0.446	0.630	0.689	0.591
4	0.432	0.576	0.974	0.930
5	0.658	0.907	0.938	0.722
6	0.603	0.774	0.964	0.811
7	0.462	0.584	0.856	0.812
8	0.824	0.838	0.882	0.789
9	0.683	0.722	0.911	0.908
10	0.637	0.643	0.962	0.825
11	0.419	0.350	0.425	0.447
12	0.646	0.654	0.819	0.784
Mean	0.583	0.676	0.837	0.757

Table 6.2: Computed Neuroscore of each participant for each category. Higher score indicates better performance of GAN.

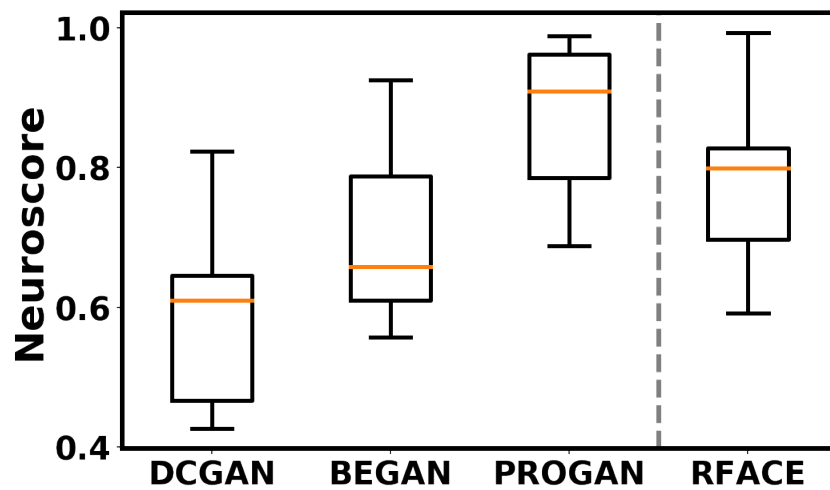


Figure 6.5: Box plot of Neuroscore for each image category across 12 participants.

correlated with the BE accuracy i.e., $\text{PROGAN} > \text{BEGAN} > \text{DCGAN}$ (see Fig. 6.6). In order

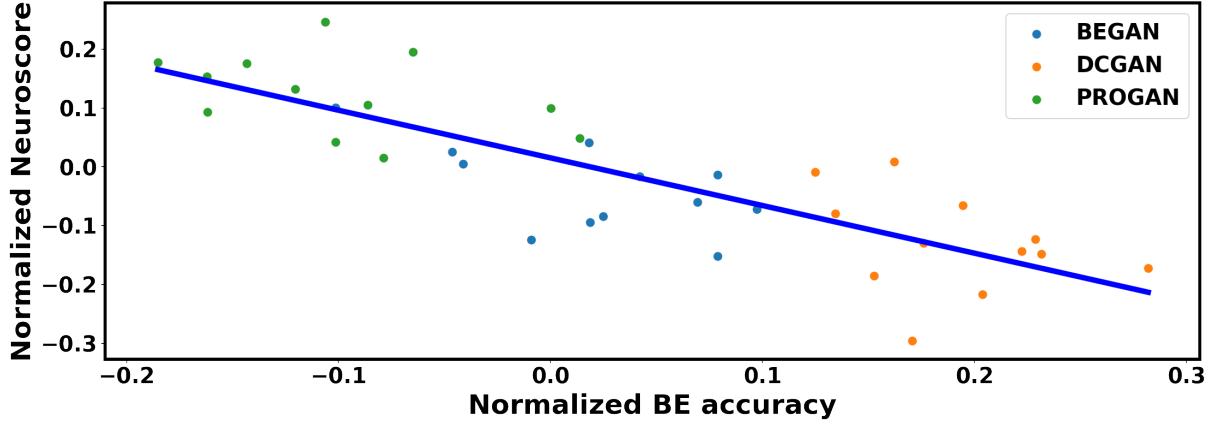


Figure 6.6: Correlation between Neuroscore and BE accuracy (as reflection of human perceptual judgment). Neuroscore and BE are both mean centered within each participant. Using a Pearson correlation coefficient test we find: $r(36) = -0.828, p = 4.766 \times 10^{-10}$.

to statistically measure this correlative relationship, we calculated the Pearson correlation coefficient and p-value (two-tailed) between Neuroscore and BE accuracy ($r(36) = -0.828, p = 4.766 \times 10^{-10}$). Details of results considering including RFACE and with/without normalization are presented as supplementary figures (Fig. D.1, Fig. D.2 and Fig. D.3) in Appendix D.

We used a bootstrap procedure [301, 302] to validate our Pearson correlation coefficient test since aggregating repeated measurements for participants (i.e., treating DCGAN, BEGAN, PROGAN and RFACE measurements as being independent) like this results in a violation of assumptions for our statistical test (violation of independence). Using a bootstrap procedure with our correlation measure allows us to sidestep this violation of assumptions and still obtain a reliable statistic. We did this by repeatedly randomly shuffling the BE accuracy values and Neuroscore (within each participant) and then applying a Pearson correlation coefficient test. After following this process 10,000 times, we counted how many p-values calculated on randomly shuffled values (using within participant shuffling) (i) are smaller than the original p-value (where within-participant shuffling was not applied). $\frac{i}{10000}$ now becomes the bootstrapped Pearson p-value i.e., it estimates the probability of getting the calculated p-value by chance. For the Pearson correlation coefficient test, this strongly supports the interpretation that our Neuroscore is predictive of human judgment. Due to time-based constraints in running the bootstrap procedure, we stopped at 10,000 iterations. This is consistent with our hypothesis that higher Neuroscore indicates better GAN models which is also indicated by lower BE accuracy. The bootstrapped p-value for the Pearson correlation coefficient test is significant ($p \leq 0.0001$),

which means that it is unlikely we have obtained these correlation results by chance¹.

It is notable that PROGAN achieved a higher Neuroscore than RFACE. There are differences between the RFACE and GAN-generated images that are likely impacting the P300 amplitudes for the RFACE images. In the RFACE images, there are a wide range of background textures (e.g., sky, sea and indoor environments) that are not presented in the GAN-generated images. The GAN-generated images tend to have homogeneous backgrounds, where in most cases they are almost monochromatic and/or out of focus. Furthermore, the RFACE images contain a greater variety of other artifacts (e.g., jewellery) that tend not to be discernibly reproduced by the GANs. The lower Neuroscore for RFACE (i.e., RFACE < PROGAN) images is likely a result of these non-task related visual components in the RFACE images increasing the discrimination difficulty. It is known that increasing task difficulty results in a diminished the P300 amplitude [303]. For instance, increasing the amount of visual distractors in an image in a target detection task reduces the P300 amplitude [304]. A further contributing factor may be the stereotyped visual structure of the GAN-generated images (i.e., a face with a bland background), which facilitates the GAN-generated images to be detected more easily in the fast RSVP paradigm used. From the human assessment results in the previous section, it can be seen that participants found the PROGAN output quite convincing, rating faces produced by the GANs similarly in accuracy as the RFACE images.

6.4.3 Comparison to Other Evaluation Metrics

Three traditional methods are employed to evaluate the GANs used in this study. Table 6.3 shows the three traditional scores, Neuroscore and human judgment for three GANs. To be

Methods	DCGAN	BEGAN	PROGAN
1/IS	0.44	0.57	0.42
MMD	0.22	0.29	0.12
FID	63.29	83.38	34.10
1/Neuroscore	1.715	1.479	1.195
Human	0.995	0.824	0.705

Table 6.3: Score comparison for each GAN category. Lower score indicates better performance of GAN.

consistent with other metrics (smaller score indicates better GANs performance), we use 1/Neuroscore for comparison. It can be seen that all three methods are consistent with each other and

¹Without per-participant mean subtraction, the Pearson correlation is $(r(36) = -0.649, p = 1.859 \times 10^{-5})$ and the bootstrapped $p \leq 0.0001$. Details of plot can be seen in Appendix D.

they rank the GANs in order of PROGAN, DCGAN and BEGAN from high performance to low performance. By comparing the three traditional evaluation metrics to the human, it can be seen that they are not consistent with human judgment of GANs performance. It should be noted that Inception Score is able to measure the quality for the generated images [194] while the other two methods cannot do so. However, Inception Score here still rates DCGAN as outperforming BEGAN. Our proposed Neuroscore is consistent with human judgment.

6.5 Discussion

We have compared human assessment with three representative quantitative metrics and used these for comparison with our proposed neural scoring approach. In short, our Neuroscore conveys a measure of the visual quality of facial images generated from GANs. This is based on our hypothesis that a generated image which looks more like a real face image will elicit a larger *reconstructed averaged P300 amplitude* in an RSVP task. Although the other three traditional evaluation methods do provide insight into several aspects of GANs performance, we studied their effectiveness from a visual image quality perspective only as this is the focus of our work. The results are compelling in their demonstration that the proposed Neuroscore is better correlated with human judgment than any of the three quantitative metrics. This is important as an evaluation of the visual quality of a generated image is useful in understanding performance characteristics of specific GANs designs and training datasets. The method proposed can meet this need and is independent of any data modelling assumptions. In contrast, conventional quantitative metrics may fail in this regard.

For example, Inception Score is a model-based evaluation method and the model is very sensitive to adversarial samples as shown in [305]. Inception Score will also produce a very high score if the generated images are produced using adversarial training [250]. Our Neuroscore approach would not be compromised with such images in comparison. It is worth noting that compared with MMD and FID, both Inception Score and our Neuroscore provide a potentially good way of comparing the visual quality between generated images and real images i.e., Inception Score and Neuroscore may give higher score for the generated image that has better visual quality than the real image. Inception Score, however, unlike the neural scoring approach is not able to improve on the ranking of the three GANs compared to MMD or FID.

As mentioned earlier, more realistic GANs will produce a higher Neuroscore. This is be-

cause Neuroscore is sensitive to different stimulus processing requirements for different types of GANs i.e., the larger averaged single-trial P300 amplitudes for GANs reflect properties related to different stimulus information processing requirements [30]. It is also worth commenting that while GANs for generating facial images have been explored in this study, our approach could be used for other types of generated images because the P300 ERP can be elicited using a wide variety of significantly different visual stimuli e.g., Neuroscore may be applicable in the evaluation of GANs in bedroom image generation [252, 256, 258, 274].

The work presented here focuses on evaluating image visual quality only. Consequently there are some limitations when using the Neuroscore to evaluate GANs in this way. Overfitting, mode dropping and mode collapsing are very important aspects of GANs performance and most quantitative methods are able to assess these in some way. However for these broader assessments, we can augment quantitative methods with our Neuroscore to gain a better assessment of overall GANs performance. In reality, choosing the appropriate evaluation metric for GANs depends on the application and which type of problem is being addressed by the GAN. If the goal of a GAN application is the generation of high visual quality images, e.g., super resolution image reconstruction, a qualitative metric is preferred in that case. If a GAN is to be trained to capture the categories of large image datasets, a quantitative metric would be a better choice. Therefore the inclusion of a neural scoring approach as we have demonstrated should be considered in the context of the application’s requirements.

Neuroscore is produced from human EEG signals and directly reflects human perception and neural processes. Compared to human judgment on images generated from GANs, our paradigm has several advantages as follows. Firstly, it is much faster than human judgment as a rapid image stream is presented to participants as part of the RSVP protocol. Traditional human judgment approaches entails the evaluation of images one-by-one whereas our paradigm supports batch evaluation of images. Secondly, as the EEG recorded corresponds to individual images, the method allows the tracking of image quality at the level of the individual image rather than the aggregated quality of a group of images. Thirdly, Neuroscore produces a continuous value while human judgment is binary (“real” or “fake”). Finally, it is possible to use EEG signals such as the P300 as supervisory information for improving training of GANs in the future.

In this work, we focus on the evaluation of images generated from GANs. However, time series evaluation of GANs is even more challenging and even less discussed in the literature.

We believe that our paradigm may be extended and applied to auditory BCIs [306] for auditory evaluation for GANs in the future.

The work discussed in this chapter addresses **research question 2**: *Can neural signals be used to provide indications on image quality that is consistent with human perceptual judgment and is it possible to use this as a biological score to evaluate generative models such as GANs?*

We demonstrated the feasibility of using neural responses to evaluate the image quality and showed it is consistent with human perceptual judgment. We then proposed a neurally-produced score as an evaluation metric for assessing the quality of images produced by GANs.

6.6 Conclusion

We have conducted a comprehensive comparison between human assessments and three quantitative metrics for the comparison of image quality in the specific GANs application of facial imagery synthesis. We proposed and assessed a neural interfacing approach in which a Neuroscore is introduced as an alternative evaluation of GANs in terms of image visual quality. We interpreted our results to conclude that our Neuroscore is more consistent with assessments made by humans when compared to the three quantitative metrics and we showed that the correlation between our Neuroscore and human judgment is not produced by chance i.e., $p \leq 0.0001$. We believe that our proposed paradigm based on a rapid serial visual presentation approach is more efficient and less prone to error compared to conventional human annotation. Consequently we suggest that approaches using such neural signals may complement or for some specific applications, replace, conventional metrics for evaluation of GANs performance.

Chapter 7

Pseudo Neuroscore: Using A Neuro-AI Interface for Evaluating GANs

Abstract: *We have introduced the Neuroscore for evaluating generative adversarial networks (GANs) in Chapter 6. The calculation of this metric presently requires the cumbersome measurement of a participant’s neural signals in response to GAN generated images. For generalizing the use of Neuroscore, a convolutional neural network (CNN) based neuro-AI interface is proposed to predict the Neuroscore from GAN-generated images in this chapter, which is able to evaluate the performance of GANs without directly using electroencephalography (EEG) data on the testing image presentation. Importantly, we show that including neural responses during the training phase of the network can significantly improve the prediction capability of the proposed framework and the proposed framework is able to predict the Neuroscore using image data alone. Our results demonstrate this type of neuro-AI interface has superior performances to the current evaluation metrics in that: (1) It is more consistent with human judgment; (2) The evaluation process needs much smaller numbers of samples; and (3) It is able to rank the quality of images on a per GAN basis.*

7.1 Introduction

Although some evaluation metrics, e.g., Inception Score (IS), Kernel Maximum Mean Discrepancy (MMD), Fréchet Inception Distance (FID), have already been proposed [194, 244, 246], their limitations are obvious: (1) These metrics do not agree with human perceptual judgments and human rankings of GANs models, which have been demonstrated in Chapter 6. A small

artifact on images can have a large effect on the decision made by a machine learning system [307], whilst the intrinsic image content does not change. In this respect, we consider human perception to be more robust to adversarial images samples when compared to a machine learning system; (2) These metrics require large sample sizes for evaluation [194, 245]. Large-scale samples for evaluation sometimes are not realistic in real-world applications since it is time-consuming; and (3) They are not able to rank individual GAN-generated images by their quality i.e., the metrics are generated on a collection of images rather than on a single image basis. The within GAN variances are crucial because they can provide the insight on the variability of that GAN.

Neuroscore proposed in the previous chapter has demonstrated consistency with human perceptual judgment. It is able to score a GAN regarding the image quality with small sample size requirement. The main limitation of previous work is that Neuroscore relies on EEG signals recorded from a human for evaluating each single image, which is impractical for real-world applications. Researchers from the computational neuroscience area are keen to explore the relationships and interconnections between DNNs and biological neural systems. Research has demonstrated that it is able to use CNNs to model sensory cortical processing [90, 308]. Inspired by the work coming from the field of computational neuroscience, we explore the use of deep learning approaches for modelling the Neuroscore in order to generalize the use of Neuroscore.

In this work, we propose a CNN based neuro-AI interface framework that is able to produce Neuroscore to evaluate the performance of GANs, which is trained by using neuropsychological responses recorded via non-invasive electroencephalography (EEG). Furthermore, we validate this framework where it calculates the Neuroscore for images without corresponding neural responses. We test this framework via three models: A Shallow convolutional neural network, Mobilenet V2 [309] and Inception V3 [296].

The unique benefit of Neuroscore is that it more directly reflects human perceptual judgment of images, which is intuitively more reliable compared to the conventional metrics in the literature [244]. We list the supported features of Neuroscore and traditional metrics in Table 7.1. Neuroscore can not only evaluate image quality as other metrics, but also have 3 unique characteristics, which will be demonstrated in section 7.4.

In summary, our primary contribution in this chapter is that we propose a neuro-AI interface and training strategy to generalize the use of Neuroscore, which can be directly used for GAN evaluations without requiring EEG. This enables our Neuroscore to be more widely applied to

Feature	IS	MMD	FID	Neuroscore
Evaluate image quality	✓	×	✓	✓
Consistent with human	×	×	×	✓
Small sample size	×	×	×	✓
Rank images	×	×	×	✓

Table 7.1: Comparison between Neuroscore and other metrics.

real-world scenarios.

7.2 Related Work

The current literature demonstrated that a CNN is able to predict neural responses in the inferior temporal (IT) cortex during an image recognition task [90, 308] using invasive BCI techniques [310]. The investigation of using DNNs to predict neural responses from a non-invasive BCI aspect is still an open question. Figure 7.1 illustrates a schematic of different neural dy-

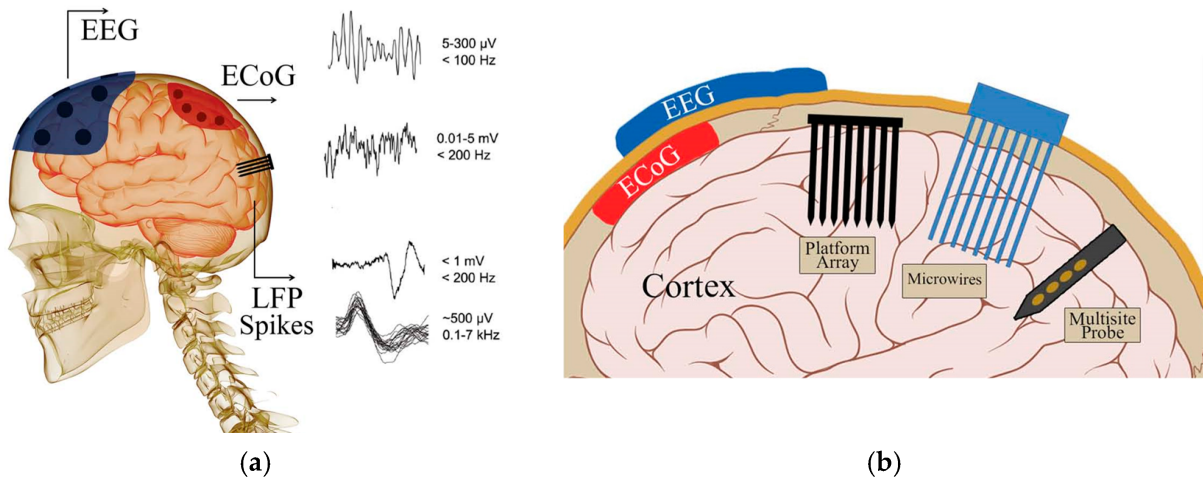


Figure 7.1: Schematic of different types of recorded neural signals (illustrated in (a)) via invasive and non-invasive measurements (illustrated in (b)). Figure from [311].

namics measured from invasive and non-invasive approaches. In this schematic, only the EEG (electroencephalogram) is non-invasively measured from human scalp. Other types of neural dynamics such as electrocorticogram (ECoG) [312] and local field potential (LFP) [313] are measured invasively, which requires electrodes implanted in encephalic. Compared to invasively measured neural dynamics, the advantage of EEG is that it is easy to perform, comfortable and more easily generalized to real-world applications. However, EEG also suffers challenges such as low signal quality (i.e., low SNR), low spatial resolution (interesting neural

activities span all of the scalp and are thus difficult to localize), all of which makes predicting EEG responses challenging.

With the success achieved by deep neural networks (DNNs) in areas including computer vision and natural language processing, the operation and functionality of DNNs and their connection with the human brain has been extensively studied and investigated in the literature [308, 314–321]. In this research area, CNNs are widely studied and compared with the visual system in the human brain because both are hierarchical systems and the processing steps are similar. For example in an object recognition task, both CNNs and humans recognize an object based on its shape, edges, color, etc. and then pass that information on to higher levels of processing. Work reported in [308] outlined a CNN approach to delving even more deeply into understanding the development and organization of sensory cortical processing. It has recently been demonstrated that a CNN is able to reflect the spatio-temporal neural dynamics in the human brain visual processing area [314, 317, 318]. Despite much work carried out to reveal the similarity between CNNs and brain systems, research on interactions between CNNs and neural dynamics is limited.

In [90] the authors demonstrated that a CNN matched with neural data recorded from the IT cortex of a monkey [322] has high performance in an object recognition task. Given the evidence above that a CNN is able to predict neural responses in the brain, we are exploring the use of CNNs to predict the Neuroscore in this chapter. This type of model can then produce neural feedback for different types of GANs.

With advanced machine learning technologies applied to non-invasive BCI area, source localization and reconstruction are feasible for EEG signals. Our previous work (Chapter 4 and Chapter 6) [92, 93] demonstrated the efficacy of using spatial filtering approaches for reconstructing P300 source ERP signals. The low SNR issue can be remedied by averaging EEG trials. Based on this evidence, we explore the use of DNNs to predict a neurally-produced metric Neuroscore introduced in Chapter 6, when neural information is available.

In this work, we demonstrate and validate a neural-AI interface (as seen in Fig. 7.2), which uses neural responses as supervisory information to train a CNN. The trained CNN model is able to predict Neuroscore for images without requiring corresponding neural responses. We test this framework via three models: Shallow convolutional neural network, Mobilenet V2 [309] and Inception V3 [296].

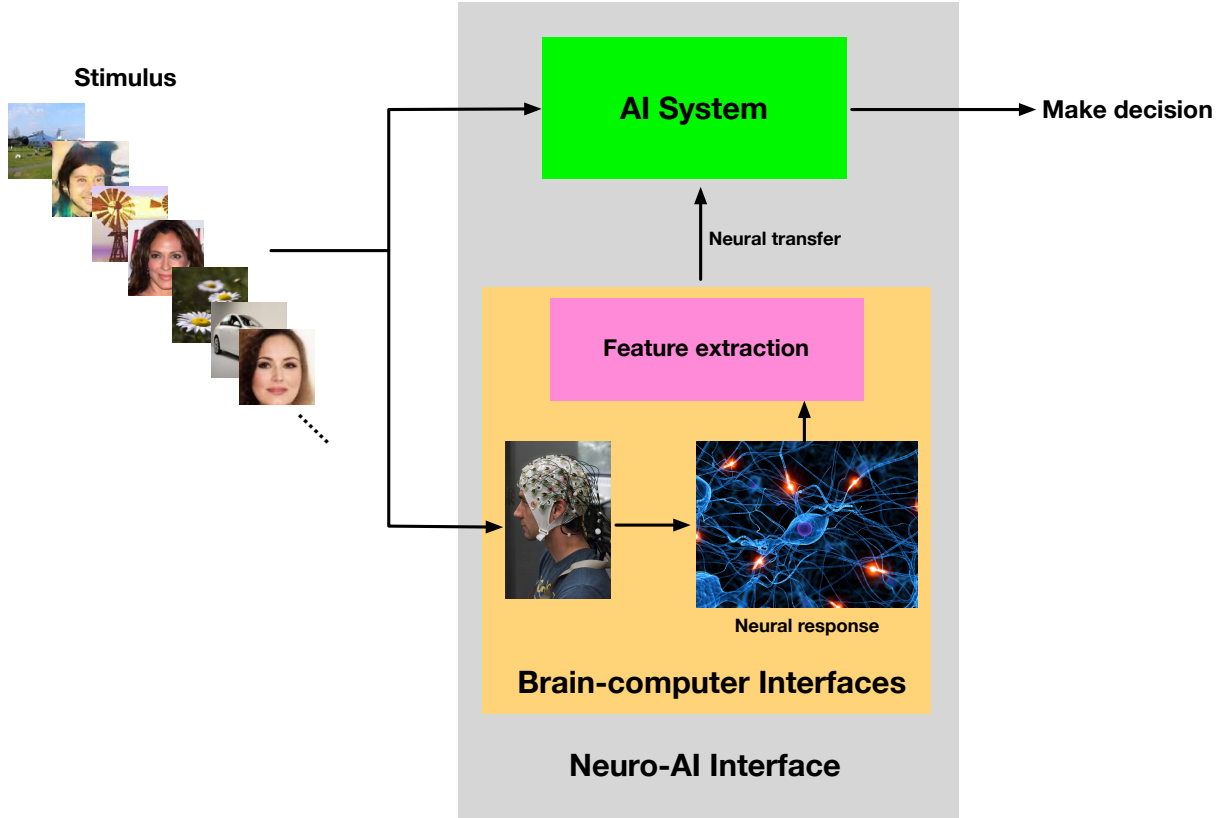


Figure 7.2: Schematic of the neuro-AI interface described in this chapter. Stimuli (image stimuli used in this work) are simultaneously presented to an AI system and to participants. Participants’ neural responses are transferred to the AI system as supervisory information for assisting the AI system to make decisions.

7.3 Methodology

7.3.1 Neuro-AI Interface

We propose a type of neuro-AI interface in order to generalize the use of Neuroscore. This kind of framework is used for predicting Neuroscore corresponding to images generated by one of the popular GANs models. Figure 7.3 demonstrates the CNNs based neuro-AI interface used in this work¹. Flow 1 shows that the image processed by the human brain and produces single-trial P300 source signal for each input image. Flow 2 in Fig. 7.3 demonstrates a CNNs based neuro-AI interface. The convolutional and pooling layers process the image similarly as how the retina does [85]. Fully-connected layers (FC) 1 – 3 aim to emulate the brain’s functionality that produces EEG signal. Yellow dense layer in the architecture aims to predict the single-trial P300 source signal in 400 ms – 600 ms time window response from each image input. In

¹We understand that the human brain is much more complex than what we demonstrated in this work and the flow in the brain is not one-directional [89,323]. Our framework can be further extended to be more biologically plausible.

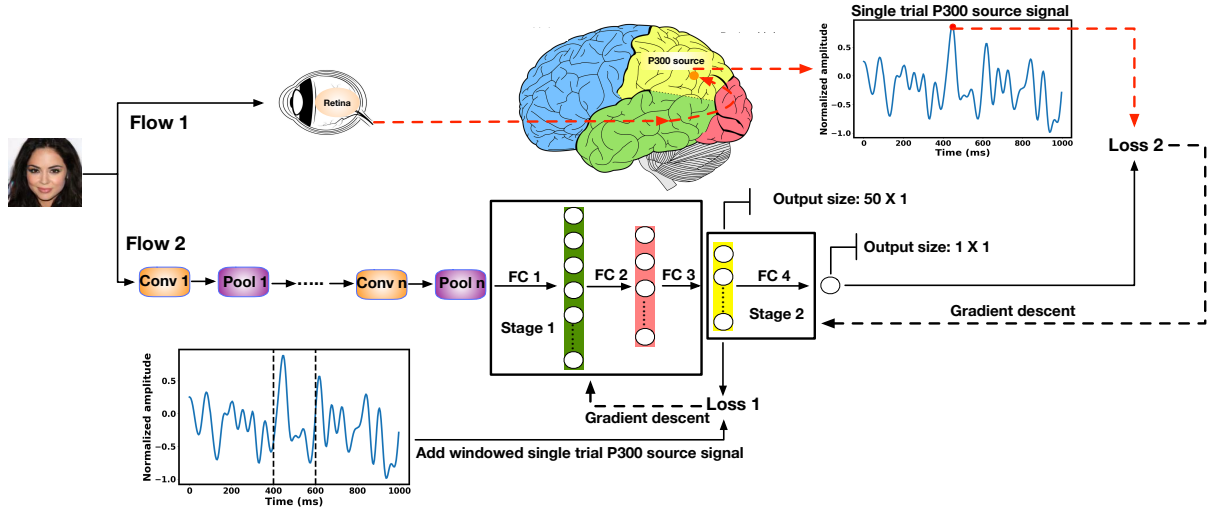


Figure 7.3: A neuro-AI interface and training details with adding EEG information. Our training strategy includes two stages: (1) Learning from image to P300 source signal; (2) Learning from P300 source signal to P300 amplitude. $loss_1$ is the L_2 distance between the yellow layer and the single-trial P300 source signal in the 400 ms – 600 ms time window corresponding to the single input image. $loss_2$ is the mean square error between model prediction and the *single-trial P300 amplitude*². $loss_1$ and $loss_2$ will be introduced in section 7.3.2.

order to help the model make a more accurate prediction for the *single-trial P300 amplitude*² for the output, the single-trial P300 source signal in 400 ms – 600 ms time window was fed to the yellow dense layer to learn parameters for the previous layers in the training step. In this step, the *reconstructed single-trial P300 signals* (i.e., as we discussed in the Chapter 6, the *reconstructed single-trial P300 signals* are single-channel signals which are reconstructed by using LDA beamformer) located in 400 ms – 600 ms were used. In detail, 50 temporal points will be derived in this case because our sampling rate is 250 Hz i.e., $0.2 \times 250 = 50$. Therefore, we used 50 neurons in the yellow dense layer in order to make it consistent with the *windowed reconstructed single-trial P300 signals*. The model was then trained to predict the single-trial P300 source amplitude (red point shown in signal-trail P300 source signal of Fig. 7.3). More details can refer to the next section.

7.3.2 Training Details

Mobilenet V2, Inception V3 and Shallow network were explored in this work, where in Flow 2 we used these three network backbones: such as Conv1-pooling layers. For Mobilenet V2 and Inception V3. We used pretrained parameters from up to the FC 1 shown in Fig. 7.3. We

²As we explained in section 6.3.1 in Chapter 6, single-trial P300 amplitude refers the maximum value in the 400 ms – 600 ms time window of a single-trial EEG signal.

trained parameters from FC 1 to FC 4 for Mobilenet V2 and Inception V3. θ_1 is used to denote the parameters from FC 1 to FC 3 and θ_2 indicates the parameters in FC 4. For the Shallow network, we trained all parameters from scratch. Figure 7.4 shows the architectural detail of the

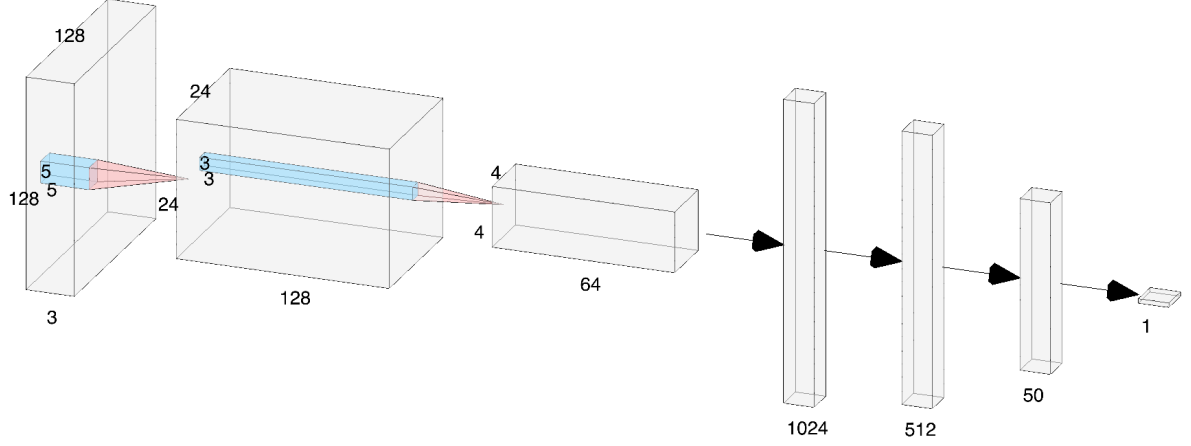


Figure 7.4: Architecture of Shallow network used in this work.

Shallow CNN used in this work.

We added EEG to the model because we first want to find a function $f : \chi \rightarrow s$ that maps the image space χ to the corresponding single-trial P300 source signal. This prior knowledge can help us to predict the single-trial P300 amplitude in the second learning stage.

We compared the performance of the models with and without EEG for training. We defined a two-stage loss function (loss_1 for single-trial P300 source signal in the 400 ms – 600 ms time window and loss_2 for single-trial P300 amplitude) as

$$\begin{aligned} \text{loss}_1(\theta_1) &= \frac{1}{m} \sum_{i=1}^m \|\mathbf{S}_i^{\text{true}} - \mathbf{S}_i^{\text{pred}}(\theta_1)\|_2^2 \\ \text{loss}_2(\theta_1, \theta_2) &= \frac{1}{m} \sum_{i=1}^m (y_i^{\text{true}} - y_i^{\text{pred}}(\theta_1, \theta_2))^2 \end{aligned} \tag{7.1}$$

where $\mathbf{S}_i^{\text{true}} \in \mathbb{R}^{1 \times T}$ is the single-trial P300 signal in the 400 ms – 600 ms time window to the presented image (T denotes the number of time points in the 400 ms – 600 ms time window), y_i refers to the single-trial P300 amplitude to each image and m is the batch size. In this case, we trained 20 epochs with batch size equaling to 256. An Adam optimizer with default hyperparameters was used and learning rate is 0.001. Details of two-stage training results can be referred to Fig. D.4 in Appendix D.

The training of the models without using EEG is straightforward, models were trained di-

rectly to minimize $\text{loss}_2(\theta_1, \theta_2)$ by feeding images and the corresponding single-trial P300 amplitude. Training with EEG information is explained in Algorithm 3 and visualized in the

Algorithm 3 Two-stage training with EEG information.

Stage 1: Training parameters θ_1 .

Input: Images \mathbf{x} and single-trial P300 source signals \mathbf{S}^{true} .

- 1: **for** number of training iterations **do**
- 2: Sample minibatch of m image samples $\{\mathbf{x}_1, \dots, \mathbf{x}_m\}$ and m single-trial P300 source signal samples $\{\mathbf{S}_1^{true}, \dots, \mathbf{S}_m^{true}\}$.
- 3: Update θ_1 by descending its stochastic gradient: $\nabla_{\theta_1} \frac{1}{m} \sum_{i=1}^m \|\mathbf{S}_i^{true}(\mathbf{x}_i) - \mathbf{S}_i^{pred}(\theta_1, \mathbf{x}_i)\|_2^2$.
- 4: **end for**

Stage 2: Freezing θ_1 , training parameters θ_2 .

Input: Images \mathbf{x} and single-trial source P300 amplitudes \mathbf{y}^{true} .

- 5: **for** number of training iterations **do**
 - 6: Sample minibatch of m image samples $\{\mathbf{x}_1, \dots, \mathbf{x}_m\}$ and m single-trial P300 source amplitude samples $\{y_1^{true}, \dots, y_m^{true}\}$.
 - 7: Update θ_2 by descending its stochastic gradient: $\nabla_{\theta_2} \frac{1}{m} \sum_{i=1}^m (y_i^{true}(\mathbf{x}_i) - y_i^{pred}(\theta_1, \theta_2, \mathbf{x}_i))^2$.
 - 8: **end for**
-

“Flow 2” of Fig. 7.3 with two stages. Stage 1 learns parameters θ_1 to predict P300 source signal while stage 2 learns parameters θ_2 to predict single-trial P300 amplitude with θ_1 fixed.

Table 7.2 shows the number of EEG trials in the NIFPA dataset after eye-related artifact

<i>ID</i>	<i>DCGAN</i>	<i>BEGAN</i>	<i>PROGAN</i>	<i>RFACE</i>
1	116	108	107	113
2	100	106	110	98
3	156	153	154	154
4	144	153	143	144
5	110	101	92	80
6	135	131	122	106
7	138	139	143	141
8	151	151	150	151
9	146	149	140	149
10	104	87	93	82
11	149	138	144	142
12	97	92	99	101

Table 7.2: Number of trials for each stimulus type remaining after artifact rejection across each participant (ID) and different GAN categories.

rejection. There are 6012 trials (including RFACE) and 4551 trials (excluding RFACE) for all participants. Training and testing were splitted into 70% and 30% in this study.

7.4 Results

7.4.1 EEG Improves Model Performance

Individual Participant Performance. Each model was trained by using each individual participant’s EEG data in this part. Three models have been validated for each individual participant as shown in Fig. 7.5. It can be seen that all three models trained with EEG outperform the models trained without EEG, exhibiting smaller error and variances across almost all individual subjects. For those cases where the reverse is true (7 from 36 have better or equal performance without EEG), this may result from the number of EEG trials for an individual participant not being sufficient enough for training of deep networks to learn the prediction for the Neuroscore.

Cross-participants Performance. In this case, each model was trained via using all participants’ EEG data together. Table 7.3 shows the error for each model with EEG signals, with randomized EEG signals **within each type of GAN**³ and without EEG signals. All models with EEG perform better than models without EEG, with much smaller errors and variances.

	Model	Error mean(std)
Shallow net	Shallow-EEG	0.209 (± 0.102)
	Shallow-EEG _{random}	0.348 (± 0.114)
	Shallow	0.360 (± 0.183)
Mobilenet	Mobilenet-EEG	0.198 (± 0.087)
	Mobilenet-EEG _{random}	0.404 (± 0.162)
	Mobilenet	0.366 (± 0.261)
Inception	Inception-EEG	0.173 (± 0.069)
	Inception-EEG _{random}	0.392 (± 0.057)
	Inception	0.344 (± 0.149)

Table 7.3: Testing errors of 9 models for cross participants (“-EEG” indicates models are trained with paired EEG, “-EEG_{random}” refers to EEG trials which are randomized in the loss_1 **within each type of GAN**). Results are averaged by shuffling training/testing sets for 20 times. One-way ANOVA test has been done between “model with EEG” and “model without EEG”: Shallow net [$F(1, 38) = 9.77, p = 0.003$], Mobilenet [$F(1, 38) = 7.05, p = 0.012$] and Inception [$F(1, 38) = 20.37, p = 5.98 \times 10^{-5}$].

Adding EEG information reduces error in all three models (as the same error shown in Fig. 7.5), which are 0.151, 0.168 and **0.171** for Shallow, Mobilenet, and Inception respectively. This indicates that the Inception model benefits the most when adding EEG information in the training stage. The performance of models with EEG is ranked as follows: Inception-EEG,

³We randomized the pairing between images and their corresponding single-trial EEG signals (within GANs).

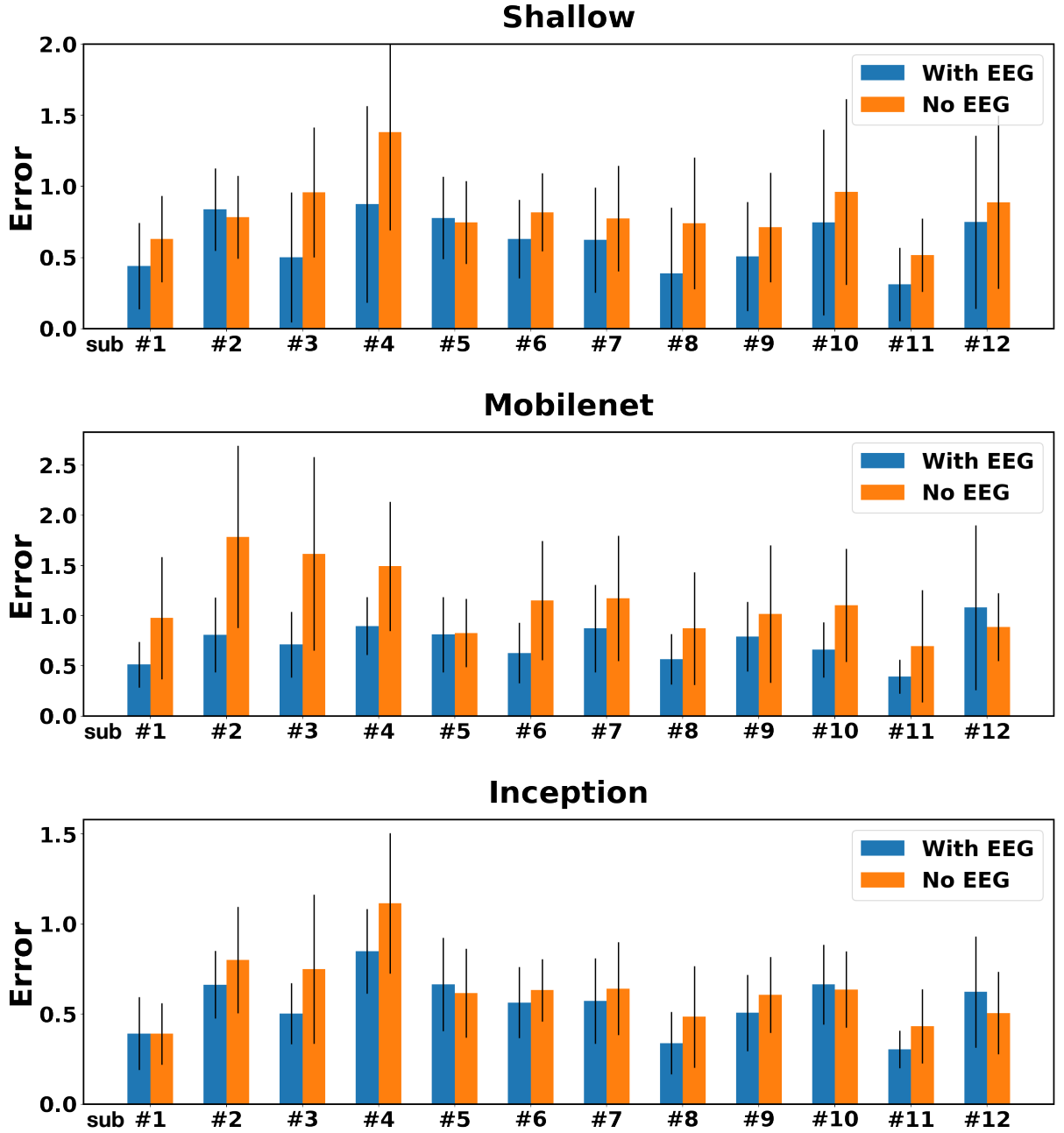


Figure 7.5: Testing error of 3 models with and without EEG. Error is defined as: $\sum_i^m |\text{Neuroscore}_{pred}^{(i)} - \text{Neuroscore}_{true}^{(i)}|$, where $m = 3$ is the number of GAN category used (DCGAN, BEGAN, PROGAN) and Neuroscore is obtained by averaging single-trial P300 amplitude. A smaller value indicates better performance.

Mobilenet-EEG, and Shallow-EEG, which indicates that deeper neural networks may achieve better performance in this task. We used the randomized EEG signal here as a baseline to see the efficacy of adding EEG to produce better Neuroscore output. When randomizing the EEG, it shows that the error for each three model increases significantly. For Mobilenet and Inception, the error of the randomized EEG is even higher than those without EEG in the training stage,

demonstrating that the EEG information in the training stage is crucial to each model.

Figure 7.6 shows that the models with EEG information have a stronger correlation between

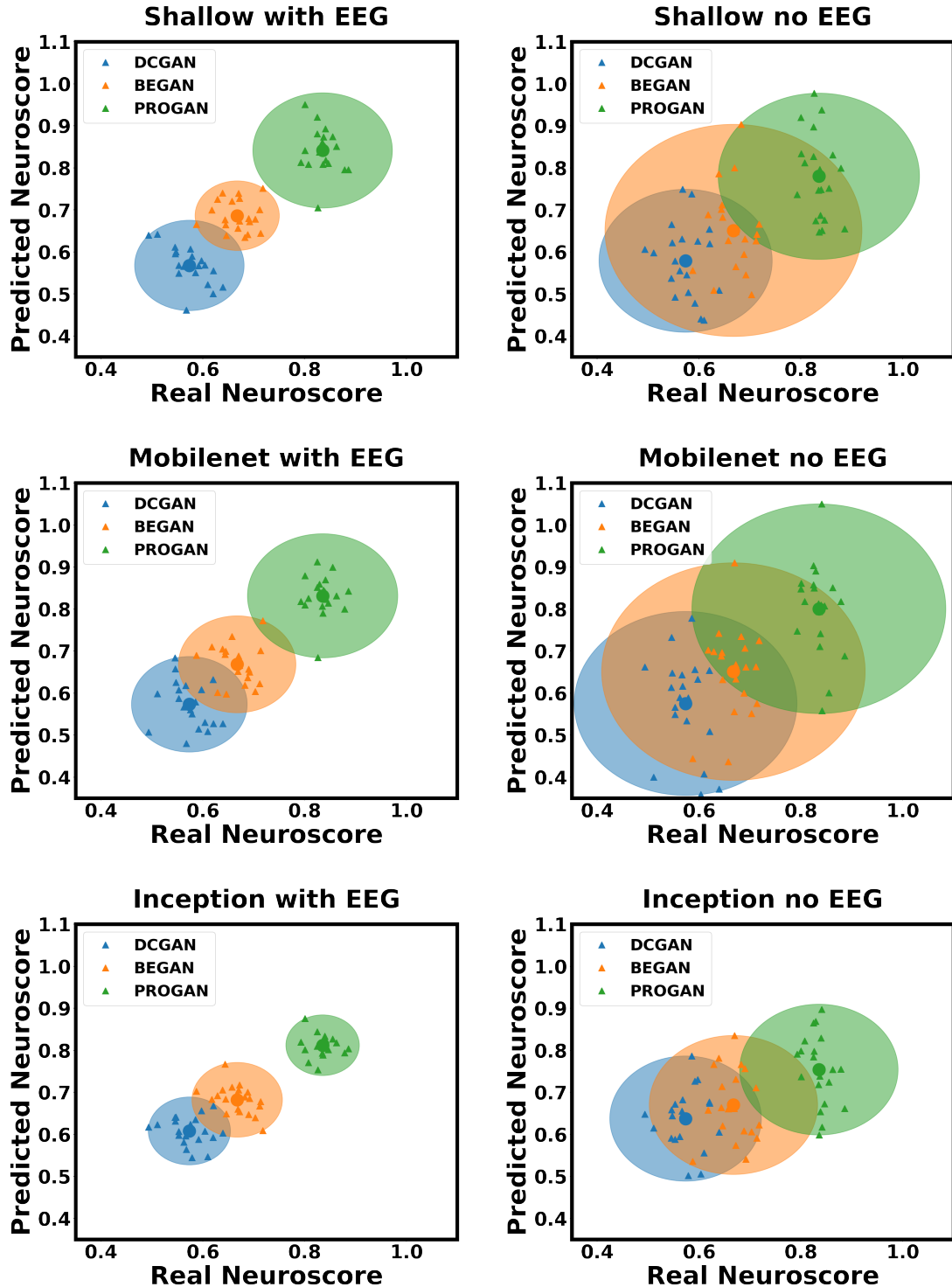


Figure 7.6: Scatter plot of predicted and real Neuroscore of 6 models (Shallow, Mobilenet, Inception with and without EEG for training) cross participants by 20 times repeated shuffling training and testing set. Each circle represents the cluster for a specific category. Small triangle markers inside each cluster correspond to each shuffling process. The dot at the center of each cluster is the mean.

predicted Neuroscore and real Neuroscore (biologically produced by participants). The cluster (blue, orange, and green circles) for each category of the model trained with EEG (left column) is more separable than the cluster produced by the model without EEG (right column). This conveys with EEG for training models: (1) Neuroscore is more accurate; and (2) Neuroscore is able to rank the performances of different GANs.

7.4.2 Neuroscore Aligns with Human Perceptions

Metrics		DCGAN	BEGAN	PROGAN
1/IS		0.44	0.57	0.42
MMD		0.22	0.29	0.12
FID		63.29	83.38	34.10
Ours	1/Shallow-EEG	1.60	1.39	1.14
	1/Mobilenet-EEG	1.71	1.29	1.20
	1/Inception-EEG	1.51	1.34	1.24
Human (BE accuracy)		0.995	0.824	0.705

Table 7.4: Three conventional scores: Inception Score (IS), Kernel Maximum Mean Discrepancy (MMD), Fréchet Inception Distance (FID), and Neuroscore produced by three models with EEG for each GAN category. A lower score indicates better performance of GAN. Neuroscore is consistent with human judgment.

The ultimate goal of GANs is to generate images that are indistinguishable from real images by human beings. Therefore, consistency between an evaluation metric and human perception is a critical requirement for the metric to be considered good. We compare the Neuroscore with three widely used evaluation metrics. Table 7.4 shows the comparison between Neuroscore and three traditional scores. To be consistent with all the scores (smaller score indicates better GAN), we used 1/IS and 1/Neuroscore for comparisons in the Table 7.4. It can be seen that human ranks the GANs performance as: $\text{PROGAN} > \text{BEGAN} > \text{DCGAN}$. All three Neuroscores produced by the three models with EEG are consistent with human judgment while the other three conventional scores are not (they all score DCGAN higher than BEGAN).

7.4.3 Neuroscore Needs Much Smaller Samples

The number of samples for evaluations is crucial in real-world applications considering computational efficiency and efforts for labeling. Traditional metrics need a large sample size to capture the underlying statistical properties of the real and generated images [194,245]. In prac-

tice, it should prefer the metric is not very sensitive to the sample size i.e., the small sample size can also make a good estimation. Figure 7.7 shows that Neuroscore converges stably at around

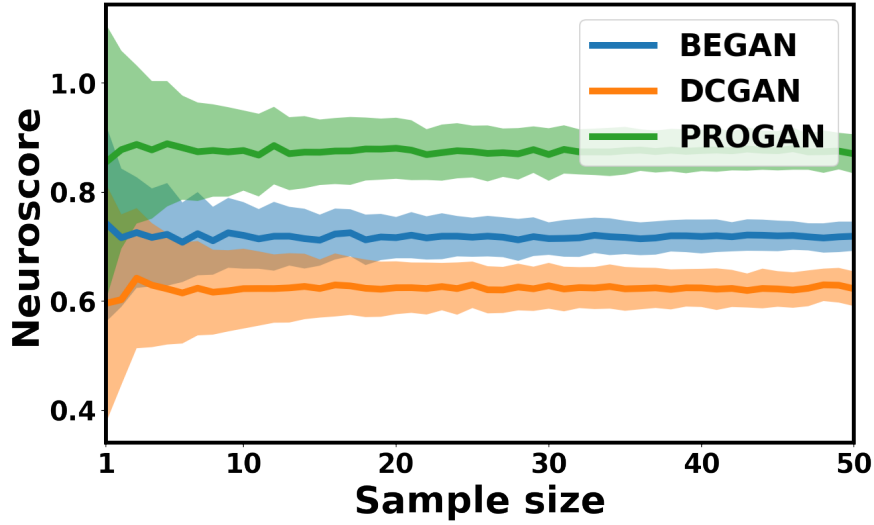


Figure 7.7: Neuroscore of different evaluated sample size for each type of GAN. 200 repeated measurements have been made by randomly shuffling the image samples. The shaded areas refer to the standard deviations while the solid lines refer to the mean values.

20 presentations of a specific type of GAN-produced image (for signal-enhancement purposes), which is much less than the thousands of images required by traditional methods [244, 245]. This is due to the fact that the P300 becomes stable when dozens of EEG trials corresponding to one category are available.

7.4.4 Neuroscore Can Rank Images

Another property of using Neuroscore is the ability to track the quality of an individual image. Traditional evaluation metrics are unable to score each individual image for the two reasons: (1) They need large-scale samples for evaluation; and (2) Most methods (e.g., MMD and FID) evaluate GANs based on the dissimilarity between real images and generated images so they are not able to score the generated image one by one. For our proposed method, the score of each single image can also be evaluated as a single-trial P300 amplitude. We demonstrate that using the predicted single-trial P300 amplitude to observe the single image quality in Fig. 7.8. This property provides Neuroscore with a novel capability that can observe the variations within a typical GAN. Although Neuroscore and Inception Score are generated from DNNs. Neuroscore is more suitable than Inception Score for evaluating GANs in that: (1) It is more explainable than Inception Score as it is a direct reflection of human perception; (2) Much smaller sample

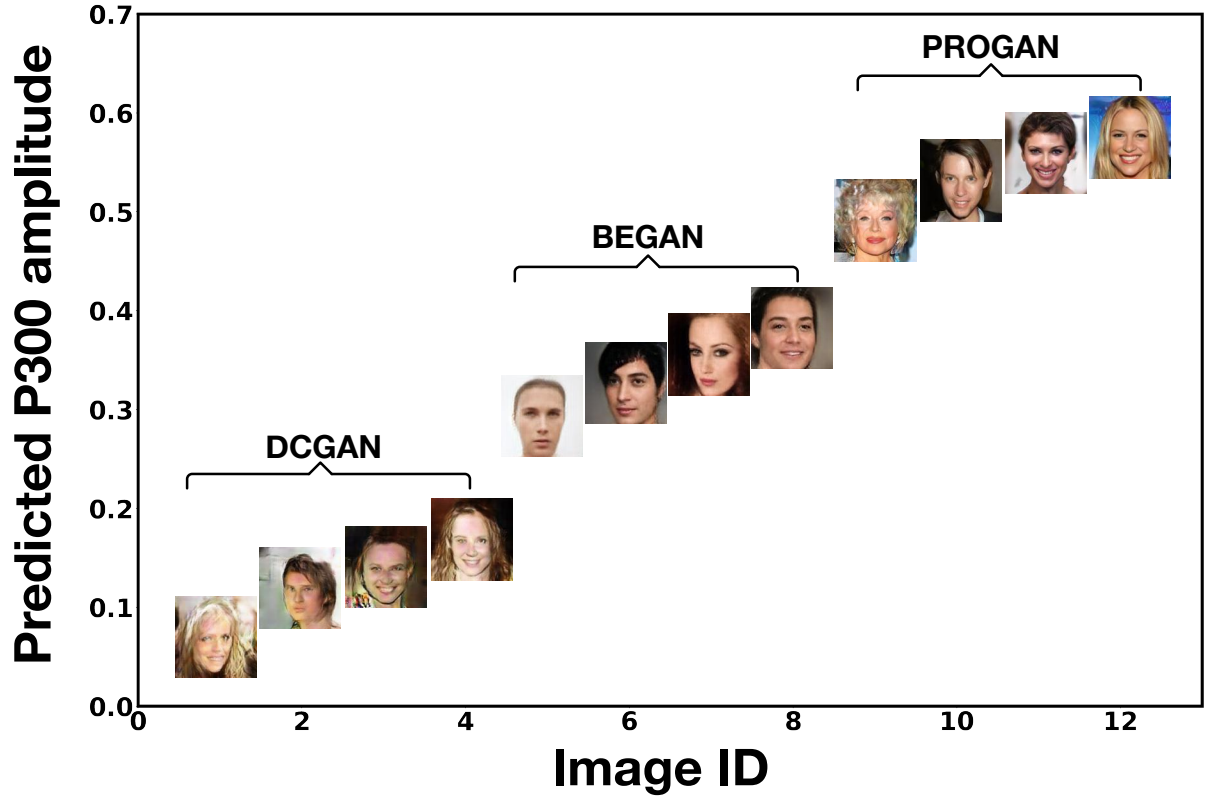


Figure 7.8: P300 amplitude predicted by proposed framework for each single image.

size is required for evaluation; and (3) Higher Neuroscore exactly indicates better image quality while Inception score does not.

7.4.5 Generalization of Neuroscore

We also included the RFACE images in our generalization test. Figure 7.9 demonstrates that the predicted Neuroscore is still correlated with the real Neuroscore when adding the RFACE images and the model ranks the types of images from high quality to low quality as: PROGAN, RFACE, BEGAN and DCGAN, which is consistent with the Neuroscore that has been measured directly from participants shown in Fig. 6.5 in the previous chapter.

Compared to traditional evaluation metrics, Neuroscore is able to score the GANs based on very few image samples relatively. Recording EEG in the training stage could be the limitation of generalizing Neuroscore to evaluate a new GAN. However, the use of dry electrode EEG recording system [324] can accelerate and simplify the data acquisition significantly. Moreover, GANs enable the possibility of synthesizing the EEG [68], which has wide applications in BCIs research.

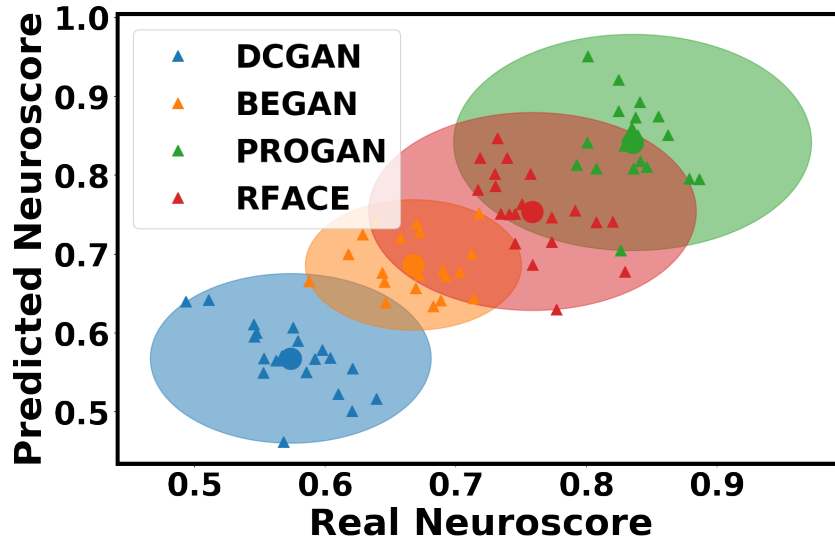


Figure 7.9: Generalization performance of the proposed framework for testing images. EEG corresponding to real face (RFACE) has been included to test the generalization of the architecture.

7.5 Conclusion

In this chapter, we proposed a neuro-AI interface to calculate a synthetic Neuroscore for evaluating the performance of GANs. We validated that this framework is able to learn to predict the Neuroscore using image data alone having been trained using both neural and image data in tandem. Three deep network architectures were explored and the results demonstrate that including neural responses during the training phase of a CNN based neuro-AI interface improves its accuracy even when neural measurements are absent when evaluating on the test set. We compared our Neuroscore measure to traditional evaluation metrics and demonstrated the unique advantages of Neuroscore: (1) It is consistent with human perception; (2) It requires a much smaller number of samples for calculation; and (3) It can rank individual images in terms of quality within a specific GAN. The work presented in this chapter address **research question 3**: *Is it possible to interface biological neural systems and AI systems and if so, can biological neural signals provide any type of informative knowledge for helping AI systems to learn a difficult task?* Our results demonstrated that DNNs can benefit from the information that is encoded by human neural systems when processing the image stimuli.

Chapter 8

Conclusion

***Abstract:** In this chapter, we summarize our work in each previous chapter of the thesis. We also provide potential directions for future research.*

8.1 Summary

This thesis discussed both theoretical aspects and applications of the CCCV systems for image computing. To prepare our discussion, we presented the background knowledge in Chapter 1 from four related areas: (1) Neuroscience area, where we introduced the theoretical explanation and generation for the ERPs (mainly the P300 component) employed in this thesis; (2) Psychology area, where we introduced the RSVP experimental protocol as an oddball paradigm for eliciting the P300 from psychological perspective; (3) BCI area, where we explained the CCCV systems as a specific type of BCI systems when processing image inputs; and (4) Deep learning area, where a GAN, a typical type of DGMs, was introduced for producing image stimulus. This model was then demonstrated for bidirectional communication with the CCCV systems. Three research questions have been proposed related to the work presented in this thesis. The contents in Chapter 4, Chapter 6 and Chapter 7 answered **research question 1:** *Can we improve on the extraction of discriminative ERP components while preserving neurophysiological interpretability for a CCCV system?*, **research question 2:** *Can neural signals be used to provide indications on image quality that is consistent with human perceptual judgment and is it possible to use this as a biological score to evaluate generative models such as GANs?* and **research question 3:** *Is it possible to interface biological neural systems and AI systems and if so, can biological neural signals provide any type of informative knowledge for helping AI systems to*

learn a difficult task?

Our first contribution lies in the area of traditional CCCV research. We contribute a spatial filtering approach — MTWLB, which takes care of both classification and neurophysiological interpretation. This approach would be useful for ERP researchers who are interested in spatial and temporal representation that drive EEG under specific stimulus. We successfully demonstrated the use of MTWLB in Chapter 6 and Chapter 7. Our second contribution proposes the use of CCCV to produce the Neuroscore which is for evaluating the performance of GANs. This type of research will have potential impacts in the area of GANs. First, our framework can be further generalized to auditory BCI, which can be used to evaluate time series based GANs. As evaluation for time series based GANs is very challenging and current literature is very limitedly investigated, our pioneering work can provide some inspiration for this field. Second, our framework provides an example of communication between human neural systems and GANs, where future work can be investigated by using neural signals to improve or even train GANs. This type of research will bridge the gap between human intelligence and artificial intelligence. Our last contribution demonstrates a type of neuro-AI interface, which uses human's neural information as supervisory information for training DNNs. The importance of this work provides indications that human's neural information is helpful for DNNs, which provides prospect for the future research of human-AI interaction. With the advancement of generative models for synthesizing time series data, EEG can be synthesized and of course we do not need to record human's EEG signals when evaluating GANs in the future, which will make the Neuroscore more generalized for practical applications in the future.

With the blooming of machine learning and deep learning techniques today, most image tasks can be solved by advanced computer vision technology. However, current machine learning approaches are still less effective when limited labelled images are provided. We visited this problem by addressing the use of CCCV for image search, which requires less images compared to the traditional computer vision techniques in the literature. AI attracts increasing interest and has significant impact on human society today. It is well known that AI is inspired from human intelligence. ANNs are representative examples that aim to emulate human brain functionality when processing information. Current AI literature focuses on developing learning algorithms to solve real-world tasks but communications and interconnections between AI and human intelligence are explored in a limited way in the current scientific literature. This thesis demonstrated deployment of CCCV, combining traditional BCI technology with GANs, as

bidirectional communications between human's neural responses and GANs where we named this technology as neuro-AI interfacing. We addressed this technology by demonstrating two paradigms. First, as introduced in Chapter 6, GANs produced a number of image stimuli to be presented to participants. Then those stimuli can elicit a different types of neural responses according to different GANs, as a return it produced a biological score called Neuroscore to evaluate the performance of GANs. This work was inspired by the fact that current evaluation metrics for GANs are not able to reflect human perceptual judgment directly. Considering the low efficiency of using human annotations on images one by one, we came up with the use of cortically coupled image computing technology to produce higher throughput processing. It provided insights and directions for designing the human-AI interface in the future. Second, in Chapter 7, we provided details of a CNN based neuro-AI interface to synthesize the Neuroscore when neural recording is absent, where neural signals could be used as supervisory information to train CNNs models. This work demonstrated that biological neural information was able to provide informative knowledge for DNNs when doing a difficult task i.e., evaluate the quality of GAN-generated images in our case. Below we provided an outlook for future research regarding several areas.

8.2 Stepping into ERP Research

Traditional ERP research uses manual-crafted stimuli to be presented to participants and records EEG signals simultaneously. Preparation of stimuli for ERP experiments is normally time-consuming especially for those experiments requiring large number of stimuli e.g., the RSVP experiment discussed in this thesis.

With the fast pace of developments in deep learning, DGMs have been heavily researched in recent years. The most representative example is the GAN. GANs have been demonstrated to be able to produce plausible images in the literature [219,258,267]. In this thesis, we demonstrated the success of eliciting ERPs such as the N170 and the P300 by using face images produced by GANs. Neuroscientific properties such as amplitude, latency, ERSP and ITC of the ERPs produced by real images and generated images can be further investigated. We demonstrated that it is possible to use deep learning technology such as GANs for producing stimuli for a neuroscientific experiment, which is time-saving and self-customizable. Literature shows N2pc [325] can be detected when doing aerial images search task, which can be further investigated if it can

contribute to the Neuroscore.

Compared to image based GANs, evaluation for time series based GANs e.g., use GANs to produce audio and music [326–328] is very limitedly discussed and is more difficult. As a starting point, research on time series based GANs and auditory BCIs can be further investigated. Successful demonstration can be beneficial to either evaluation for time series GANs or producing the auditory stimulus for auditory BCIs in the future.

8.3 Stepping into Computational Neuroscience

In Chapter 7, we have provided a CNN-based neuro-AI interface to predict the Neuroscore. This framework was trained by including human’s neural information (EEG). We used four FC layers after convolutional and pooling layers for encoding information from image feature space to neural space. It is to be explored further if modified architectures could be more robust and have a better performance.

This can be investigated from a computational neuroscience perspective. It has already been demonstrated that DNNs are able to predict the neural response in the V4 cortex inside the brain and achieve human-level task performance simultaneously, where the neural signals are measured via using the electrodes implanted in the V4 cortex inside the brain [90, 308]. From the non-invasive recording side, it has been shown that DNNs are able to capture the stages of human visual processing in both time and space from early visual areas towards the dorsal stream and ventral stream [314, 329], where MEG and fMRI are used as comparison. Thus, literature in the area of computational neuroscience has demonstrated that there are interconnections between DNNs and biological neural systems when encoding information.

BCIs favor the use of EEG because of its safety, inexpensive nature, high time resolution etc. It deserves investigating interconnections between DNNs and biological neural systems using EEG as a media for comparison. Our work in Chapter 7 demonstrates that a DNN model is able to predict Neuroscore with including EEG information and also achieves the good task performance. However, the theory of the P300 generation is still unclear in the neuroscience literature [330–334], which makes it difficult to build a plausible model for better predicting the P300. This requires further research on the P300 generation in the neuroscience field in order to build a DNN model for better emulating the P300 generation process.

8.4 Stepping into Neuro-AI Interface

With development of ANNs architecture and massive amounts of data today, AI systems are able to solve more and more complex tasks in real world. In some tasks such as ImageNet challenge [287], the state-of-the-art DNN [335] is able to achieve competitive performance comparing to human. However, researchers are still exploring what can bring AI closer to human intelligence.

In this thesis, we introduced the basic concept of neuro-AI interfaces, which aim to deploy human beings' neural signals, e.g., EEG, MEG and fMRI, for interfacing AI systems. We briefly described two examples of neuro-AI interfaces. First, in Chapter 6, we demonstrated a neuro-AI interface to provide the neural feedback that is able to assess the performance of GANs. This type of neural feedback is human's neural response that perceived by stimulus generated by GANs. We have demonstrated the ability of this neuro-AI interface for evaluating the performance of GANs and illustrated the its advantages compared to conventional evaluation metrics. Two directions can be further explored in the future: (1) Explore the efficacy of this neuro-AI interface through auditory-related neural dynamics; and (2) Explore the feasibility of using neural information to improve the training of GANs.

Second, we illustrated a neuro-AI interface that a AI system learns from human beings' neural signals when processing the input images. This framework guides the AI system to perceive the object and make the response more similar to the human brain. Compared to the traditional AI systems, the structure of the neuro-AI Interface in this example is indeed more complex. However, it would be useful to solve a problem in a certain circumstance, where it is difficult for the AI system to represent the problem. We have demonstrated an example of this framework in Chapter 7, which is then used to evaluate the performance for GANs based on image quality. Here the "image quality", different from traditional tasks such as object recognition and category classification, is more difficult to define a learning task for an AI system. However, human beings are easily able to perceive different levels of image quality. In this task, neuro-AI interface is able to transfer biological neural information to the AI system so that the AI system can "perceive" image quality like a human.

We introduced the concept of neuro-AI interfaces, which interacts neural dynamics with AI systems for perceiving AI systems based on neural dynamics, helping AI systems make decision and so on. Neuro-AI interfaces deserve being explored in the future because of the following reasons: (1) They are able to assist AI systems in some difficult tasks e.g., evaluate performance

8.4. Stepping into Neuro-AI Interface

for GANs based on image quality presented in this thesis; (2) They make AI systems respond to inputs more similar to the brain; and (3) It is possible that neural systems and AI systems can learn from each other in the future, which brings AI to the real “human intelligence”.

Bibliography

- [1] S. Smith, “EEG in the diagnosis, classification, and management of patients with epilepsy,” *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 76, no. suppl 2, pp. ii2–ii7, 2005.
- [2] N. Kannathal, M. L. Choo, U. R. Acharya, and P. Sadasivan, “Entropies for detection of epilepsy in EEG,” *Computer Methods and Programs in Biomedicine*, vol. 80, no. 3, pp. 187–194, 2005.
- [3] G. Moruzzi and H. W. Magoun, “Brain stem reticular formation and activation of the EEG,” *Electroencephalography and Clinical Neurophysiology*, vol. 1, no. 1-4, pp. 455–473, 1949.
- [4] P. Gloor, G. Ball, and N. Schaul, “Brain lesions that produce delta waves in the EEG,” *Neurology*, vol. 27, no. 4, pp. 326–326, 1977.
- [5] E. Wyllie, D. Lachhwani, A. Gupta, A. Chirla, G. Cosmo, S. Worley, P. Kotagal, P. Ruggieri, and W. Bingaman, “Successful surgery for epilepsy due to early brain lesions despite generalized EEG findings,” *Neurology*, vol. 69, no. 4, pp. 389–397, 2007.
- [6] H. S. Akiskal, T. L. Rosenthal, R. F. Haykal, H. Lemmi, R. H. Rosenthal, and A. Scott-Strauss, “Characterological depressions: Clinical and sleep EEG findings separating ‘subaffective dysthymias’ from ‘character spectrum disorders’,” *Archives of General Psychiatry*, vol. 37, no. 7, pp. 777–783, 1980.
- [7] D. Petit, J.-F. Gagnon, M. L. Fantini, L. Ferini-Strambi, and J. Montplaisir, “Sleep and quantitative EEG in neurodegenerative disorders,” *Journal of Psychosomatic Research*, vol. 56, no. 5, pp. 487–496, 2004.
- [8] C. Besthorn, H. Förstl, C. Geiger-Kabisch, H. Sattel, T. Gasser, and U. Schreiter-Gasser, “EEG coherence in alzheimer disease,” *Electroencephalography and Clinical Neurophysiology*, vol. 90, no. 3, pp. 242–245, 1994.
- [9] M. Penttilä, J. V. Partanen, H. Soininen, and P. Riekkinen, “Quantitative analysis of occipital EEG in different stages of alzheimer’s disease,” *Electroencephalography and Clinical Neurophysiology*, vol. 60, no. 1, pp. 1–6, 1985.
- [10] N. So, G. Savard, F. Andermann, A. Olivier, and L. Quesney, “Acute postictal psychosis: a stereo EEG study,” *Epilepsia*, vol. 31, no. 2, pp. 188–193, 1990.
- [11] R. W. Thatcher, R. Walker, I. Gerson, and F. Geisler, “EEG discriminant analyses of mild head trauma,” *Electroencephalography and Clinical Neurophysiology*, vol. 73, no. 2, pp. 94–106, 1989.

- [12] W. T. Blume, “Drug effects on EEG,” *Journal of Clinical Neurophysiology*, vol. 23, no. 4, pp. 306–311, 2006.
- [13] T. Ganes and T. Lundar, “EEG and evoked potentials in comatose patients with severe brain damage,” *Electroencephalography and Clinical Neurophysiology*, vol. 69, no. 1, pp. 6–13, 1988.
- [14] O. N. Markand, “Alpha rhythms,” *Journal of Clinical Neurophysiology*, vol. 7, no. 2, pp. 163–190, 1990.
- [15] M. J. Aminoff, “Electroencephalography: General principles and clinical applications,” *Aminoff’s Electrodiagnosis in Clinical Neurology: Expert Consult-Online and Print*, p. 37, 2012.
- [16] D. L. Schacter, “EEG theta waves and psychological phenomena: A review and analysis,” *Biological Psychology*, vol. 5, no. 1, pp. 47–82, 1977.
- [17] W. Klimesch, “EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis,” *Brain Research Reviews*, vol. 29, no. 2-3, pp. 169–195, 1999.
- [18] M. Teplan *et al.*, “Fundamentals of EEG measurement,” *Measurement Science Review*, vol. 2, no. 2, pp. 1–11, 2002.
- [19] I. Feinberg, T. Floyd, and J. March, “Effects of sleep loss on delta (0.3–3 Hz) EEG and eye movement density: New observations and hypotheses,” *Electroencephalography and Clinical Neurophysiology*, vol. 67, no. 3, pp. 217–221, 1987.
- [20] W. Carroll and F. Mastaglia, “Alpha and beta coma in drug intoxication uncomplicated by cerebral hypoxia,” *Electroencephalography and Clinical Neurophysiology*, vol. 46, no. 1, pp. 95–105, 1979.
- [21] G. Buzsaki, *Rhythms of the Brain*. Oxford, England, UK: Oxford University Press, 2006.
- [22] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, “Brain–computer interfaces for communication and control,” *Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.
- [23] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, “A review of classification algorithms for EEG-based brain–computer interfaces,” *Journal of Neural Engineering*, vol. 4, no. 2, p. R1, 2007.
- [24] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, “A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update,” *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, 2018.
- [25] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He, “Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface,” *Journal of Neural Engineering*, vol. 10, no. 4, p. 046003, 2013.
- [26] R. Fazel-Rezai and K. Abhari, “A region-based P300 speller for brain–computer interface,” *Canadian Journal of Electrical and Computer Engineering*, vol. 34, no. 3, pp. 81–85, 2009.

- [27] E. A. Pohlmeier, J. Wang, D. C. Jangraw, B. Lou, S.-F. Chang, and P. Sajda, "Closing the loop in cortically-coupled computer vision: A brain-computer interface for searching image databases," *Journal of Neural Engineering*, vol. 8, no. 3, p. 036025, 2011.
- [28] D. Blackwood and W. Muir, "Cognitive brain potentials and their application," *The British Journal of Psychiatry*, vol. 157, no. S9, pp. 96–101, 1990.
- [29] E. Natale, C. Marzi, M. Girelli, E. Pavone, and S. Pollmann, "ERP and fMRI correlates of endogenous and exogenous focusing of visual-spatial attention," *European Journal of Neuroscience*, vol. 23, no. 9, pp. 2511–2521, 2006.
- [30] S. Sur and V. Sinha, "Event-related potential: An overview," *Industrial Psychiatry Journal*, vol. 18, no. 1, p. 70, 2009.
- [31] S. Sutton, M. Braren, J. Zubin, and E. John, "Evoked-potential correlates of stimulus uncertainty," *Science*, vol. 150, no. 3700, pp. 1187–1188, 1965.
- [32] E. Donchin, W. Ritter, W. C. McCallum *et al.*, "Cognitive psychophysiology: The endogenous components of the ERP," *Event-related Brain Potentials in Man*, pp. 349–411, 1978.
- [33] W. S. Pritchard, "Psychophysiology of P300," *Psychological Bulletin*, vol. 89, no. 3, p. 506, 1981.
- [34] R. Johnson Jr, "On the neural generators of the P300 component of the event-related potential," *Psychophysiology*, vol. 30, no. 1, pp. 90–97, 1993.
- [35] E. Courchesne, S. A. Hillyard, and R. Galambos, "Stimulus novelty, task relevance and the visual evoked potential in man," *Electroencephalography and Clinical Neurophysiology*, vol. 39, no. 2, pp. 131–143, 1975.
- [36] R. T. Knight, "Decreased response to novel stimuli after prefrontal lesions in man," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 59, no. 1, pp. 9–20, 1984.
- [37] E. Snyder and S. A. Hillyard, "Long-latency evoked potentials to irrelevant, deviant stimuli," *Behavioral Biology*, vol. 16, no. 3, pp. 319–331, 1976.
- [38] N. K. Squires, K. C. Squires, and S. A. Hillyard, "Two varieties of long-latency positive waves evoked by unpredictable auditory stimuli in man," *Electroencephalography and Clinical Neurophysiology*, vol. 38, no. 4, pp. 387–401, 1975.
- [39] J. Polich, "Updating P300: An integrative theory of P3a and P3b," *Clinical Neurophysiology*, vol. 118, no. 10, pp. 2128–2148, 2007.
- [40] L. A. Farwell and E. Donchin, "Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials," *Electroencephalography and Clinical Neurophysiology*, vol. 70, no. 6, pp. 510–523, 1988.
- [41] R. Fazel-Rezai, B. Z. Allison, C. Guger, E. W. Sellers, S. C. Kleih, and A. Kübler, "P300 brain-computer interface: Current challenges and emerging trends," *Frontiers in Neuro-engineering*, vol. 5, 2012.

- [42] E. Donchin, K. M. Spencer, and R. Wijesinghe, “The mental prosthesis: Assessing the speed of a P300-based brain-computer interface,” *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 174–179, 2000.
- [43] A. Kübler, A. Furdea, S. Halder, E. M. Hammer, F. Nijboer, and B. Kotchoubey, “A brain-computer interface controlled auditory event-related potential (P300) spelling system for locked-in patients,” *Annals of the New York Academy of Sciences*, vol. 1157, no. 1, pp. 90–100, 2009.
- [44] E. W. Sellers, T. M. Vaughan, and J. R. Wolpaw, “A brain-computer interface for long-term independent home use,” *Amyotrophic Lateral Sclerosis*, vol. 11, no. 5, pp. 449–455, 2010.
- [45] S. C. Kleih, T. Kaufmann, C. Zickler, S. Halder, F. Leotta, F. Cincotti, F. Aloise, A. Riccio, C. Herbert, D. Mattia *et al.*, “Out of the frying pan into the fire—the P300-based BCI faces real-world challenges,” in *Progress in Brain Research*. Elsevier, 2011, vol. 194, pp. 27–46.
- [46] Z. Lin, C. Zhang, Y. Zeng, L. Tong, and B. Yan, “A novel P300 BCI speller based on the triple RSVP paradigm,” *Scientific Reports*, vol. 8, no. 1, p. 3350, 2018.
- [47] G. Townsend, B. LaPallo, C. Boulay, D. Krusienski, G. Frye, C. Hauser, N. Schwartz, T. Vaughan, J. R. Wolpaw, and E. Sellers, “A novel P300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns,” *Clinical Neurophysiology*, vol. 121, no. 7, pp. 1109–1120, 2010.
- [48] M. Arvaneh, I. H. Robertson, and T. E. Ward, “A P300-based brain-computer interface for improving attention,” *Frontiers in Human Neuroscience*, vol. 12, 2018.
- [49] D. H. Lawrence, “Two studies of visual search for word targets with controlled rates of presentation,” *Perception & Psychophysics*, vol. 10, no. 2, pp. 85–89, 1971.
- [50] R. Spence and M. Witkowski, *Rapid serial visual presentation: Design for cognition*. Salmon Tower Building, New York City, United States: Springer, 2013.
- [51] M. C. Potter, “Short-term conceptual memory for pictures.” *Journal of Experimental Psychology: Human Learning and Memory*, vol. 2, no. 5, p. 509, 1976.
- [52] J. Duncan, “The locus of interference in the perception of simultaneous stimuli.” *Psychological Review*, vol. 87, no. 3, p. 272, 1980.
- [53] A. M. Treisman, “Effect of irrelevant material on the efficiency of selective listening.” *The American Journal of Psychology*, 1964.
- [54] U. Neisser, *Cognitive psychology: Classic edition*. 39 Church Rd, Hove BN3 2BU, UK: Psychology Press, 2014.
- [55] J. E. Raymond, K. L. Shapiro, and K. M. Arnell, “Temporary suppression of visual processing in an RSVP task: An attentional blink?” *Journal of Experimental Psychology: Human Perception and Performance*, vol. 18, no. 3, p. 849, 1992.
- [56] Z. Lin, Y. Zeng, H. Gao, L. Tong, C. Zhang, X. Wang, Q. Wu, and B. Yan, “Multirapid serial visual presentation framework for EEG-based target detection,” *BioMed Research International*, 2017.

- [57] K. Cooper, O. de Bruijn, R. Spence, and M. Witkowski, "A comparison of static and moving presentation modes for image collections," in *Proceedings of the Working Conference on Advanced Visual Interfaces*. ACM, 2006, pp. 381–388.
- [58] S. Corsato, M. Mosconi, and M. Porta, "An eye tracking approach to image search activities using RSVP display techniques," in *Proceedings of the Working Conference on Advanced Visual Interfaces*. ACM, 2008, pp. 416–420.
- [59] A. D. Gerson, L. C. Parra, and P. Sajda, "Cortically coupled computer vision for rapid image search," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 174–179, 2006.
- [60] J. Meng, L. M. Meriño, N. B. Shamlo, S. Makeig, K. Robbins, and Y. Huang, "Characterization and robust classification of EEG signal from image RSVP events with independent time-frequency features," *PLoS ONE*, vol. 7, no. 9: e44464. <https://doi.org/10.1371/journal.pone.0044464>, 2012.
- [61] N. Bigdely-Shamlo, A. Vankov, R. R. Ramirez, and S. Makeig, "Brain activity-based image classification from rapid serial visual presentation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 16, no. 5, pp. 432–441, 2008.
- [62] P. Sajda, E. Pohlmeier, J. Wang, L. C. Parra, C. Christoforou, J. Dmochowski, B. Hanna, C. Bahlmann, M. K. Singh, and S.-F. Chang, "In a blink of an eye and a switch of a transistor: Cortically coupled computer vision," *Proceedings of the IEEE*, vol. 98, no. 3, pp. 462–478, 2010.
- [63] London satellite map. [Online]. Available: <https://www.harrisgeospatial.com/Data-Imagery/DownloadArea/>.
- [64] P. Sajda, A. D. Gerson, M. G. Philiastides, and L. C. Parra, "Single-trial analysis of EEG during rapid visual discrimination: Enabling cortically-coupled computer vision," *Towards Brain-computer Interfacing*, pp. 423–44, 2007.
- [65] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.
- [66] I. A. Corley and Y. Huang, "Deep EEG super-resolution: Upsampling EEG spatial resolution with generative adversarial networks," in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics*. Las Vegas, NV, USA: IEEE, 2018, pp. 100–103.
- [67] S. M. Abdelfattah, G. M. Abdelrahman, and M. Wang, "Augmenting the size of eeg datasets using generative adversarial networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*. Rio de Janeiro, Brazil: IEEE, 2018, pp. 1–6.
- [68] K. G. Hartmann, R. T. Schirrmeister, and T. Ball, "EEG-GAN: Generative adversarial networks for electroencephalographic brain signals," *arXiv preprint arXiv:1806.01875*, 2018.
- [69] S. J. Luck, *An introduction to the event-related potential technique*. Third floor of 1 Rogers Street in Cambridge, MA 02142, Boston: MIT Press, 2014.

- [70] The synapse. [Online]. Available: <https://www.khanacademy.org/science/biology/human-biology/neuron-nervous-system/a/the-synapse>.
- [71] T. Splettstoesser. Synapse schematic. [Online]. Available: <https://commons.wikimedia.org/w/index.php?curid=41349083>.
- [72] D. H. Perkel, G. L. Gerstein, and G. P. Moore, "Neuronal spike trains and stochastic point processes: I. the single spike train," *Biophysical Journal*, vol. 7, no. 4, pp. 391–418, 1967.
- [73] C. M. Bishop, *Pattern recognition and machine learning*. New York City, United States: Springer, 2006.
- [74] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Third floor of 1 Rogers Street in Cambridge, MA 02142, Boston: MIT press, 2016.
- [75] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [76] What is the differences between artificial neural network (computer science) and biological neural network? [Online]. Available: <https://www.quora.com/What-is-the-differences-between-artificial-neural-network-computer-science-and-biological-neural-network>.
- [77] P. Dayan and L. Abbott, *Theoretical Neuroscience*. Third floor of 1 Rogers Street in Cambridge, MA 02142, Boston: MIT Press, 2001.
- [78] Q. She, "Flexible and interpretable multivariate point processes for neural dynamics," Ph.D. dissertation, City University of Hong Kong, 2018.
- [79] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems*, 2017, pp. 3856–3866.
- [80] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [81] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, no. 2, pp. 270–280, 1989.
- [82] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurones in the cat's striate cortex," *The Journal of Physiology*, vol. 148, no. 3, pp. 574–591, 1959.
- [83] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.
- [84] D. Marr *et al.*, *Vision: A computational investigation into the human representation and processing of visual information*. San Francisco, United States: W.H. Freeman, 1982.
- [85] L. McIntosh, N. Maheswaranathan, A. Nayebi, S. Ganguli, and S. Baccus, "Deep learning models of the retinal response to natural scenes," in *Advances in Neural Information Processing Systems*, 2016, pp. 1369–1377.

- [86] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Conference of the International Speech Communication Association*, Makuhari, Chiba, Japan, 2010, pp. 1045–1048.
- [87] D. S. Bassett and E. Bullmore, “Small-world brain networks,” *The Neuroscientist*, vol. 12, no. 6, pp. 512–523, 2006.
- [88] M. P. van den Heuvel and O. Sporns, “Network hubs in the human brain,” *Trends in Cognitive Sciences*, vol. 17, no. 12, pp. 683–696, 2013.
- [89] Q. She, G. Chen, and R. H. Chan, “Evaluating the small-world-ness of a sampled network: Functional connectivity of entorhinal-hippocampal circuitry,” *Scientific Reports*, vol. 6, p. 21468, 2016.
- [90] D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo, “Performance-optimized hierarchical models predict neural responses in higher visual cortex,” *Proceedings of the National Academy of Sciences*, vol. 111, no. 23, pp. 8619–8624, 2014.
- [91] P. Bashivan, K. Kar, and J. J. DiCarlo, “Neural population control via deep image synthesis,” *Science*, vol. 364, no. 6439, p. eaav9436, 2019.
- [92] Z. Wang, G. Healy, A. F. Smeaton, and T. E. Ward, “Chapter 12: A review of feature extraction and classification algorithms for image RSVP based BCI,” *Signal Processing and Machine Learning for Brain-Machine Interfaces*, pp. 243–270, The Institute of Engineering and Technology. Michael Faraday House, Six Hills Way, Stevenage, SG1 2AY, UK, 2018.
- [93] —, “Spatial filtering pipeline evaluation of cortically coupled computer vision system for rapid serial visual presentation,” *Brain-Computer Interfaces*, vol. 5, pp. 132–145, 2018.
- [94] K. Kurach, M. Lucic, X. Zhai, M. Michalski, and S. Gelly, “The GAN landscape: Losses, architectures, regularization, and normalization,” *arXiv preprint arXiv:1807.04720*, 2018.
- [95] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, “Generative adversarial networks: Introduction and outlook,” *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.
- [96] S. Hitawala, “Comparative study on generative adversarial networks,” *arXiv preprint arXiv:1801.04271*, 2018.
- [97] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative adversarial networks: An overview,” *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.
- [98] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, “How generative adversarial networks and their variants work: An overview,” *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, p. 10, 2019.

- [99] J. Shamwell, H. Lee, H. Kwon, A. R. Marathe, V. Lawhern, and W. Nothwang, "Single-trial EEG RSVP classification using convolutional neural networks," in *Micro-and Nanotechnology Sensors, Systems, and Applications VIII*, vol. 9836. Baltimore, Maryland, United States: International Society for Optics and Photonics, 2016, p. 983622.
- [100] R. Manor and A. B. Geva, "Convolutional neural network for multi-category rapid serial visual presentation BCI," *Frontiers in Computational Neuroscience*, vol. 9, p. 146, 2015.
- [101] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.
- [102] G. Healy, T. Ward, C. Gurrin, and A. F. Smeaton, "Overview of NTCIR-13 NAILS task," in *The 13th NTCIR (2016 - 2017) Evaluation of Information Access Technologies June 2016 - December 2017 Conference*, Tokyo, Japan, 5-8 Dec 2017.
- [103] V. Lawhern, A. Solon, N. Waytowich, S. M. Gordon, C. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, 2018.
- [104] G. Healy, Z. Wang, C. Currin, T. E. Ward, and A. F. Smeaton, "An EEG image-search dataset: A first-of-its-kind in IR/IIR. NAILS: neurally augmented image labelling strategies," in *CHIIR Workshop on Challenges in Bringing Neuroscience to Research in Human-Information Interaction*, Oslo, Norway, 11 Mar 2017.
- [105] T. N. Lal, T. Hinterberger, G. Widman, M. Schröder, N. J. Hill, W. Rosenstiel, C. E. Elger, N. Birbaumer, and B. Schölkopf, "Methods towards invasive human brain-computer interfaces," in *Advances in Neural Information Processing Systems*, 2005, pp. 737–744.
- [106] J. R. Wolpaw and D. J. McFarland, "Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proceedings of the National Academy of Sciences*, vol. 101, no. 51, pp. 17 849–17 854, 2004.
- [107] H. Anupama, N. Cauvery, and G. Lingaraju, "Brain-computer interface and its types-a study," *International Journal of Advances in Engineering & Technology*, vol. 3, no. 2, p. 739, 2012.
- [108] E. Mohedano, K. McGuinness, G. Healy, N. E. O'Connor, A. F. Smeaton, A. Salvador, S. Porta, and X. Giró-i Nieto, "Exploring EEG for object detection and retrieval," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. Shanghai, China: ACM, 2015, pp. 591–594.
- [109] K. K. Ang and C. Guan, "Brain-computer interface in stroke rehabilitation," *Journal of Computing Science and Engineering*, vol. 7, no. 2, pp. 139–146, 2013.
- [110] A. Rakotomamonjy and V. Guigue, "BCI competition III: Dataset II-ensemble of svms for bci p300 speller," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 1147–1154, 2008.
- [111] S. Lees, N. Dayan, H. Cecotti, P. Mccullagh, L. Maguire, F. Lotte, and D. Coyle, "A review of rapid serial visual presentation-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 2, p. 021001, 2018.

- [112] Z. Wang, G. Healy, A. F. Smeaton, and T. E. Ward, “An investigation of triggering approaches for the rapid serial visual presentation paradigm in brain computer interfacing,” in *27th Irish Signals and Systems Conference*. Londonderry, UK: IEEE, 2016, pp. 1–6.
- [113] M. Plöchl, J. P. Ossandón, and P. König, “Combining EEG and eye tracking: Identification, characterization, and correction of eye movement artifacts in electroencephalographic data,” *Frontiers in Human Neuroscience*, vol. 6, 2012.
- [114] P. Sajda, A. Gerson, and L. Parra, “High-throughput image search via single-trial event detection in a rapid serial visual presentation task,” in *First International IEEE EMBS Conference on Neural Engineering*. Capri Island, Italy: IEEE, 2003, pp. 7–10.
- [115] K.-R. Müller, M. Krauledat, G. Dornhege, G. Curio, and B. Blankertz, “Machine learning techniques for brain-computer interfaces,” *Biomed. Tech*, vol. 49, no. 1, pp. 11–22, 2004.
- [116] Y. Huang, D. Erdogmus, M. Pavel, S. Mathan, and K. E. Hild II, “A framework for rapid visual image search using single-trial brain evoked responses,” *Neurocomputing*, vol. 74, no. 12-13, pp. 2041–2051, 2011.
- [117] P. Indyk and R. Motwani, “Approximate nearest neighbors: Towards removing the curse of dimensionality,” in *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*. ACM, 1998, pp. 604–613.
- [118] J. W. Peirce, “PsychoPy—psychophysics software in Python,” *Journal of Neuroscience Methods*, vol. 162, no. 1-2, pp. 8–13, 2007.
- [119] B. MacWhinney, J. S. James, C. Schunn, P. Li, and W. Schneider, “STEP—A system for teaching experimental psychology using E-Prime,” *Behavior Research Methods, Instruments, & Computers*, vol. 33, no. 2, pp. 287–296, 2001.
- [120] K. Yu, K. Shen, S. Shao, W. C. Ng, K. Kwok, and X. Li, “Common spatio-temporal pattern for single-trial detection of event-related potential in rapid serial visual presentation triage,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 9, pp. 2513–2520, 2011.
- [121] C. Kothe. (26th November 2018) Lab streaming layer. [Online]. Available: <http://github.com/sccn/labstreaminglayer/wiki>.
- [122] H. Begleiter, *Evoked brain potentials and behavior*. Berlin, Germany: Springer Science & Business Media, 2012, vol. 2.
- [123] A. M. Norcia, L. G. Appelbaum, J. M. Ales, B. R. Cottreau, and B. Rossion, “The steady-state visual evoked potential in vision research: A review,” *Journal of Vision*, vol. 15, no. 6, pp. 4–4, 2015.
- [124] C. Vidaurre and B. Blankertz, “Towards a cure for BCI illiteracy,” *Brain Topography*, vol. 23, no. 2, pp. 194–198, 2010.
- [125] C. Elkan, “The foundations of cost-sensitive learning,” in *International Joint Conference on Artificial Intelligence*, vol. 17, no. 1, Seattle, Washington, United States, 2001, pp. 973–978.

- [126] M. X. Cohen, *Analyzing neural time series data: Theory and practice*. Third floor of 1 Rogers Street in Cambridge, MA 02142, Boston: MIT Press, 2014.
- [127] A. J. Solon, S. M. Gordon, B. Lance, and V. Lawhern, “Deep learning approaches for p300 classification in image triage: Applications to the NAILS task,” in *Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies, NTCIR-13, Tokyo, Japan*, 2017, pp. 5–8.
- [128] J. A. Hanley and B. J. McNeil, “The meaning and use of the area under a receiver operating characteristic (ROC) curve,” *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.
- [129] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, “The balanced accuracy and its posterior distribution,” in *2010 20th International Conference on Pattern Recognition*. Istanbul, Turkey: IEEE, 2010, pp. 3121–3124.
- [130] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert, “xDAWN algorithm to enhance evoked potentials: Application to brain–computer interface,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 8, pp. 2035–2043, 2009.
- [131] Y. Jonmohamadi, G. Poudel, C. Innes, D. Weiss, R. Krueger, and R. Jones, “Comparison of beamformers for EEG source signal reconstruction,” *Biomedical Signal Processing and Control*, vol. 14, pp. 175–188, 2014.
- [132] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Müller, “Optimizing spatial filters for robust EEG single-trial analysis,” *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 41–56, 2008.
- [133] Y. Wang, S. Gao, and X. Gao, “Common spatial pattern method for channel selection in motor imagery based brain-computer interface,” in *Engineering in Medicine and Biology Society. 2005 Annual International Conference of the IEEE*. Shanghai, China: IEEE, 2006, pp. 5392–5395.
- [134] H. Cecotti, M. P. Eckstein, and B. Giesbrecht, “Single-trial classification of event-related potentials in rapid serial visual presentation tasks using supervised spatial filtering,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 11, pp. 2030–2042, 2014.
- [135] M. S. Treder, A. K. Porbadnigk, F. S. Avarvand, K.-R. Müller, and B. Blankertz, “The LDA beamformer: Optimal estimation of ERP source time series using linear discriminant analysis,” *NeuroImage*, vol. 129, pp. 279–291, 2016.
- [136] S. Makeig, J. Onton *et al.*, “ERP features and EEG dynamics: An ICA perspective,” *Oxford Handbook of Event-Related Potential Components*. New York, NY: Oxford, 2009.
- [137] A. Hyvärinen and E. Oja, “Independent component analysis: Algorithms and applications,” *Neural Networks*, vol. 13, no. 4, pp. 411–430, 2000.
- [138] M. Mohammadi, S. H. Sardouie, and M. B. Shamsollahi, “Denoising of interictal EEG signals using ICA and time varying AR modeling,” in *2014 21th Iranian Conference on Biomedical Engineering*. Tehran, Iran: IEEE, 2014, pp. 144–149.

- [139] T.-P. Jung, S. Makeig, C. Humphries, T.-W. Lee, M. J. Mckeown, V. Iragui, and T. J. Sejnowski, "Removing electroencephalographic artifacts by blind source separation," *Psychophysiology*, vol. 37, no. 2, pp. 163–178, 2000.
- [140] J. Shlens, "A tutorial on principal component analysis," *arXiv preprint arXiv:1404.1100*, 2014.
- [141] M. Naeem, C. Brunner, and G. Pfurtscheller, "Dimensionality reduction and channel selection of motor imagery electroencephalographic data," *Computational Intelligence and Neuroscience*, vol. 2009, 2009.
- [142] T. Zanotelli, S. Santos Filho, and C. Tierra-Criollo, "Optimum principal components for spatial filtering of EEG to detect imaginary movement by coherence," in *Engineering in Medicine and Biology Society, 2010 Annual International Conference of the IEEE*. Buenos Aires, Argentina: IEEE, 2010, pp. 3646–3649.
- [143] G. F. Alpert, R. Manor, A. B. Spanier, L. Y. Deouell, and A. B. Geva, "Spatiotemporal representations of rapid visual target detection: A single-trial EEG classification algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 8, pp. 2290–2303, 2014.
- [144] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, 2001.
- [145] S. Makeig, "Auditory event-related dynamics of the EEG spectrum and effects of exposure to tones," *Electroencephalography and Clinical Neurophysiology*, vol. 86, no. 4, pp. 283–293, 1993.
- [146] L. C. Parra, C. Christoforou, A. C. Gerson, M. Dyrholm, A. Luo, M. Wagner, M. G. Philiastides, and P. Sajda, "Spatiotemporal linear decoding of brain state," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 107–115, 2008.
- [147] J. H. Friedman, "Regularized discriminant analysis," *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 165–175, 1989.
- [148] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. Hoboken, New Jersey, United States: John Wiley & Sons, 2012.
- [149] J.-H. Xue and D. M. Titterton, "Do unbalanced data have a negative effect on LDA ?" *Pattern Recognition*, vol. 41, no. 5, pp. 1558–1571, 2008.
- [150] D. J. MacKay, "Bayesian interpolation," *Neural Computation*, vol. 4, no. 3, pp. 415–447, 1992.
- [151] U. Hoffmann, J.-M. Vesin, T. Ebrahimi, and K. Diserens, "An efficient P300-based brain-computer interface for disabled subjects," *Journal of Neuroscience methods*, vol. 167, no. 1, pp. 115–125, 2008.
- [152] H. Cecotti and A. J. Ries, "Best practice for single-trial detection of event-related potentials: Application to brain-computer interfaces," *International Journal of Psychophysiology*, vol. 111, pp. 156–169, 2017.

- [153] Y. Huang, D. Erdogmus, S. Mathan, and M. Pavel, "Boosting linear logistic regression for single-trial ERP detection in rapid serial visual presentation tasks," in *Engineering in Medicine and Biology Society*. New York, United States: IEEE, 2006, pp. 3369–3372.
- [154] E. E. Osuna, "Support vector machines: Training and applications," Ph.D. dissertation, Massachusetts Institute of Technology, 1998.
- [155] A. Abdiansah and R. Wardoyo, "Time complexity analysis of support vector machines (svm) in libsvm," *International Journal Computer and Application*, 2015.
- [156] G. Healy and A. F. Smeaton, "Optimising the number of channels in EEG-augmented image search," in *Proceedings of the 25th BCS Conference on Human-Computer Interaction*. Newcastle, UK: British Computer Society, 2011, pp. 157–162.
- [157] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, Oct 2011.
- [158] C. M. Bishop *et al.*, *Neural networks for pattern recognition*. Oxford, UK: Oxford university press, 1995.
- [159] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in *Neural Networks for Perception*. Elsevier, 1992, pp. 65–93.
- [160] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.
- [161] D. Balakrishnan and S. Puthusserypady, "Multilayer perceptrons for the classification of brain-computer interface data," in *Proceedings of the IEEE 31st Annual Northeast Bioengineering Conference*. Hoboken, NJ, United States: IEEE, 2005, pp. 118–119.
- [162] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.
- [163] G. Carneiro, J. C. Nascimento, and A. Freitas, "The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 968–982, 2012.
- [164] J. Gao, X. He, and L. Deng, "Deep learning for web search and natural language processing," Deep Learning Technology Center (DLTC), Microsoft Research, Redmond, USA, Tech. Rep., January 2015. [Online]. Available: <https://www.microsoft.com/en-us/research/publication/deep-learning-for-web-search-and-natural-language-processing/>.
- [165] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, United States, 2015, pp. 4353–4361.
- [166] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.

- [167] B. Hu, Z. Lu, H. Li, and Q. Chen, “Convolutional neural network architectures for matching natural language sentences,” *Advances in Neural Information Processing Systems*, vol. 3, pp. 2042–2050, 2015.
- [168] M. Liang and X. Hu, “Recurrent convolutional neural network for object recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, United States, 2015, pp. 3367–3375.
- [169] H. Cecotti, “Convolutional neural networks for event-related potential detection: Impact of the architecture,” in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Jeju Island, Korea: IEEE, 2017, pp. 2031–2034.
- [170] E. D. Übeyli, “Analysis of EEG signals by implementing eigenvector methods/recurrent neural networks,” *Digital Signal Processing*, vol. 19, no. 1, pp. 134–143, 2009.
- [171] N. F. Güler, E. D. Übeyli, and I. Güler, “Recurrent neural networks employing lyapunov exponents for EEG signals classification,” *Expert Systems with Applications*, vol. 29, no. 3, pp. 506–514, 2005.
- [172] E. M. Forney and C. W. Anderson, “Classification of EEG during imagined mental tasks by forecasting with elman recurrent neural networks,” in *International Joint Conference on Neural Networks*, San Jose, California, United States, 2011, pp. 2749–2755.
- [173] G. E. Hinton, “Deep belief networks,” *Scholarpedia*, vol. 4, no. 6, p. 5947, 2009.
- [174] —, “A practical guide to training restricted boltzmann machines,” *Momentum*, vol. 9, no. 1, pp. 599–619, 2012.
- [175] S. Ahmed, L. M. Merino, Z. Mao, J. Meng, K. Robbins, and Y. Huang, “A deep learning method for classification of images rsvp events with EEG data,” in *IEEE Global Conference on Signal and Information Processing*, Austin, Texas, United States, 2013, pp. 33–36.
- [176] G. Healy and A. F. Smeaton, “Eye fixation related potentials in a target search task,” in *Engineering in Medicine and Biology Society, 2011 Annual International Conference of the IEEE*. Boston, United States: IEEE, 2011, pp. 4203–4206.
- [177] J. Onton, M. Westerfield, J. Townsend, and S. Makeig, “Imaging human EEG dynamics using independent component analysis,” *Neuroscience & Biobehavioral Reviews*, vol. 30, no. 6, pp. 808–822, 2006.
- [178] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero, “Shrinkage algorithms for MMSE covariance estimation,” *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5016–5029, 2010.
- [179] H. Cecotti, B. Rivet, M. Congedo, C. Jutten, O. Bertrand, E. Maby, and J. Mattout, “A robust sensor-selection method for P300 brain–computer interfaces,” *Journal of Neural Engineering*, vol. 8, no. 1, p. 016001, 2011.
- [180] B. N. Parlett, *The symmetric eigenvalue problem*. University City, Philadelphia, United States: SIAM, 1998.

- [181] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.
- [182] B. F. O’Donnell and R. A. Cohen, “The N2-P3 complex of the evoked potential and human performance,” *NASA Technical Reports*, pp. 269–286, 1988.
- [183] Y. Yang, S. Chevallier, J. Wiart, and I. Bloch, “Automatic selection of the number of spatial filters for motor-imagery bci,” in *The Proceeding of 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges, Belgium, 2012, pp. 109–114.
- [184] L. Hu, A. Mouraux, Y. Hu, and G. D. Iannetti, “A novel approach for enhancing the signal-to-noise ratio and detecting automatically event-related potentials (ERPs) in single trials,” *Neuroimage*, vol. 50, no. 1, pp. 99–111, 2010.
- [185] S. Debener, A. Strobel, B. Sorger, J. Peters, C. Kranczioch, A. K. Engel, and R. Goebel, “Improved quality of auditory event-related potentials recorded simultaneously with 3-T fMRI: Removal of the ballistocardiogram artefact,” *Neuroimage*, vol. 34, no. 2, pp. 587–597, 2007.
- [186] T. C. Handy, *Event-related potentials: A methods handbook*. Third floor of 1 Rogers Street in Cambridge, MA 02142, Boston: MIT press, 2005.
- [187] Yu, Ke and Shen, Kaiquan and Shao, Shiyun and Ng, Wu Chun and Kwok, Kenneth and Li, Xiaoping, “A spatio-temporal filtering approach to denoising of single-trial ERP in rapid image triage,” *Journal of Neuroscience Methods*, vol. 204, no. 2, pp. 288–295, 2012.
- [188] R. Hari and R. Salmelin, “Human cortical oscillations: A neuromagnetic view through the skull,” *Trends in Neurosciences*, vol. 20, no. 1, pp. 44–49, 1997.
- [189] H. Park, “An introduction to logistic regression: From basic concepts to interpretation with particular attention to nursing domain,” *Journal of Korean Academy of Nursing*, vol. 43, no. 2, pp. 154–164, 2013.
- [190] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller, “Optimal spatial filtering of single trial EEG during imagined hand movement,” *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 4, pp. 441–446, 2000.
- [191] H. Lu, H.-L. Eng, C. Guan, K. N. Plataniotis, and A. N. Venetsanopoulos, “Regularized common spatial pattern with aggregation for EEG classification in small-sample setting,” *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 12, pp. 2936–2946, 2010.
- [192] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlogl, B. Obermaier, and M. Pregenzer, “Current trends in graz brain-computer interface (BCI) research,” *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 216–219, 2000.
- [193] M.-Y. Liu and O. Tuzel, “Coupled generative adversarial networks,” in *Advances in Neural Information Processing Systems*, 2016, pp. 469–477.

- [194] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.
- [195] M. O. Turkoglu, L. Spreeuwens, W. Thong, and B. Kicanaoglu, “A layer-based sequential framework for scene generation with GANs,” in *Thirty-Third AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii, United States, 2019.
- [196] H. Wu, S. Zheng, J. Zhang, and K. Huang, “GP-GAN: Towards realistic high-resolution image blending,” *arXiv preprint arXiv:1703.07195*, 2017.
- [197] J. Pan, C. C. Ferrer, K. McGuinness, N. E. O’Connor, J. Torres, E. Sayrol, and X. Giro-i Nieto, “SalGAN: Visual saliency prediction with generative adversarial networks,” *arXiv preprint arXiv:1701.01081*, 2017.
- [198] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, “Training generative neural networks via maximum mean discrepancy optimization,” *arXiv preprint arXiv:1505.03906*, 2015.
- [199] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, “Pose guided person image generation,” in *Advances in Neural Information Processing Systems*, 2017, pp. 406–416.
- [200] C. Vondrick, H. Pirsiavash, and A. Torralba, “Generating videos with scene dynamics,” in *Advances in Neural Information Processing Systems*, 2016, pp. 613–621.
- [201] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, “High-resolution image inpainting using multi-scale neural patch synthesis,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6721–6729.
- [202] A. Odena, C. Olah, and J. Shlens, “Conditional image synthesis with auxiliary classifier gans,” in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 2642–2651.
- [203] Y. Li, K. Swersky, and R. Zemel, “Generative moment matching networks,” in *International Conference on Machine Learning*, 2015, pp. 1718–1727.
- [204] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, “Generative visual manipulation on the natural image manifold,” in *European Conference on Computer Vision*. Springer, 2016, pp. 597–613.
- [205] C. Lassner, G. Pons-Moll, and P. V. Gehler, “A generative model of people in clothing,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 853–862.
- [206] W. Fedus, I. Goodfellow, and A. M. Dai, “MaskGAN: Better text generation via filling in the _.” *arXiv preprint arXiv:1801.07736*, 2018.
- [207] Z. Yang, J. Hu, R. Salakhutdinov, and W. Cohen, “Semi-supervised QA with generative domain-adaptive nets,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, 2017, pp. 1040–1050.

- [208] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, “Good semi-supervised learning that requires a bad GAN,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6510–6520.
- [209] N. Jetchev, U. Bergmann, and R. Vollgraf, “Texture synthesis with spatial generative adversarial networks,” *arXiv preprint arXiv:1611.08207*, 2016.
- [210] C. Donahue, J. McAuley, and M. Puckette, “Synthesizing audio with generative adversarial networks,” *arXiv preprint arXiv:1802.04208*, 2018.
- [211] C. Esteban, S. L. Hyland, and G. Rätsch, “Real-valued (medical) time series generation with recurrent conditional GANs,” *arXiv preprint arXiv:1706.02633*, 2017.
- [212] D. Li, D. Chen, L. Shi, B. Jin, J. Goh, and S.-K. Ng, “MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks,” *arXiv preprint arXiv:1901.04997*, 2019.
- [213] E. Brophy, Z. Wang, and T. E. Ward, “Quick and easy time series generation with established image-based GANs,” *arXiv preprint arXiv:1902.05624*, 2019.
- [214] W. Zhu, X. Xiang, T. D. Tran, and X. Xie, “Adversarial deep structural networks for mammographic mass segmentation,” *arXiv preprint arXiv:1612.05970*, 2016.
- [215] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, “Semantic segmentation using adversarial networks,” *arXiv preprint arXiv:1611.08408*, 2016.
- [216] H. Dong, S. Yu, C. Wu, and Y. Guo, “Semantic image synthesis via adversarial learning,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5706–5714.
- [217] Z. Qiu, Y. Pan, T. Yao, and T. Mei, “Deep semantic hashing with generative adversarial networks,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2017, pp. 225–234.
- [218] N. Souly, C. Spampinato, and M. Shah, “Semi supervised semantic segmentation using generative adversarial network,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5688–5696.
- [219] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” *arXiv preprint arXiv:1812.04948*, 2018.
- [220] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, “High-resolution image synthesis and semantic manipulation with conditional GANs,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8798–8807.
- [221] B. Poole, A. A. Alemi, J. Sohl-Dickstein, and A. Angelova, “Improved generator objectives for GANs,” *arXiv preprint arXiv:1612.02780*, 2016.
- [222] J. Choe, S. Park, K. Kim, J. Hyun Park, D. Kim, and H. Shim, “Face generation for low-shot learning using generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1940–1948.

- [223] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, “StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5907–5915.
- [224] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” *arXiv preprint arXiv:1703.10593v6*, 2017.
- [225] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, “Toward multimodal image-to-image translation,” in *Advances in Neural Information Processing Systems*, 2017, pp. 465–476.
- [226] M. Tomei, M. Cornia, L. Baraldi, and R. Cucchiara, “Art2Real: Unfolding the reality of artworks via semantically-aware image-to-image translation,” *arXiv preprint arXiv:1811.10666*, 2018.
- [227] M.-Y. Liu, T. Breuel, and J. Kautz, “Unsupervised image-to-image translation networks,” in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.
- [228] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.
- [229] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, “StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.
- [230] S. Ma, J. Fu, C. Wen Chen, and T. Mei, “DA-GAN: Instance-level image translation by deep attention generative adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5657–5666.
- [231] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, “Photo-realistic single image super-resolution using a generative adversarial network,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017, pp. 105–114.
- [232] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, “ESRGAN: Enhanced super-resolution generative adversarial networks,” in *European Conference on Computer Vision Workshop*, 2018.
- [233] D. Mahapatra, B. Bozorgtabar, S. Hewavitharanage, and R. Garnavi, “Image super resolution using generative adversarial networks and local saliency maps for retinal image analysis,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 382–390.
- [234] D. Mahapatra, B. Bozorgtabar, and R. Garnavi, “Image super-resolution using progressive generative adversarial networks for medical image analysis,” *Computerized Medical Imaging and Graphics*, vol. 71, pp. 30–39, 2019.
- [235] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, “Generative image inpainting with contextual attention,” *arXiv preprint arXiv:1801.07892*, 2018.

- [236] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, “Semantic image inpainting with deep generative models,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5485–5493.
- [237] B. Dolhansky and C. Canton Ferrer, “Eye in-painting with exemplar generative adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7902–7911.
- [238] Z. Chen, S. Nie, T. Wu, and C. G. Healey, “High resolution face completion with multiple controllable attributes via fully end-to-end progressive generative adversarial networks,” *arXiv preprint arXiv:1801.07632*, 2018.
- [239] Y. Li, S. Liu, J. Yang, and M.-H. Yang, “Generative face completion,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3911–3919.
- [240] J. Kossaifi, L. Tran, Y. Panagakis, and M. Pantic, “GANGAN: Geometry-aware generative adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 878–887.
- [241] Q. Dai, Q. Li, J. Tang, and D. Wang, “Adversarial network embedding,” in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [242] N. Kodali, J. Abernethy, J. Hays, and Z. Kira, “On convergence and stability of GANs,” *arXiv preprint arXiv:1705.07215*, 2017.
- [243] Y. Li, A. Schwing, K.-C. Wang, and R. Zemel, “Dualing GANs,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5606–5616.
- [244] A. Borji, “Pros and cons of GAN evaluation measures,” *Computer Vision and Image Understanding*, vol. 179, pp. 41–65, 2019.
- [245] Q. Xu, G. Huang, Y. Yuan, C. Guo, Y. Sun, F. Wu, and K. Weinberger, “An empirical study on evaluation metrics of generative adversarial networks,” *arXiv preprint arXiv:1806.07755*, 2018.
- [246] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs trained by a two time-scale update rule converge to a local nash equilibrium,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6626–6637.
- [247] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, “A kernel two-sample test,” *Journal of Machine Learning Research*, vol. 13, no. Mar, pp. 723–773, 2012.
- [248] Z. Wang, G. Healy, A. F. Smeaton, and T. E. Ward, “Use of neural signals to evaluate the quality of generative adversarial network performance in facial image generation,” *Cognitive Computation*, Aug 2019.
- [249] Z. Wang, Q. She, A. F. Smeaton, T. E. Ward, and G. Healy, “Neuroscore: A brain-inspired evaluation metric for generative adversarial networks,” *arXiv preprint arXiv:1905.04243*, 2019.

- [250] S. Barratt and R. Sharma, “A note on the inception score,” *arXiv preprint arXiv:1801.01973*, 2018.
- [251] L. Theis, A. v. d. Oord, and M. Bethge, “A note on the evaluation of generative models,” *arXiv preprint arXiv:1511.01844*, 2015.
- [252] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, “LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop,” *arXiv preprint arXiv:1506.03365*, 2015.
- [253] Z. Liu, P. Luo, X. Wang, and X. Tang, “Large-scale celebfaces attributes (CelebA) dataset,” *Retrieved August*, vol. 15, p. 2018, 2018.
- [254] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images,” Citeseer, Tech. Rep., 2009.
- [255] Y. Yoshida and T. Miyato, “Spectral norm regularization for improving the generalizability of deep learning,” *arXiv preprint arXiv:1705.10941*, 2017.
- [256] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.
- [257] D. Berthelot, T. Schumm, and L. Metz, “BEGAN: Boundary equilibrium generative adversarial networks,” *arXiv preprint arXiv:1703.10717*, 2017.
- [258] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of GANs for improved quality, stability, and variation,” *arXiv preprint arXiv:1710.10196*, 2017.
- [259] S. Iizuka, E. Simo-Serra, and H. Ishikawa, “Globally and locally consistent image completion,” *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 107, 2017.
- [260] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text to image synthesis,” *arXiv preprint arXiv:1605.05396*, 2016.
- [261] E. L. Denton, S. Chintala, R. Fergus *et al.*, “Deep generative image models using a laplacian pyramid of adversarial networks,” in *Advances in Neural Information Processing Systems*, 2015, pp. 1486–1494.
- [262] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” *arXiv preprint arXiv:1411.1784*, 2014.
- [263] P. Burt and E. Adelson, “The Laplacian pyramid as a compact image code,” *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.
- [264] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.
- [265] J. Zhao, M. Mathieu, and Y. LeCun, “Energy-based generative adversarial network,” *arXiv preprint arXiv:1609.03126*, 2016.
- [266] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, “Progressive neural networks,” *arXiv preprint arXiv:1606.04671*, 2016.

- [267] A. Brock, J. Donahue, and K. Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” *arXiv preprint arXiv:1809.11096*, 2018.
- [268] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [269] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-attention generative adversarial networks,” *arXiv preprint arXiv:1805.08318*, 2018.
- [270] M. Arjovsky and L. Bottou, “Towards principled methods for training generative adversarial networks,” *arXiv preprint arXiv:1701.04862*, 2017.
- [271] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” *arXiv preprint arXiv:1701.07875*, 2017.
- [272] Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.
- [273] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, “Improved training of Wasserstein GANs,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5767–5777.
- [274] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks,” in *2017 IEEE International Conference on Computer Vision*. IEEE, 2017, pp. 2813–2821.
- [275] S. Nowozin, B. Cseke, and R. Tomioka, “f-GAN: Training generative neural samplers using variational divergence minimization,” in *Advances in Neural Information Processing Systems*, 2016, pp. 271–279.
- [276] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, “Unrolled generative adversarial networks,” *arXiv preprint arXiv:1611.02163*, 2016.
- [277] G.-J. Qi, “Loss-sensitive generative adversarial networks on lipschitz densities,” *arXiv preprint arXiv:1701.06264*, 2017.
- [278] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, “Mode regularized generative adversarial networks,” *arXiv preprint arXiv:1612.02136*, 2016.
- [279] J. H. Lim and J. C. Ye, “Geometric GAN,” *arXiv preprint arXiv:1705.02894*, 2017.
- [280] A. Jolicœur-Martineau, “The relativistic discriminator: A key element missing from standard GAN,” *arXiv preprint arXiv:1807.00734*, 2018.
- [281] B. K. Sriperumbudur, K. Fukumizu, A. Gretton, B. Schölkopf, and G. R. Lanckriet, “On integral probability metrics, ϕ -divergences and binary classification,” *arXiv preprint arXiv:0901.2698*, 2009.
- [282] A. Müller, “Integral probability metrics and their generating classes of functions,” *Advances in Applied Probability*, vol. 29, no. 2, pp. 429–443, 1997.
- [283] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, “Spectral normalization for generative adversarial networks,” *arXiv preprint arXiv:1802.05957*, 2018.

- [284] T. Donchev and E. Farkhi, “Stability and euler approximation of one-sided lipschitz differential inclusions,” *SIAM Journal on Control and Optimization*, vol. 36, no. 2, pp. 780–796, 1998.
- [285] L. Armijo, “Minimization of functions having lipschitz continuous first partial derivatives,” *Pacific Journal of Mathematics*, vol. 16, no. 1, pp. 1–3, 1966.
- [286] A. Goldstein, “Optimization of lipschitz continuous functions,” *Mathematical Programming*, vol. 13, no. 1, pp. 14–22, 1977.
- [287] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.
- [288] Y. Luo, X. Cai, Y. Zhang, J. Xu *et al.*, “Multivariate time series imputation with generative adversarial networks,” in *Advances in Neural Information Processing Systems*, 2018, pp. 1596–1607.
- [289] L. Yu, W. Zhang, J. Wang, and Y. Yu, “SeqGAN: Sequence generative adversarial nets with policy gradient,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [290] D. Bahdanau, P. Brakel, K. Xu, A. Goyal, R. Lowe, J. Pineau, A. Courville, and Y. Bengio, “An actor-critic algorithm for sequence prediction,” *arXiv preprint arXiv:1607.07086*, 2016.
- [291] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, “Deep reinforcement learning for dialogue generation,” *arXiv preprint arXiv:1606.01541*, 2016.
- [292] H. A. Abbass, “Social integration of artificial intelligence: Functions, automation allocation logic and human-autonomy trust,” *Cognitive Computation*, vol. 11, pp. 159–171, April 2019.
- [293] Z. G. Doborjeh, M. G. Doborjeh, and N. Kasabov, “Attentional bias pattern recognition in spiking neural networks from spatio-temporal EEG data,” *Cognitive Computation*, vol. 10, no. 1, pp. 35–48, 2018.
- [294] J. Li, Z. Zhang, and H. He, “Hierarchical convolutional neural networks for EEG-based emotion recognition,” *Cognitive Computation*, vol. 10, pp. 1–13, 2018.
- [295] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, “Learning from simulated and unsupervised images through adversarial training,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, vol. 2, no. 4. IEEE, 2017, p. 5.
- [296] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.
- [297] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, “TensorFlow: A system for large-scale machine learning,” in *12th Symposium on Operating Systems Design and Implementation*, 2016, pp. 265–283.

- [298] A. Gulli and S. Pal, *Deep Learning with Keras*. Livery Place, 35 Livery Street, Birmingham B3 2PB, United Kingdom: Packt Publishing Ltd, 2017.
- [299] N. Ketkar, “Introduction to PyTorch,” in *Deep learning with python*. Salmon Tower Building, New York City, United States: Springer, 2017, pp. 195–208.
- [300] D. A. Forsyth and J. Ponce, *Computer vision: A Modern Approach*. Upper Saddle River, New Jersey, United States: Prentice-Hall, 2003.
- [301] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap*. Boca Raton, Florida, United States: CRC press, 1994.
- [302] J. Z. Bakdash and L. R. Marusich, “Repeated measures correlation,” *Frontiers in Psychology*, vol. 8, p. 456, 2017.
- [303] K. H. Kim, J. H. Kim, J. Yoon, and K.-Y. Jung, “Influence of task difficulty on the features of event-related potential during visual oddball task,” *Neuroscience Letters*, vol. 445, no. 2, pp. 179–183, 2008.
- [304] S. J. Luck and S. A. Hillyard, “Electrophysiological evidence for parallel and serial processing during visual search,” *Perception & Psychophysics*, vol. 48, no. 6, pp. 603–617, 1990.
- [305] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” *arXiv preprint arXiv:1607.02533*, 2016.
- [306] Z. Cai, S. Makino, and T. M. Rutkowski, “Brain evoked potential latencies optimization for spatial auditory brain–computer interface,” *Cognitive Computation*, vol. 7, no. 1, pp. 34–43, 2015.
- [307] P. W. Koh and P. Liang, “Understanding black-box predictions via influence functions,” in *International Conference on Machine Learning*, 2017, pp. 1885–1894.
- [308] D. L. Yamins and J. J. DiCarlo, “Using goal-driven deep learning models to understand sensory cortex,” *Nature Neuroscience*, vol. 19, no. 3, p. 356, 2016.
- [309] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *2018 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2018, pp. 4510–4520.
- [310] S. Waldert, “Invasive vs. non-invasive neuronal signals for brain-machine interfaces: Will one prevail?” *Frontiers in Neuroscience*, vol. 10, p. 295, 2016.
- [311] N. Lago and A. Cester, “Flexible and organic neural interfaces: A review,” *Applied Sciences*, vol. 7, no. 12, p. 1292, 2017.
- [312] H. Jasper and W. Penfield, “Electrocorticograms in man: Effect of voluntary movement upon the electrical activity of the precentral gyrus,” *Archive for Psychiatrie und Nervenkrankheiten*, vol. 183, no. 1-2, pp. 163–174, 1949.
- [313] V. N. Murthy and E. E. Fetz, “Synchronization of neurons during local field potential oscillations in sensorimotor cortex of awake monkeys,” *Journal of Neurophysiology*, vol. 76, no. 6, pp. 3968–3982, 1996.

- [314] R. M. Cichy, A. Khosla, D. Pantazis, A. Torralba, and A. Oliva, “Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence,” *Scientific Reports*, vol. 6, p. 27755, 2016.
- [315] R. M. Cichy and D. Kaiser, “Deep neural networks as scientific models,” *Trends in Cognitive Sciences*, 2019.
- [316] I. I. Groen, M. R. Greene, C. Baldassano, L. Fei-Fei, D. M. Beck, and C. I. Baker, “Distinct contributions of functional and deep neural network features to representational similarity of scenes in human brain and behavior,” *Elife*, vol. 7, p. e32962, 2018.
- [317] I. Kuzovkin, R. Vicente, M. Petton, J.-P. Lachaux, M. Baciú, P. Kahane, S. Rheims, J. R. Vidal, and J. Aru, “Activations of deep convolutional neural networks are aligned with gamma band activity of human visual cortex,” *Communications Biology*, vol. 1, no. 1, p. 107, 2018.
- [318] T. Tu, J. Koss, and P. Sajda, “Relating deep neural network representations to EEG-fMRI spatiotemporal dynamics in a perceptual decision-making task,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1985–1991.
- [319] A. P. Batista and J. J. DiCarlo, “Deep learning reaches the motor system,” *Nature Methods*, vol. 15, no. 10, p. 772, 2018.
- [320] N. Kriegeskorte, “Deep neural networks: A new framework for modeling biological vision and brain information processing,” *Annual Review of Vision Science*, vol. 1, pp. 417–446, 2015.
- [321] S. R. Kheradpisheh, M. Ghodrati, M. Ganjtabesh, and T. Masquelier, “Deep networks can resemble human feed-forward vision in invariant object recognition,” *Scientific Reports*, vol. 6, p. 32672, 2016.
- [322] L. Chelazzi, E. K. Miller, J. Duncan, and R. Desimone, “A neural basis for visual search in inferior temporal cortex,” *Nature*, vol. 363, no. 6427, p. 345, 1993.
- [323] Q. She, Y. Gao, K. Xu, and R. H. Chan, “Reduced-rank linear dynamical systems,” in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [324] G. Gargiulo, R. A. Calvo, P. Bifulco, M. Cesarelli, C. Jin, A. Mohamed, and A. van Schaik, “A new EEG recording system for passive dry electrodes,” *Clinical Neurophysiology*, vol. 121, no. 5, pp. 686–693, 2010.
- [325] A. Matran-Fernandez and R. Poli, “Brain–computer interfaces for detection and localization of targets in aerial images,” *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 4, pp. 959–969, 2016.
- [326] Y. Gao, R. Singh, and B. Raj, “Voice impersonation using generative adversarial networks,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2506–2510.
- [327] C. Donahue, J. McAuley, and M. Puckette, “Adversarial audio synthesis,” *arXiv preprint arXiv:1802.04208*, 2018.

- [328] J. Engel, K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts, “GANSynth: Adversarial neural audio synthesis,” *arXiv preprint arXiv:1902.08710*, 2019.
- [329] G. Hickok and D. Poeppel, “Dorsal and ventral streams: a framework for understanding aspects of the functional anatomy of language,” *Cognition*, vol. 92, no. 1-2, pp. 67–99, 2004.
- [330] T. Eichele, K. Specht, M. Moosmann, M. L. Jongsma, R. Q. Quiroga, H. Nordby, and K. Hugdahl, “Assessing the spatiotemporal evolution of neuronal activation with single-trial event-related potentials and functional MRI,” *Proceedings of the National Academy of Sciences*, vol. 102, no. 49, pp. 17 798–17 803, 2005.
- [331] E. Halgren, “Considerations in source estimation of the P3,” *Event-related Potentials in Patients with Epilepsy: From Current State to Future Prospects*, p. 71, 2008.
- [332] C. Bledowski, D. Prvulovic, K. Hoechstetter, M. Scherg, M. Wibrall, R. Goebel, and D. E. Linden, “Localizing P300 generators in visual target and distractor processing: A combined event-related potential and functional magnetic resonance imaging study,” *Journal of Neuroscience*, vol. 24, no. 42, pp. 9353–9360, 2004.
- [333] C.-G. Bénar, D. Schön, S. Grimault, B. Nazarian, B. Burle, M. Roth, J.-M. Badier, P. Marquis, C. Liegeois-Chauvel, and J.-L. Anton, “Single-trial analysis of oddball event-related potentials in simultaneous EEG-fMRI,” *Human Brain Mapping*, vol. 28, no. 7, pp. 602–613, 2007.
- [334] D. E. Linden, “The P300: Where in the brain is it produced and what does it tell us?” *The Neuroscientist*, vol. 11, no. 6, pp. 563–576, 2005.
- [335] Y. Huang, Y. Cheng, D. Chen, H. Lee, J. Ngiam, Q. V. Le, and Z. Chen, “GPipe: Efficient training of giant neural networks using pipeline parallelism,” *arXiv preprint arXiv:1811.06965*, 2018.
- [336] C. Kothe. (1st Feb 2013) SNAP. [Online]. Available: <https://github.com/sccn/SNAP>.

Appendix

A Investigation of Triggering Issue

To investigate whether or not software-derived stimulus timing can be considered an accurate reflection of the physical stimuli timing in a representative RSVP implementation, we designed a simple circuit consisting of a light diode resistor comparator circuit (LDRCC), as seen in Fig. A.1, for recording the physical presentation of stimuli and which in turn generates what we

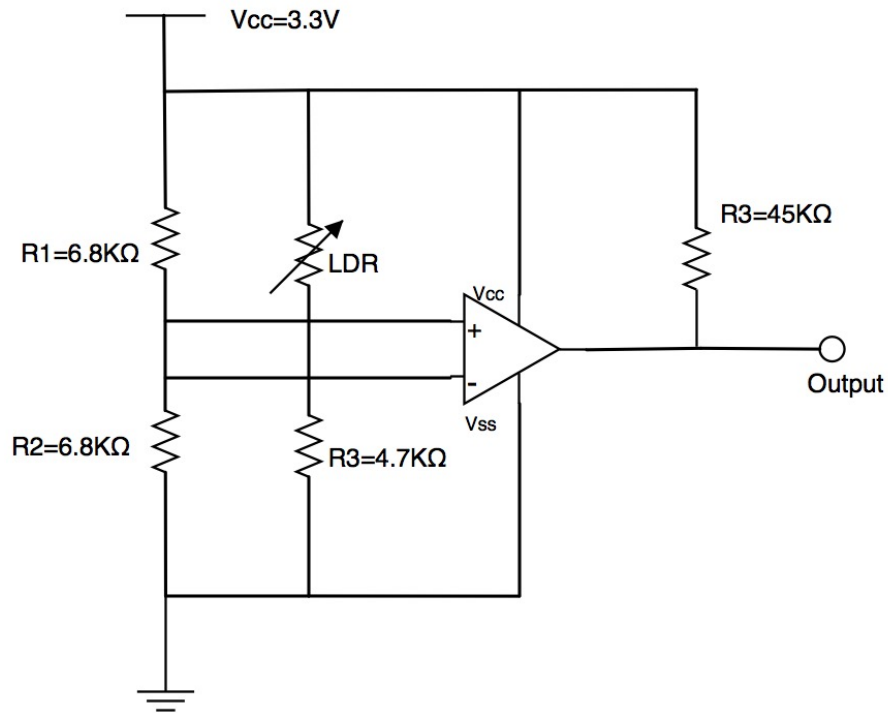


Figure A.1: LDRCC architecture.

refer to as hardware triggered events. The presentation machine used in this benchmarking test was a Dell XPS 8700 desktop which uses a 4th generation Intel core i7-4790 processor, AMD Radeon HD R9 270 2GB GDDR5 video card and Dell 2313H monitor.

Hardware Trigger Acquisition In order to generate hardware triggers in a RSVP experiment, the light diode resistor (LDR) in LDRCC was taped to the top-left corner of the presentation monitor. Black and white images started to appear alternately in that region i.e., the black and white were superimposed on the upper corner of each image stimulus. When the presented image changes each time, the LDRCC generates a rising or falling edge, which marks the onset of the next image. There are two reasons for choosing only black and white images in that region: (1) The first reason is that it gives the same light change to the LDR independently of the image presented; and (2) The second reason is that the light change between while and black images is larger than between any other two colors. Hence, this light change enables the largest change in the resistance of the LDR, which in turn makes the largest voltage change at the positive input of comparator. It is faster for comparator to generate triggers for the changing of presented images. Hardware triggers were then sent to a BioSemi Active View 2 system. The LDRCC output was connected to pin 16 of the trigger IO at the back of the BioSemi stimulus box and the other pins were connected to ground. Capturing both EEG and triggers in this way on a single data acquisition device allows for the highest precision in timing align [69].

Software Trigger Acquisition Image presentation and software trigger generation were implemented by using the simulation and neuroscience application platform (SNAP) [336] in Python. Software triggers were generated directly prior to the execution of image presentation code. Triggers were sent to lab streaming layer (LSL) [121] via SNAP.

Figure A.2 shows an example of 20 hardware and software triggers generated from 20 pre-

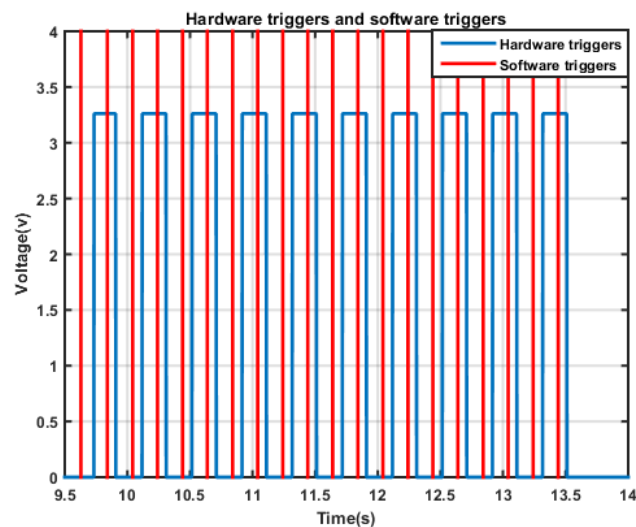


Figure A.2: Captured hardware and software triggers.

A. Investigation of Triggering Issue

sented images. The rate of image streaming was 5 Hz (time interval between each image is 0.2 s). The blue line is the hardware trigger signal while the red line is the software trigger signal. It can be seen that the hardware trigger signal is a square wave where the high voltage is 3.3 V and low voltage is 0 V. The amplitude of the software trigger does not have any meaning in terms of experimental interpretation. The default value of LDRCC output is 0 V at the beginning and it can be seen that the software trigger precedes the hardware trigger in this case.

The latency between image presentation in software and physical image presentation was attained by $t_h - t_s$, where t_h and t_s are hardware and software timestamps. Figure A.3 shows histograms of these latency values for 50, 100, 200, 500 and 1000 images group cases. The reason we did this was to assess differences in recorded presentation times in software and physical image presentation times. We calculated statistical characteristics of each distribution (see Table A.1). As the distributions seen in Fig. A.3 appear non-gaussian (particularly for group 50 &

Group	50 images	100 images	200 images	500 images	1000 images
Median (s)	0.0255	0.0250	0.0295	0.0369	0.0638
First 50 points (s)	0.0255	0.0270	0.0250	0.0360	0.0520
Last 50 points (s)	0.0255	0.0240	0.0300	0.0410	0.0630

Table A.1: Time-related latencies between image presentation in software and physical image presentation of different groups.

100), we used a median statistic for reporting in Table A.1. Examining Table A.1 and Fig. A.3 in tandem we can see that increasing image numbers negatively impacts (i.e., increases) our median latencies. These statistical characteristics show that the first two groups have smaller latency errors between image presentation in software and physical image presentation compared to the last three groups. The increasing latency encountered for increasing image count is in all likelihood caused by the software implementation. When implementing RSVP experiments in some software for large datasets, it is necessary to make efforts such as pre-buffering images into memory and/or unallocating memory for each loaded image after it is presented. Without employing such efforts, the presentation software may exhibit issues such as slowing presentation speed as each image is loaded but not removed from memory after presentation. Such overheads in turn can cause other operating system functionality and/or network functionality to be impeded, giving rise to a range of other complex effects that can in turn potentially affect timing characteristics of the presentation software further. We calculated the median latency values of the first 50 points and the last 50 points for 100, 200, 500 and 1000 image groups in

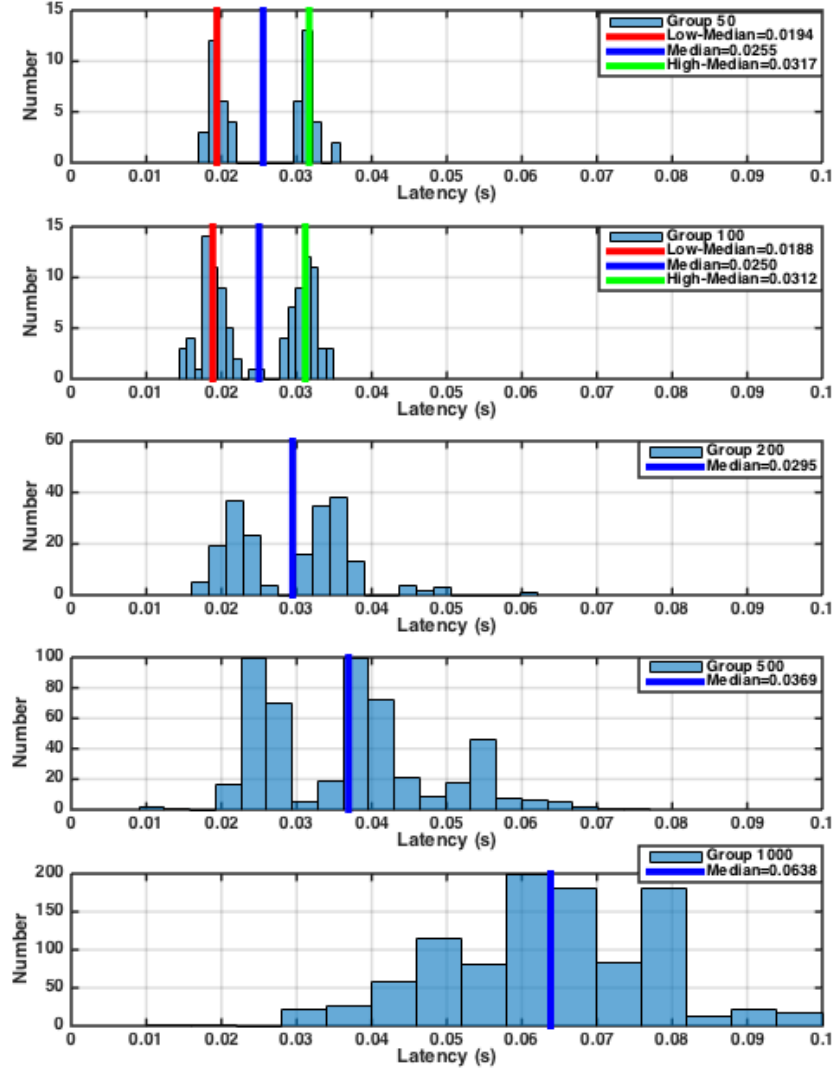


Figure A.3: Histograms of latencies derived from (paired) differences between hardware and software trigger timestamps (hardware timestamp - software timestamp), as a way to assess timing differences arising due to issues such as those caused in software implementation. We showed distributions for groups of 50, 100, 200, 500 and 1000 images (respectively row by row). Shown in blue vertical lines are median values. In the first two histograms (i.e., group 50/group 100) in red and green we can see vertical lines corresponding to median values for lower and upper ranges after a median split.

order to see whether potential software problems were causing larger latencies with increasing time. From the last three columns in Table A.1, it can be seen that such software implementation problems are the cause of larger latencies in the group of 200 images, 500 images and 1000 images but the 100-image group does not suffer from these types of software problems. Therefore, we concluded that the median difference of 0.025 s is a realistic approximation of

the real difference between image presentation in software and physical image presentation in this RSVP implementation for our system.

Notably, however, in Fig. A.3 we can see that group 50, group 100 and group 200 are bimodal distributions. In order to evaluate what might be causing this we (1) used a median split to firstly divide the latencies into lower and upper ranges, (2) applied further median splits to these two new ranges for group 50 and group 100, and (3) calculated the difference between these respective upper and lower median splits. This can be seen in Fig. A.3 for group 50 and group 100 where the lower median split is in red and the upper median split is in green. What we find is that there is a 0.0121 second and 0.0124 second difference between these upper and lower medians for group 50 and group 100 respectively. In effect, we can say there is an additional latency affecting half of our trigger samples that is between 0.0121 second and 0.0124 second.

In Fig. A.4 for the group 100 case (where we examined time intervals for both hardware and

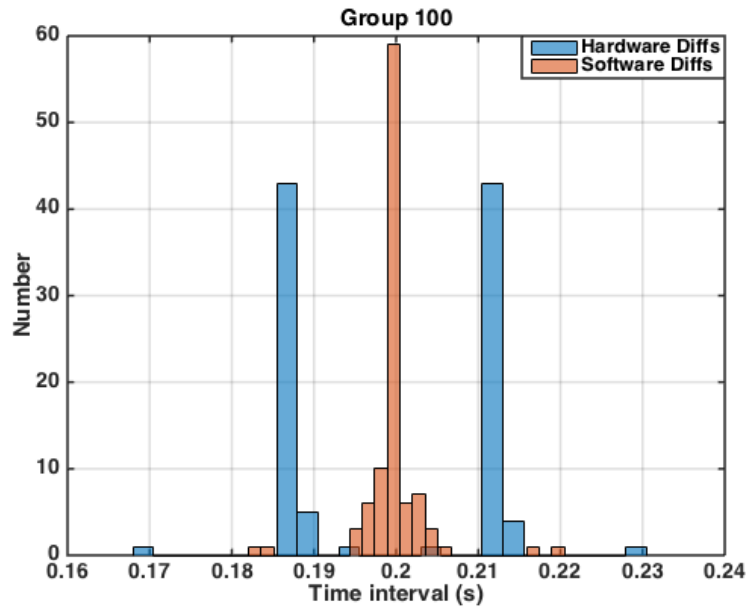


Figure A.4: Distribution of interval differences in timestamps for hardware triggers (in blue) and software triggers (in orange).

software triggers), we can see that there is relative stability to the frequency of software triggers where hardware-sensed triggers are seen to be more variable (and bimodal). As we were using the difference of these relative timestamps to generate Fig. A.3, we identify this as causing the bimodal distributions we see in Fig. A.3. These differences are likely related to the refresh rate of the monitor used where there is an approximate 50% likelihood that the stimulus presentation will not happen until the next refresh. These variable timing differences (0.0121 s and 0.0124

s for Group 50/100 respectively) are relatively close to the refresh time interval of the monitor (0.0167 s i.e., 60 Hz) used in our experiment.

We investigated timing discrepancy in stimulus presentation timing when relying on software only timing information. Hardware in the form of light detection circuits were used to provide accurate timing information on stimulus presentation and this was compared to events generated in the corresponding software for a RSVP experiment. Results demonstrate that the latency exists between the image presentation in software and physical image presentation event for 50-image and 100-image group and software problems arise with increasing image datasets (i.e., starting from 200 images). It should be stressed that this is due to software problems (e.g., crippling memory overhead) of presenting images and the refresh time interval of the monitor. We suggest that for RSVP protocols where temporal accuracy is important that unless demonstrated otherwise a hardware solution for monitoring physical presentation of images should be used.

B Supplementary Tables

Participant	SNR			
	N170		P300	
	MTWLB	xDAWN	MTWLB	xDAWN
1	1.09	0.83	1.42	1.25
2	1.27	1.21	1.07	1.00
3	1.71	1.67	1.40	1.07
4	1.43	1.37	1.84	0.85
5	1.16	0.93	1.41	1.05
6	0.94	0.68	1.40	1.28
7	1.03	1.03	1.14	1.09
8	1.86	1.71	1.85	1.63
9	1.20	1.16	1.37	1.27
10	1.27	0.91	1.41	1.24
11	1.65	0.60	1.37	1.12
12	0.81	0.71	1.30	1.06
Mean (Standard deviation)	1.29 (0.32)	1.08 (0.37)	1.33 (0.24)	1.16 (0.20)

Table B.1: Details of SNR for Fig. 4.7 in Chapter 4.

Participant	AUC score			
	N170		P300	
	MTWLB	xDAWN	MTWLB	xDAWN
1	0.878	0.835	0.845	0.830
2	0.858	0.848	0.797	0.783
3	0.886	0.884	0.878	0.790
4	0.849	0.842	0.746	0.745
5	0.862	0.821	0.836	0.783
6	0.778	0.700	0.852	0.841
7	0.844	0.843	0.806	0.792
8	0.910	0.899	0.912	0.897
9	0.869	0.862	0.840	0.820
10	0.860	0.748	0.830	0.815
11	0.872	0.678	0.859	0.807
12	0.792	0.762	0.851	0.771
Mean (Standard deviation)	0.855 (0.037)	0.810 (0.071)	0.838 (0.042)	0.806 (0.039)

Table B.2: Details of AUC score for Fig. 4.8 in Chapter 4.

C Notation on Chapter 5

C.1 Lipschitz Continuity

Given a real-valued function $f: \mathbb{R} \rightarrow \mathbb{R}$, f is Lipschitz continuous *if and only if* there exists a constant $K \geq 0$ for all real x_1 and x_2

$$|f(x_1) - f(x_2)| \leq K|x_1 - x_2| \quad (8.1)$$

which holds *if and only if* the absolute value of the slopes of all secant lines are bounded by K .

C.2 Matrix Norm

We introduced spectral normalization GAN (SN-GAN) in section 5.6.10, where the weights in D are normalized by the L_2 matrix norm i.e., equation (5.22). For each layer $g: \mathbf{h}_{in} \rightarrow \mathbf{h}_{out}$, the L_2 matrix norm of \mathbf{W} is

$$\sigma(\mathbf{W}) := \max_{\mathbf{h}: \mathbf{h} \neq 0} \frac{\|\mathbf{W}\mathbf{h}\|_2}{\|\mathbf{h}\|_2} \quad (8.2)$$

which is equivalent to the largest singular value of \mathbf{W} . With spectral normalization being applied, weight matrix \mathbf{W} satisfies the Lipschitz constraint $\sigma(\mathbf{W}) = 1$. Rigorous proof can be referred to the original paper [283].

D Supplementary Figures

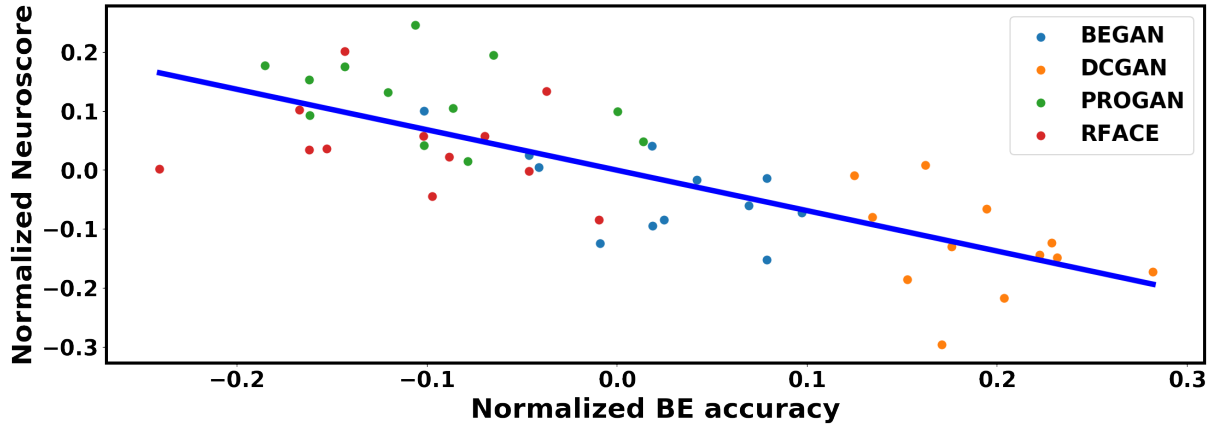


Figure D.1: Correlation between Neuroscore and BE accuracy with normalization (including RFACE category). Pearson correlation statistics is: $r(48) = -0.767, p = 2.089 \times 10^{-10}$. Bootstrapped $p \leq 0.0001$.

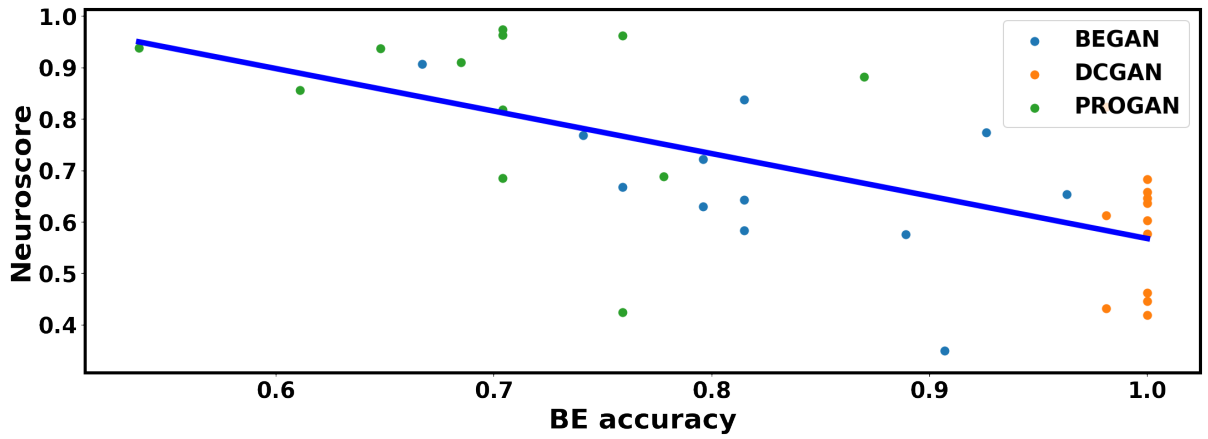


Figure D.2: Correlation between Neuroscore and BE accuracy without normalization. Pearson correlation statistics is: $r(36) = -0.649, p = 1.859 \times 10^{-5}$. Bootstrapped $p \leq 0.0001$.

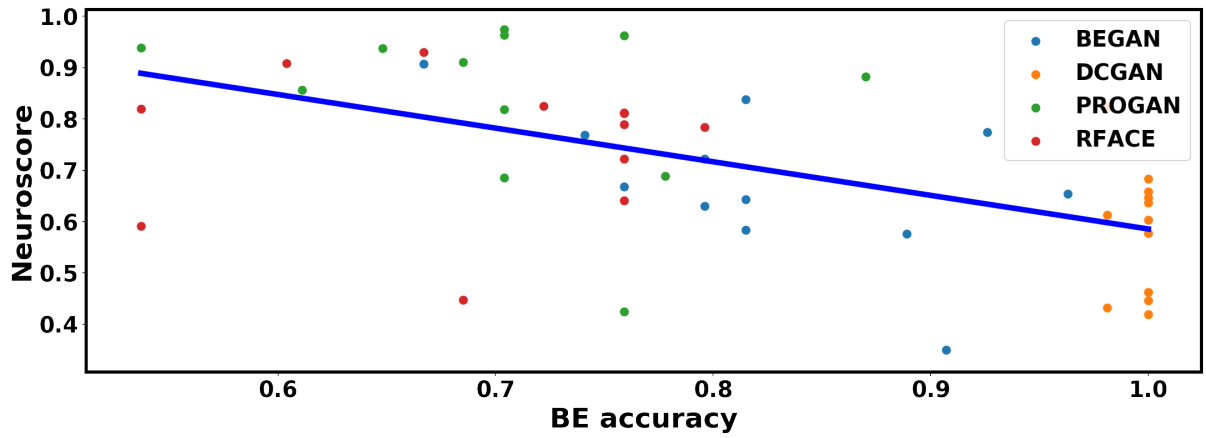
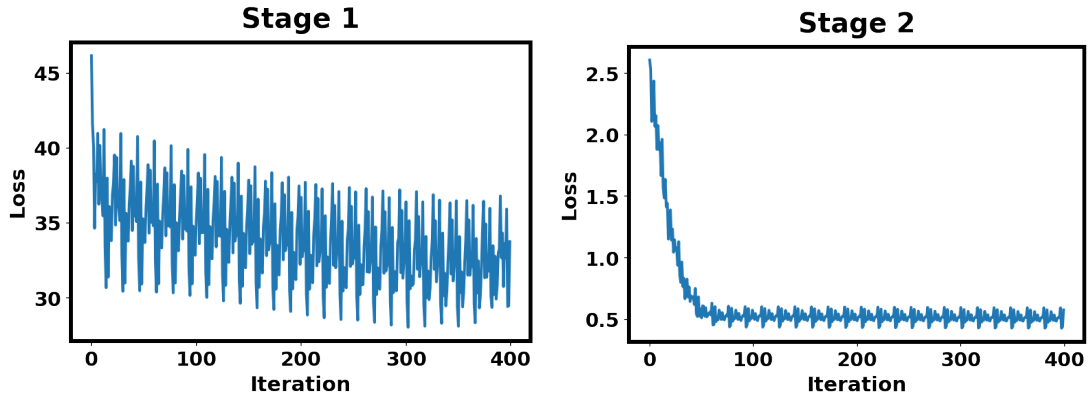
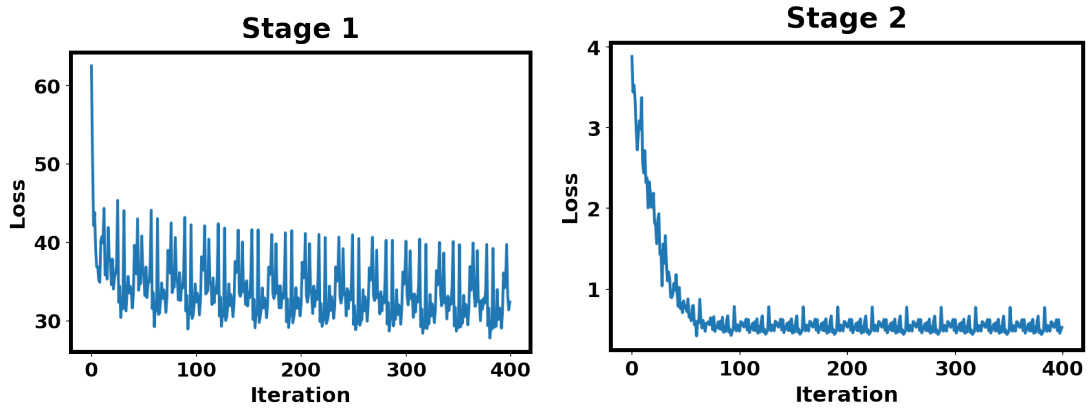


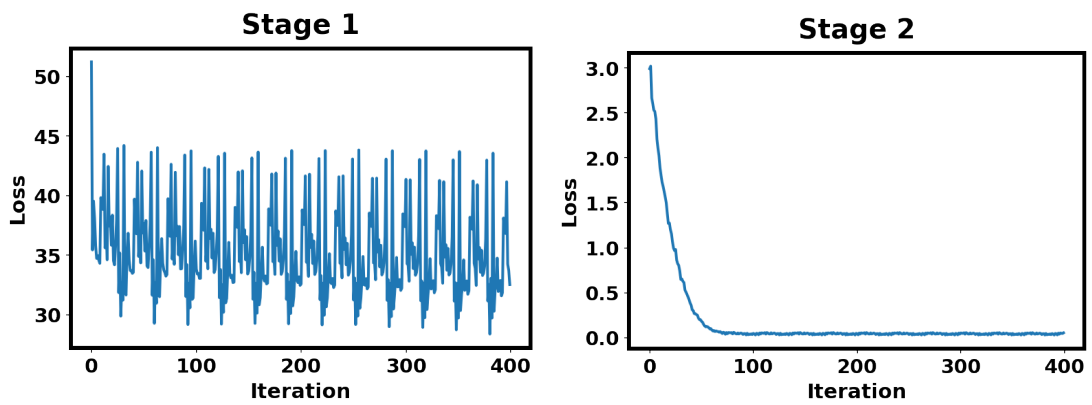
Figure D.3: Correlation between Neuroscore and BE accuracy without normalization (including RFACE category). Pearson correlation statistics is: $r(48) = -0.556, p = 4.038 \times 10^{-5}$. Bootstrapped $p \leq 0.0001$.



(a) Shallow net



(b) Mobilenet



(c) Inception

Figure D.4: Two-stage training details for Chapter 7. Training was conducted through the cross-participant model. Batch size is 256 in this case.

E NAILS Ethic Approval

Ollscoil Chathair Bhaile Átha Cliath
Dublin City University



Dr Graham Healy
School of Computing/INSIGHT

1st July 2016

REC Reference: DCUREC/2016/099

Proposal Title: Neurally Augmented Image Labelling Strategies

Applicant(s): Dr Graham Healy, Dr Cathal Gurrin & Prof Alan Smeaton

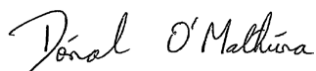
Dear Graham,

Further to expedited review, the DCU Research Ethics Committee approves this research proposal.

Materials used to recruit participants should note that ethical approval for this project has been obtained from the Dublin City University Research Ethics Committee.

Should substantial modifications to the research protocol be required at a later stage, a further amendment submission should be made to the REC.

Yours sincerely,

A handwritten signature in black ink, reading 'Dónal O'Mathúna'.

Dr Dónal O'Mathúna
Chairperson
DCU Research Ethics Committee



Taighde & Nuálaíocht Tacaíocht
Ollscoil Chathair Bhaile Átha Cliath,
Baile Átha Cliath, Éire

Research & Innovation Support
Dublin City University,
Dublin 9, Ireland

T +353 1 700 8000
F +353 1 700 8002
E research@dcu.ie
www.dcu.ie

F NIFPA Ethic Approval

Ollscoil Chathair Bhaile Átha Cliath
Dublin City University



Dr Graham Healy

School of Computing/INSIGHT

19th June 2018

REC Reference: DCUREC/2018/115

Proposal Title: Neural Indices for Face Perception Analysis (NIFPA)

Applicant(s): Dr Graham Healy, Zhengwei Wang, Professor Thomas Ward

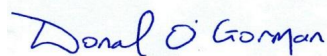
Dear Graham,

Further to expedited review, the DCU Research Ethics Committee approves this research proposal.

Materials used to recruit participants should note that ethical approval for this project has been obtained from the Dublin City University Research Ethics Committee.

Should substantial modifications to the research protocol be required at a later stage, a further amendment submission should be made to the REC.

Yours sincerely,

A handwritten signature in blue ink that reads 'Dónal O'Gorman'.

Dr Dónal O'Gorman
Chairperson
DCU Research Ethics Committee



Taighde & Nuálaíocht Tacaíocht
Ollscoil Chathair Bhaile Átha Cliath,
Baile Átha Cliath, Éire

Research & Innovation Support
Dublin City University,
Dublin 9, Ireland

T +353 1 700 8000
F +353 1 700 8002
E research@dcu.ie
www.dcu.ie