

Universidade de Brasília – UnB
Faculdade UnB Gama – FGA
Engenharia Eletrônica

**Estudo Comparativo da Implementação em
Sistemas em Chip do Algoritmo de Treinamento
LDA Aplicado em Interfaces Cérebro Máquina**

Autores: Heleno da Silva Moraes, Oziel da Silva Santos
Orientador: Dr, Marcus Vinícius Chaffim Costa

Brasília, DF
2017

Heleno da Silva Moraes, Oziel da Silva Santos

Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Dr, Marcus Vinícius Chaffim Costa

Coorientador: PhD, Daniel Mauricio Muñoz Arboleda

Brasília, DF

2017

Heleno da Silva Morais, Oziel da Silva Santos

Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina/ Heleno da Silva Morais, Oziel da Silva Santos. – Brasília, DF, 2017-

49 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr, Marcus Vinícius Chaffim Costa

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade UnB Gama – FGA , 2017.

1. Sistemas em Chip. 2. BCI. I. Dr, Marcus Vinícius Chaffim Costa. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina

CDU 02:141:005.6

Errata

Elemento opcional da ??, 4.2.1.2).

Heleno da Silva Moraes, Oziel da Silva Santos

Estudo Comparativo da Implementação em Sistemas em Chip do Algoritmo de Treinamento LDA Aplicado em Interfaces Cérebro Máquina

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Trabalho aprovado. Brasília, DF, 01 de junho de 2013 – Data da aprovação do trabalho:

Dr, Marcus Vinícius Chaffim Costa
Orientador

Titulação e Nome do Professor
Convidado 01
Convidado 1

Titulação e Nome do Professor
Convidado 02
Convidado 2

Brasília, DF
2017

**A dedicatória é opcional. Caso não deseje uma, deixar todo este arquivo em
branco.**

*Este trabalho é dedicado às crianças adultas que,
quando pequenas, sonharam em se tornar cientistas.*

Agradecimentos

A inclusão desta seção de agradecimentos é opcional, portanto, sua inclusão fica a critério do(s) autor(es), que caso deseje(em) fazê-lo deverá(ão) utilizar este espaço, seguindo a formatação de *espaço simples e fonte padrão do texto (sem negritos, aspas ou itálico)*.

Caso não deseje utilizar os agradecimentos, deixar toda este arquivo em branco.

A epígrafe é opcional. Caso não deseje uma, deixe todo este arquivo em
branco.

*“Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)*

Resumo

Palavras-chaves: BCI; LDA; FPGA; SoC; Sistemas Embarcados.

Abstract

Key-words: BCI; LDA; FPGA; SoC; Embedded Systems.

Lista de ilustrações

Figura 1 – Três principais áreas do cérebro (ALVAREZ; LEMOS, 2006)	31
--	----

Lista de tabelas

Tabela 1 – Número de tarefas rotuladas e não rotuladas por sujeito (BLANKERTZ et al., 2006).	33
--	----

Lista de abreviaturas e siglas

BCI	<i>Brain Computer Interface</i>
EEG	Eletroencefalograma
SVM	<i>Support Vector Machine</i>
LDA	<i>Linear Discriminant Analysis</i>
SoC	<i>System on Chip</i>
FPGA	<i>Field Programmable Array</i>
SNC	Sistema Nervoso Central
VHDL	<i>VHISC Hardware Description Language</i>
VHISC	<i>Very High Speed Integrated Circuit</i>
LUT	<i>Look Up Table</i>
FF	Flip-Flop
DSP	<i>Digital Signal Processor</i>
RAM	<i>Random Access Memory</i>
MUX	Multiplexador

Lista de símbolos

\in	Pertence
-------	----------

Sumário

1	INTRODUÇÃO	27
1.1	Contextualização	27
1.2	Justificativa	29
1.3	Objetivos	29
1.3.1	Objetivos Gerais	29
1.3.2	Objetivos Específicos	29
2	REFERENCIAL TEÓRICO	31
2.1	O Cérebro	31
2.2	Eletroencefalografia	32
2.3	<i>Brain Computer Interface</i>	32
2.4	<i>BCI Competition</i>	32
2.4.1	<i>BCI Competition III</i>	32
2.4.2	<i>BCI Competition III - Dataset IVa</i>	32
2.5	<i>Linear Discriminant Analysis</i>	33
2.6	<i>System-on-Chip</i>	34
2.7	Estado da Arte	34
3	PROPOSTA METODOLÓGICA	35
3.1	Implementação em hardware	35
3.1.1	Dispositivos e ferramentas	35
3.1.2	Metodologias de desenvolvimento	35
3.2	Implementação em software	36
3.3	Cronograma de Atividades	36
	REFERÊNCIAS	37
	APÊNDICES	39
	APÊNDICE A – PRIMEIRO APÊNDICE	41
	APÊNDICE B – SEGUNDO APÊNDICE	43

ANEXOS	45
ANEXO A – PRIMEIRO ANEXO	47
ANEXO B – SEGUNDO ANEXO	49

1 Introdução

Este capítulo apresenta uma visão geral do projeto, apresentando sua contextualização, assim como a proposta do presente trabalho e seus objetivos a serem alcançados.

1.1 Contextualização

Através de uma rede de mais de 100 bilhões de células nervosas interconectadas, o cérebro realiza o controle de nossas ações, percepções, emoções e etc (KANDEL, 2013). Estas células são chamadas de *neurônios*, e neles são armazenados sinais elétricos, que representam todas as informações de controle (SIULY; LI; ZHANG, 2017). Estes sinais podem ser medidos pela eletroencefalografia (EEG), que é um sistema de medição de sinais elétricos produzidos pelo cérebro durante atividades cerebrais (LOTTE; GUAN, 2011). Segundo (SIULY, 2012), a EEG é uma das mais importantes ferramentas para diagnosticar doenças cerebrais.

Além do diagnóstico de doenças cerebrais uma outra aplicação para os sinais adquiridos pela EEG são as *Brain Computer Interfaces* (BCIs) (LOTTE; GUAN, 2011). Uma BCI é um sistema que realiza a comunicação entre o cérebro e um computador (SIULY; LI; ZHANG, 2017), onde sua principal função é a tradução dos sinais elétricos, obtidos através da EEG, em comandos de controle para qualquer dispositivo eletrônico (SIULY; LI; ZHANG, 2017).

A BCI realiza a tradução destes comandos através de seis passos: 1) medição dos sinais provenientes de atividades cerebrais através da EEG, 2) pré-processamento destes sinais, 3) extração de características, 4) classificação, 5) tradução dos sinais em comandos e 6) realimentação (MASON; BIRCH, 2003). Um dos principais passos para a implementação de uma BCI é a **classificação**, pois é após este passo que é realizada a tradução dos sinais provenientes da EEG em comandos de controle (MASON; BIRCH, 2003).

A classificação de um sinal é caracterizada, em aprendizado de máquina e em reconhecimento de padrões, como um algoritmo que atribui parte de um dado sinal de entrada a um dado número de classes ou categorias (BRUNELLI, 2009). Um exemplo é a classificação de um e-mail como "spam" ou "não-spam". Os algoritmos que realizam a classificação dos sinais de entrada são chamados de **classificadores** (SIULY; LI; ZHANG, 2017). De acordo com (LOTTE, 2008, p. 41), "estes classificadores são capazes de aprender como identificar um vetor de características, graças aos processos de treinamentos". Estes conjuntos são formados por vetores de características previamente atribuídos às suas

respectivas classes (LOTTE, 2008).

O algoritmo que realiza a classificação é caracterizado por uma função matemática que mapeia um sinal de entrada em sua respectiva classe (LOTTE, 2008). Os classificadores preferidos pelos pesquisadores são os **classificadores supervisionados**, pois estes tipos de classificadores necessitam de um conjunto de dados de treinamento. Os dados de treinamento são um conjunto de dados previamente classificados (SIULY; LI; ZHANG, 2017). Portanto os classificadores supervisionados são implementados a partir de dois processos: *treinamento* e *testes* (SIULY; LI; ZHANG, 2017). As *Support Vector Machines* (SVM), os *Linear Discriminant Analysis* (LDA), os filtros Kalman, as árvores de decisões são alguns exemplos de classificadores do tipo supervisionados (SIULY; LI; ZHANG, 2017).

O LDA como dito anteriormente é um dos classificadores supervisionados e tem como suas principais vantagens a simplicidade e atratividade computacional, por se tratar de um classificador linear (THEODORIDIS; KOUTROUMBAS et al., 1999). O objetivo do LDA é usar uma transformação linear para encontrar um conjunto otimizado de vetores discriminantes e remapear o conjunto de características original, em um outro conjunto de dimensão inferior (SHASHOA et al., 2016).

Apesar do LDA ser um algoritmo interessante no que se trata de consumo computacional, em geral, os algoritmos de classificação são complexos computacionalmente com tempo de execução elevado, isso os tornam restritivos a poucas aplicações, para executar um código que descreve esses algoritmos é necessário uma máquina de proporções não versátil e não portátil, assim não podendo equipar projetos que têm restrições de dimensão e peso.

Um *System on Chip* (SoC) é caracterizado pela implementação de todo um sistema computacional, tais como memórias, processadores, entradas e saídas, lógicas digitais, entre outros, em um único chip de silício (CROCKETT et al., 2014). Diferente dos computadores tradicionais, que possuem seu sistema implementado a partir de módulos isolados e combinados em uma placa de circuito impresso, ou placa-mãe, os SoCs possuem como principais características um baixo custo de implementação, além de baixo consumo de potência, menor tamanho físico, maior confiabilidade e maior velocidade do sistema geral, quando comparado com um computador tradicional (CROCKETT et al., 2014). Um exemplo de um SoC é a plataforma Zynq que combina em um único chip processadores *Advanced Risc Machine* (ARM) e *Field Programmable Gate Array* (FPGA), esse último utilizado para configurar todos os módulos de um computador tradicional (CROCKETT et al., 2014).

Tendo em vista a grande vantagem dos SoCs sobre os computadores tradicionais, onde são implementados e executados os algoritmos de classificação, este trabalho apresenta um estudo da viabilidade da implementação em hardware e em software embarcado,

do algoritmo de treinamento do classificador LDA, realizando a comparação de consumo computacional, processamento computacional (tempo de execução), entre as implementações em hardware e software, onde a implementação em hardware consiste no mapeamento do algoritmo na plataforma FPGA, afim de paralelizar seus processos e a implementação em software consiste em executar o algoritmo em um sistema embarcado utilizando os cores ARM, ambos da plataforma Zynq, além da comparação com sua implementação inicial em *Matlab*.

1.2 Justificativa

As aplicações das BCIs apresentam um crescente desenvolvimento, graças ao aumento do interesse em pesquisas voltadas para o tema (BCI COMPETITION). Por ser considerado o principal processo das BCIs, a classificação requer um cuidado especial (MASON; BIRCH, 2003). Como o LDA é um classificador supervisionado, a acurácia da classificação depende inteiramente de um bom treinamento (LOTTE; GUAN, 2011). Isso requer do algoritmo de treinamento um maior esforço computacional. Como os SoCs apresentam características de baixo consumo de potência, tamanho físico pequeno e possuir todos módulos de um sistema computacional em um único chip, a implementação de algoritmos de classificação em sistema deste porte podem ou não tornar as BCIs mais acessíveis, tendo em vista que um algoritmo de treinamento embarcado em um SoC reduzirá a necessidade de um sistema computacional tradicional, além de um melhor processamento computacional, pois sua implementação em FPGA na plataforma *Zynq* torna-se possível paralelizar seus processos de execução (CROCKETT et al., 2014).

1.3 Objetivos

Esta seção apresenta os objetivos gerais e específicos propostos a serem desenvolvidos neste presente trabalho.

1.3.1 Objetivos Gerais

- Implementar parte do algoritmo de treinamento de um classificador LDA utilizando um SoC na plataforma Zynq afim de otimizar tanto o algoritmo, em relação a tempo de execução, quanto o seu consumo de recursos.

1.3.2 Objetivos Específicos

- Explorar o algoritmo de treinamento do classificador LDA desenvolvido por (LOTTE; GUAN, 2011);

- Mapear parte deste algoritmo em arquiteturas paralelas utilizando a linguagem VHDL;
- Implementar em sistema embarcado o algoritmo de treinamento utilizando os cores ARM da Zynq;
- Realizar teste e validação das implementações utilizando as bases de dados do BCI Competition III, em específico o conjunto de dados BCI III *dataset* IVa.

2 Referencial Teórico

Este capítulo apresenta os conceitos teóricos que abordam este trabalho, detalhando a estrutura cerebral responsável pela geração dos sinais de interesse na subseção 2.1, o sistema de captura dos sinais (a EEG) na subseção 2.2, o sistema que se refere às BCIs que realizam a tradução dos sinais em comandos para dispositivos na subseção 2.3, os detalhes técnicos a respeito dos sinais oferecidos pela base de dados *BCI Competition* na subseção 2.4, a estrutura geral de um classificador LDA na subseção 2.5, os conceitos de um SoC na subseção 2.6 e por fim o estado da arte das implementações de algoritmos de classificação na subseção 2.7.

2.1 O Cérebro

O Sistema Nervoso Central (SNC) é o responsável direto pelo comando do nosso comportamento geral (CLARK NASHAAT BOUTROS, 2005). Ele pode ser dividido em três principais áreas: tronco encefálico ou medula espinhal, o cérebro e o cerebelo (Figura 1) (ALVAREZ; LEMOS, 2006).

Figura 1 – Três principais áreas do cérebro (ALVAREZ; LEMOS, 2006)

O tronco encefálico, parte caudal do SNC recebe e processa todos os sinais dos sensores corporais, além de realizar o controle dos membros e do tronco humano (KANDEL, 2013). O cérebro é o processador do SNC, nele são recebidos e processados os sinais da medula espinhal, além de fornecer todos os sinais de controle para a própria medula, que por sua vez distribui os sinais para os membros e tronco (KANDEL, 2013). O cerebelo

é localizado logo atrás do tronco encefálico que se conectam através de fibras chamadas de pedúnculos (KANDEL, 2013), é a segunda maior estrutura do SNC contendo mais da metade dos neurônios cerebrais, (SIULY, 2012) é responsável pelo mapeamento em volta do indivíduo, ou seja, sua percepção (ALVAREZ; LEMOS, 2006), além do controle da região motora e a memória de movimentos (SIULY, 2012; ALVAREZ; LEMOS, 2006)

2.2 Eletroencefalografia

2.3 *Brain Computer Interface*

2.4 *BCI Competition*

A *BCI Competition* é uma competição que promove o desenvolvimento e melhoria da tecnologia voltada para as BCIs, onde são submetidas diferentes técnicas de análise de dados cerebrais (BLANKERTZ et al., 2006). Já foram realizadas quatro edições da competição, nos anos de 2001, 2002, 2004 e 2008 (BLANKERTZ et al., 2006). Em cada uma destas competições são fornecidos publicamente sinais cerebrais, adquiridos em laboratórios especializados (BLANKERTZ et al., 2006). Estes sinais são divididos em dois conjuntos de dados, os dados de treinamento e os dados de teste, que são utilizados para treinamento e teste dos algoritmos dos participantes (BLANKERTZ et al., 2006).

2.4.1 *BCI Competition III*

O objetivo do *BCI Competition III* é validar as metodologias de classificação e processamento de sinais cerebrais aplicados em BCIs desenvolvidas pelos participantes da competição (BLANKERTZ et al., 2005). Esta edição foi realizada entre Maio e Junho de 2004, onde foram disponibilizados 8 *datasets* (I, II, IIIa, IIIb, IVa, IVb, IVc e V), desenvolvidos com a participação de 49 laboratórios especializados (BLANKERTZ et al., 2006). Para cada um dos *datasets* foram realizadas diferentes tarefas que estimulam atividades cerebrais durante a aquisição dos sinais, configurando assim um objetivo específico para cada um dos *datasets* (BLANKERTZ et al., 2005).

2.4.2 *BCI Competition III - Dataset IVa*

O *dataset IVa* refere-se a um conjunto de dados adquiridos através da EEG, onde os sujeitos (indivíduos nos quais foram capturados os sinais) foram submetidos a estimular o cérebro por imagética motora, através de indicações visuais (BLANKERTZ et al., 2006). Os indivíduos foram submetidos a realizarem três tarefas, indicadas visualmente por 3.5s cada tarefa, sendo interrompidas em períodos aleatórios entre 1.75s e 2.25s, onde o sujeito era submetido a um período de relaxamento (BLANKERTZ et al., 2006). As três

tarefas de imagéticas motoras foram: (L) mão esquerda, (R) mão direita e (F) pé direito (BLANKERTZ et al., 2006).

Foram adquiridos sinais de 5 sujeitos rotulados em *aa*, *al*, *av*, *aw* e *ay*, onde foram executadas no total 280 tarefas por cada sujeito, algumas previamente rotulada (dados de treinamento) em cada instante de tempo onde a tarefa foi executada, outras não rotuladas (dados de teste) (BLANKERTZ et al., 2005). Estes sinais foram adquiridos, tratados e disponibilizados por *Fraunhofer FIRST, Intelligent Data Analysis Group (Head: Klaus-Robert Müller), and Charité University Medicine Berlin, Campus Benjamin Franklin, Department of Neurology, Neurophysics Group* (BLANKERTZ et al., 2006). A tabela 1 apresenta a quantidade de tarefas previamente classificadas (nomeados #tr) e a quantidade de tarefas não classificadas (nomeadas #te) para cada sujeito.

Tabela 1 – Número de tarefas rotuladas e não rotuladas por sujeito (BLANKERTZ et al., 2006).

Sujeitos	#tr	#te
<i>aa</i>	168	112
<i>al</i>	224	56
<i>av</i>	84	196
<i>aw</i>	56	224
<i>ay</i>	28	252

Os dados foram adquiridos e armazenados utilizando amplificadores do tipo *BraInAmp* e uma capa de eletrodos de 128 canais. Foram utilizados 118 canais de EEG posicionados de acordo com o sistema 10/20. Cada um destes canais foram filtrados em banda passante, utilizando um filtro *butterworth* de quinta ordem entre as frequências de 0.05 e 200 Hz, posteriormente foram digitalizados com uma frequência de amostragem de 1 kHz com precisão de 16 bits, apresentando uma resolução de 0.1 μ V, além disso também foram disponibilizados os mesmos dados com uma frequência de amostragem de 100 Hz (BLANKERTZ et al., 2005).

2.5 Linear Discriminant Analysis

Supondo a existência de um conjunto de dados L , com características multivariadas, e que cada dado seja conhecido devido ser proveniente de uma das classes K , tal que são predefinidas com características semelhantes aos dados. As classes podem ser exemplificadas como sendo: espécies de plantas, presença ou ausência de uma condição médica específica, diferentes tipos de tumores, tipos de veículos automotores entre outros. Para separar as classes conhecidas uma das outras, é atribuído um rótulo a cada classe, então os dados são representados como dados rotulados (IZENMAN, 2008).

Devido a indispensabilidade de diminuir as dimensões dos dados de um determinado conjunto, o objetivo do LDA é reduzir a dimensão do espaço de conjunto de dados, resolvendo o inconveniente da sobreposição (SINGH; PRAKASH; CHANDRASEKARAN, 2016).

2.6 *System-on-Chip*

System-on-Chip (SoC), implica que todo sistema que contém funcionalidades implementadas em hardware e software se encontra em um único chip de silício, combinando processamento, lógica de alta velocidade, interface, memória entre outros componentes ao invés de uma implementação maior em vários chips físicos diferentes agrupados em uma placa de circuito impresso (CROCKETT et al., 2014).

São vários os argumentos a favor da escolha de um SoC a uma placa de circuito impresso, pode-se citar que a solução é de menor custo, viabiliza transferência de dados mais rápidas e seguras entre vários elementos do sistema, possui maior velocidade geral do sistema, menor consumo de energia entre vários outros elementos que fortalecem a escolha de um SoC em sistemas discretos com componentes equivalentes (CROCKETT et al., 2014).

2.7 Estado da Arte

3 Proposta Metodológica

Este capítulo apresenta os procedimentos e métodos a serem utilizados para desenvolvimento deste presente trabalho, detalhando as metodologias *Bottom-Up*, utilizadas nas implementações tanto em hardware quanto em software, além das linguagens de programação utilizadas nas implementações e suas respectivas ferramentas. Por fim é apresentado um cronograma de atividades.

3.1 Implementação em hardware

Esta seção apresenta os dispositivos e ferramentas a serem utilizados neste presente trabalho, além das metodologias a serem desenvolvidas, para implementação do algoritmo de treinamento do classificador LDA em Hardware.

3.1.1 Dispositivos e ferramentas

Visando as grandes vantagens de utilização dos SoCs, o algoritmo de treinamento do classificador LDA será implementado nas lógicas digitais da FPGA na plataforma Zynq da placa *Zybo-Board*, utilizando os *IP-Cores* e a linguagem de programação *VHSIC Hardware Description Language* (VHDL), a fim de paralelizar os processos do algoritmo. Para o mapeamento do algoritmo em VHDL será utilizado a plataforma *Vivado HLS*, que possui todas as ferramentas necessárias para descrição, simulação, implementação e mapeamento hardware que descreve o algoritmo na FPGA.

3.1.2 Metodologias de desenvolvimento

Para desenvolvimento do algoritmo será adotada a metodologia *bottom-up*, onde cada sub-bloco desenvolvido é testado antes de ser inserido ao bloco principal, bloco de integração de todos sub-blocos do projeto, também conhecido como *Top module*. Após a implementação e simulação do *Top module*, será utilizado para teste e validação do hardware o *dataset IVa* do *BCI Competition III*, além de uma análise estatística do erro apresentado quando comparado com o desenvolvimento na plataforma *Matlab* por (LOTTE; GUAN, 2011). Para validação da eficiência do SoC serão coletados os dados das seguintes características:

- Consumo de hardware: LUTs, FFs, blocos de DSP, blocos de memória RAM, I/O, MUX;

3.2 Implementação em software

3.3 Cronograma de Atividades

Referências

- ALVAREZ, A.; LEMOS, I. d. C. Os neurobiomecanismos do aprender: a aplicação de novos conceitos no dia-a-dia escolar e terapêutico. *Revista Psicopedagogia*, Associação Brasileira de Psicopedagogia, v. 23, n. 71, p. 181–190, 2006. Citado 3 vezes nas páginas 17, 31 e 32.
- BLANKERTZ, B. et al. *BCI Competition III*. 2005. Disponível em: <http://www.bbc.de/competition/iii/desc_IVa.html>. Citado 2 vezes nas páginas 32 e 33.
- BLANKERTZ, B. et al. The bci competition iii: validating alternative approaches to actual bci problems. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 14, n. 2, p. 153–159, June 2006. ISSN 1534-4320. Citado 3 vezes nas páginas 19, 32 e 33.
- BRUNELLI, R. *Template matching techniques in computer vision: theory and practice*. [S.l.]: John Wiley & Sons, 2009. Citado na página 27.
- CLARK NASHAAT BOUTROS, M. M. D. *The Brain and Behavior: An Introduction to Behavioral Neuroanatomy*. 2. ed. Cambridge University Press, 2005. ISBN 0521840503,9780521840507. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=D00D7BF8931D1C6313FA52A331272806>>. Citado na página 31.
- CROCKETT, L. H. et al. *The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable Soc*. [S.l.]: Strathclyde Academic Media, 2014. Citado 3 vezes nas páginas 28, 29 e 34.
- IZENMAN, A. J. *Modern multivariate statistical techniques*. [S.l.]: Springer, 2008. v. 1. Citado na página 33.
- KANDEL, e. a. E. R. *Principles of Neural Science*. 5. ed. The address: Mc Graw Hill, 2013. v. 2. An optional note. ISBN 978007181001-2. Citado 3 vezes nas páginas 27, 31 e 32.
- LOTTE, F. *Study of electroencephalographic signal processing and classification techniques towards the use of brain-computer interfaces in virtual reality applications*. Tese (Doutorado) — INSA de Rennes, 2008. Citado 2 vezes nas páginas 27 e 28.
- LOTTE, F.; GUAN, C. Regularizing common spatial patterns to improve bci designs: Unified theory and new algorithms. *IEEE Transactions on Biomedical Engineering*, v. 58, n. 2, p. 355–362, Feb 2011. ISSN 0018-9294. Citado 3 vezes nas páginas 27, 29 e 35.
- MASON, S. G.; BIRCH, G. E. A general framework for brain-computer interface design. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 11, n. 1, p. 70–85, March 2003. ISSN 1534-4320. Citado 2 vezes nas páginas 27 e 29.
- SHASHOA, N. A. A. et al. Classification depend on linear discriminant analysis using desired outputs. In: IEEE. *Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2016 17th International Conference on*. [S.l.], 2016. p. 328–332. Citado na página 28.

SINGH, A.; PRAKASH, B. S.; CHANDRASEKARAN, K. A comparison of linear discriminant analysis and ridge classifier on twitter data. In: IEEE. *Computing, Communication and Automation (ICCCA), 2016 International Conference on*. [S.l.], 2016. p. 133–138. Citado na página 34.

SIULY, S. *Analysis and Classification of EEG Signals*. Dissertação (Mestrado) — University of Southern Queensland, <https://www.springer.com/gp/book/9783319476520>, 7 2012. Citado 2 vezes nas páginas 27 e 32.

SIULY, S.; LI, Y.; ZHANG, Y. *EEG Signal Analysis and Classification: Techniques and Applications*. [S.l.]: Springer, 2017. Citado 2 vezes nas páginas 27 e 28.

THEODORIDIS, S.; KOUTROUMBAS, K. et al. *Pattern recognition*. [S.l.]: Academic press London, 1999. Citado na página 28.

Apêndices

APÊNDICE A – Primeiro Apêndice

Texto do primeiro apêndice.

APÊNDICE B – Segundo Apêndice

Texto do segundo apêndice.

Anexos

ANEXO A – Primeiro Anexo

Texto do primeiro anexo.

ANEXO B – Segundo Anexo

Texto do segundo anexo.