



UMA INTERFACE HUMANO- MÁQUINA INTELIGENTE BASEADA NO RASTREAMENTO OCULAR PARA COMUNICAÇÃO ESCRITA DE PACIENTES COM SÍNDROME *LOCKED-IN*

Trabalho de Conclusão de Curso

Engenharia da Computação

Amanda Leonel Nascimento

Orientador: Fernando Buarque de Lima Neto

Co-orientador: Sérgio Campello Oliveira



UNIVERSIDADE
DE PERNAMBUCO

AMANDA LEONEL NASCIMENTO

**UMA INTERFACE HUMANO-
MÁQUINA INTELIGENTE BASEADA
NO RASTREAMENTO OCULAR
PARA COMUNICAÇÃO ESCRITA DE
PACIENTES COM SÍNDROME
*LOCKED-IN***

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia da Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, junho 2009.

A Jean-Do e Hawking, meus heróis.

Agradecimentos

Agradeço a Deus e à minha família, pelo constante apoio, incentivo e afeto.

Agradeço ao meu orientador, professor Fernando Buarque, por me guiar durante toda a graduação, como um verdadeiro pai; pelas constantes palavras de incentivo e motivação; pela formação acadêmica, profissional e humana, que vou levar para toda a vida.

Agradeço ao meu co-orientador, professor Sérgio Campello, pela admiração, amizade e ensinamentos.

Agradeço aos meus amigos pela constante preocupação e acompanhamento. Em especial a Hugo, meu companheiro de pesquisa e monografia.

Agradeço aos meus colegas de trabalho pela compreensão, incentivo e torcida.

Agradeço ao professor Eduardo Fontana e ao doutorando Tiago Vieira, do Grupo de Fotônica da Universidade Federal de Pernambuco, pela colaboração neste trabalho, possibilitando as gravações de dados realizadas através do dispositivo desenvolvido pelos mesmos.

Resumo

Pacientes portadores da Síndrome *Locked-in* apresentam um quadro clínico de paralisia rara e grave. Devido a um dano cerebral, eles perdem os movimentos dos músculos voluntários do corpo e a capacidade de falar. Surpreendentemente, suas funções cognitivas mantêm-se em perfeito estado, i.e. quando acordados, eles podem ver, ouvir e compreender tudo à sua volta. Em geral, o único movimento voluntário que ainda se mantém é o do controle dos olhos. Este trabalho apresenta uma prova de conceito baseada em uma Interface Humano-Máquina inteligente para comunicação escrita de pacientes com Síndrome *Locked-in*. Propomos a utilização de técnicas de Visão Computacional e Inteligência Artificial, a fim de mapear o rastreamento dos movimentos dos olhos em escrita na tela do computador. A modelagem do sistema é composta por módulos cujos resultados obtidos foram bastante satisfatórios. As imagens processadas apresentaram padrões representativos em baixa resolução espacial. Redes Neurais Artificiais foram treinadas para reconhecimento dos padrões analisados. As altas taxas de acerto e curta duração dos experimentos comprovaram a viabilidade do modelo de interface proposto.

Abstract

Patients with 'Locked-in Syndrome' (LIS) report a rare and severe clinical case as they become paralyzed and voiceless. Due to a brain damage, patients lose their movements of the voluntary muscles of the body and their ability to speak. Surprisingly, their cognitive functions remain in perfect order. In general, the only voluntary movement that still remains is the control of the eyes. This work presents a proof-of-concept and algorithm based on an Intelligent Human-Computer for writing Interface to LIS patients. The system uses techniques from Computer Vision and Artificial Intelligence in order to acquire and detect eye movements affording written communications of LIS patients. The design of the system is composed of modules which results were very satisfactory. The processed images show representative patterns at low spatial resolution. Artificial Neural Networks were trained in order to recognize the patterns analyzed. The high rates of success and short duration of the experiments demonstrate the feasibility of the proposed interface model.

Sumário

ÍNDICE DE FIGURAS.....	VI
ÍNDICE DE TABELAS.....	VIII
TABELA DE SÍMBOLOS E SIGLAS	IX
CAPÍTULO 1 INTRODUÇÃO	11
1.1 Motivação e Problema	11
1.2 Objetivos e Metas	12
1.3 Sistema proposto: Visão geral	13
1.4 Estrutura da monografia	14
CAPÍTULO 2 FUNDAMENTAÇÃO TEÓRICA.....	16
2.1 Síndrome <i>Locked-in</i>	16
2.1.1 Causas e tipos de LIS.....	16
2.1.2 Casos, reabilitação e comunicação.....	18
2.2 O olho humano.....	20
2.2.1 Anatomia e diversidade	20
2.2.2 Movimentos oculares	22
2.3 Aspectos de Visão Computacional e Imagem Digital	23
2.3.1 Representação de imagens digitais	24
2.3.2 Resoluções de uma imagem.....	25
2.3.3 Operações com imagens e filtros.....	26
2.4 Redes Neurais Artificiais.....	26
2.4.1 O neurônio artificial	27
2.4.2 Arquiteturas de RNAs	28

2.4.3	Projeto de uma RNA.....	29
2.5	Estado da arte em Interfaces Humano-Máquina	29
2.5.1	Trabalhos relacionados.....	30
 CAPÍTULO 3 UMA HCI INTELIGENTE PARA ESCRITA DE		
PACIENTES COM SÍNDROME <i>LOCKED-IN</i>		32
3.1	Estrutura da HCI proposta	32
3.2	Módulo DAQ: Aquisição de Dados.....	33
3.2.1	Base de imagens necessárias para a prova de conceito.....	33
3.2.2	Avaliação de bases de imagens.....	33
3.2.3	Aquisição de dados.....	38
3.3	Módulo PDI: Processamento Digital de Imagem.....	39
3.3.1	Extração & Classificação dos quadros.....	40
3.3.2	Pré-processamento das imagens.....	41
3.3.3	Segmentação da pupila e íris.....	41
3.3.4	Redução da densidade de resolução & Conversão.....	42
3.4	Módulo IA: Inteligência Artificial	42
3.4.1	<i>Multilayer Perceptron</i>	42
3.4.2	<i>WEKA®</i>	43
3.5	Módulo GUI: Interface Gráfica do Usuário.....	44
3.5.1	Dimensões.....	44
 CAPÍTULO 4 RESULTADOS.....		46
4.1	Resultados do Módulo DAQ.....	46
4.1.1	Dados para base de imagens “Alfabeto”	46
4.2	Resultados do Módulo PDI.....	47
4.2.1	Extração & Classificação dos quadros.....	47
4.2.2	Pré-processamento das imagens.....	50
4.2.3	Segmentação da pupila e íris.....	51
4.2.4	Redução da densidade de resolução.....	52

4.2.5	Abordagem básica	52
4.2.6	Conversão das imagens	54
4.2.7	Geração da base binária 'alfabeto.arff'	55
4.3	Resultados do Módulo IA	55
4.3.1	Treinamentos e Testes	55
4.4	Resultados do Módulo GUI	57
4.4.1	Protocolo de comunicação	58
CAPÍTULO 5 CONCLUSÃO E TRABALHOS FUTUROS.....		59
5.1	Discussões e Dificuldades.....	59
5.2	Melhorias e Trabalhos futuros.....	60
5.2.1	Adição de módulo de calibração ao sistema	60
5.2.2	Configuração automática dos parâmetros.....	61
5.2.1	Realização de testes reais	61
5.2.2	Implementação de um produto final.....	61
BIBLIOGRAFIA.....		62
APÊNDICE A: FORMATO DA BASE BINÁRIA 'ALFABETO.ARF' ..		69

Índice de Figuras

Figura 1.	Estrutura geral e organização dos módulos da HCI proposta.	13
Figura 2.	Sistema Nervoso Central, com destaque para as estruturas inferiores [16].	17
Figura 3.	Placa de comunicação usada no filme “O Escafandro e a Borboleta” [21].	19
Figura 4.	Diagrama esquemático do olho humano [27].	20
Figura 5.	Diversidade de cores, formas e contornos do olho humano [29].	21
Figura 6.	Sequência de movimentos dos olhos durante uma leitura [32].	23
Figura 7.	Relação entre áreas da visão computacional e afins [33,34].	24
Figura 8.	Filtragem espacial de imagens através de máscaras [41].	26
Figura 9.	Representação de um neurônio artificial [12].	27
Figura 10.	Exemplo de arquitetura de uma rede MLP [12].	28
Figura 11.	Ilustrações dos trabalhos relacionados. Da esquerda para a direita, de cima para baixo: [48,49,50,54,60].	31
Figura 12.	Arquitetura do modelo proposto.	32
Figura 13.	Amostras de imagens da “BioID - <i>Face Database</i> ” [62].	35
Figura 14.	Amostras de imagens da “IrisBase - <i>Smart Sensors Iris Image Database</i> ” [63].	36
Figura 15.	Amostras de imagens da “DOVES - <i>Database of Visual Eye Movements</i> ” [64].	37
Figura 16.	Amostras de imagens da “CASIA - <i>Iris Image Database</i> ” [66].	37
Figura 17.	Dispositivo de rastreamento de movimentos oculares baseado em webcam e iluminação com LED infravermelho [49].	38
Figura 18.	Sequência interna de processamentos do módulo PDI.	40
Figura 19.	Sequência de modelagem do módulo IA (RNA).	43

Figura 20.	<i>Grid</i> 4x7 de classes do módulo GUI.	45
Figura 21.	Amostras de imagens das classes 1x1 e 4x7 da base “Alfabeto”.	48
Figura 22.	Amostras de imagens de uma mesma classe: 2x6.	48
Figura 23.	Amostras de imagens das classes “piscado” e “sacádicos longos”.	49
Figura 24.	Imagem original rotacionada em 25°.	50
Figura 25.	Imagem com resolução espacial reduzida.	50
Figura 26.	Imagem final em tons de cinza e seu histograma à direita.	51
Figura 27.	Amostras de segmentação da pupila e íris e binarizações, realizadas pelos métodos <i>Sobel</i> e <i>Canny</i> , da esquerda para a direita.	51
Figura 28.	Imagem dilatada.	52
Figura 29.	Imagens com resolução reduzida. Da esquerda para a direita: 40x17 e 60x25 <i>pixels</i> , respectivamente.	52
Figura 30.	Novos resultados das operações de binarização, dilatação, redução espacial e inversão, da esquerda para a direita, respectivamente.	53
Figura 31.	Comparação dos resultados finais para classes diferentes (acima) e para uma mesma base (abaixo).	53
Figura 32.	Comparação entre os diferentes padrões de imagens gerados pelo módulo PDI: imagem original acima; segmentada à esq.; básica à dir.	54
Figura 33.	Disposição final dos caracteres da interface gráfica.	58

Índice de Tabelas

Tabela 1.	Causas de LIS em 139 casos relatados até 1986, adaptada de [7].	18
Tabela 2.	Informações técnicas sobre os vídeos gerados.	47
Tabela 3.	Classes da base de imagens “Alfabeto”	47
Tabela 4.	Classes extras da base de imagens “Alfabeto”.	48
Tabela 5.	Informações técnicas sobre a base de imagens “Alfabeto”.	49
Tabela 6.	Informações técnicas sobre a base binária “Alfabeto”.....	55
Tabela 7.	Melhor configuração de parâmetros obtidos para a MLP.	56
Tabela 8.	Melhores resultados da rede MLP configurada.	57

Tabela de Símbolos e Siglas

Em ordem de aparição no texto:

- **HCI:** *Human-Computer Interface* (Interface Humano-Máquina – IHM)
- **LIS:** *Locked-in Syndrome* (Síndrome do Encarceramento)
- **TCC:** Trabalho de Conclusão de Curso
- **DAQ:** *Data Acquisition* (Aquisição de Dados)
- **PDI:** Processamento Digital de Imagem
- **IA:** Inteligência Artificial
- **GUI:** *Graphical User Interface* (Interface Gráfica de Usuário)
- **OMS:** Organização Mundial da Saúde
- **AIDS:** *Acquired Immunodeficiency Syndrome* (Síndrome da Imunodeficiência Adquirida - SIDA)
- **SN:** Sistema Nervoso
- **SNP:** Sistema Nervoso Periférico
- **SNC:** Sistema Nervoso Central
- **ELA:** Esclerose Lateral Amiotrófica
- **RGB:** *Red Green Blue*
- **Pixel:** *Picture Element*
- **VGA:** *Video Graphics Array*
- **ppi:** *pixels per inch*
- **dpi:** *dots per inch*
- **RNAs:** Redes Neurais Artificiais
- **MLP:** *Multilayer Perceptron*
- **EMG:** Eletromiografia
- **EEG:** Eletroencefalografia

- **ISL:** *Intelligent Systems Lab*
- **DOVES:** *Database of Visual Eye Movements*
- **CBSR:** *Center for Biometrics and Security Research*
- **UFPE:** *Universidade Federal de Pernambuco*
- **USB:** *Universal Serial Bus*
- **LED:** *Light Emitting Diode*
- **RAM:** *Random Access Memory*
- **LoG:** *Laplacian of Gaussian (Laplaciano da Gaussiana)*
- **WEKA:** *Waikato Environment for Knowledge Analysis*
- **ARFF:** *Attribute-Relation File Format*
- **MAE:** *Mean Absolute Error (Erro Absoluto Médio)*
- **RMSE:** *Root Mean Square Error (Raíz do Erro Quadrático Médio)*
- **RAE:** *Relative Absolute Error (Erro Relativo Absoluto)*
- **RRSE:** *Root Relative Squared Error (Raíz do Erro Quadrático Relativo)*
- **ASCII:** *American Standard Code for Information Interchange*

Capítulo 1

Introdução

1.1 Motivação e Problema

Uma interface para usuário é um grupo de recursos com o qual o usuário interage com um sistema [1]. O termo "Interface Humano-Máquina" (IHM), do inglês "*Human-Computer Interface/Interaction*" (HCI), é mais utilizado no contexto de Computação e Engenharia.

HCI deve prover recursos de (i) entrada, que permitam aos usuários manipular o sistema e (ii) saída, que permitam ao sistema indicar os efeitos da manipulação pelo usuário. Usabilidade, acessibilidade e ergonomia são algumas características centrais em projetos de aplicações de HCI. Algumas técnicas de Computação podem ser empregadas a fim de melhorar o desempenho de uma HCI. Técnicas de Reconhecimento de Padrões [2], Inteligência Artificial [3] e Engenharia de Software [4] são frequentemente aplicadas no estado da arte de HCIs.

Novas tendências e potenciais grupos de usuários de sistemas de computacionais surgem constantemente [5]. Em particular, HCIs podem ser utilizadas para beneficiar pacientes com capacidade de comunicação restrita ou comprometida. É o caso dos pacientes com Síndrome *Locked-in* [6] (do português, "Síndrome do Encarceramento").

A Síndrome *Locked-in* (LIS) é uma condição rara em que um paciente humano apresenta os músculos voluntários paralisados (devido a um acidente vascular cerebral grave, por exemplo). Porém, mantém todas as funções cognitivas em perfeito estado, i.e., quando acordado, pode ver, ouvir e compreender tudo à sua volta. Dezenas de casos já foram descobertos e relatados na literatura médica ao redor do mundo [7].

O paciente mais célebre portador desse quadro tornou a síndrome conhecida ao escrever um livro [8] sobre suas memórias e seu estado psicológico, durante o "encarceramento". Em 1995, Jean-Dominique Bauby, redator-chefe da revista de

moda francesa “Elle”, sofreu um derrame cerebral e teve o corpo inteiro paralisado, com uma exceção: seu olho esquerdo. O caso de Jean-Do tornou-se ainda mais comovente pela maneira como ele escreveu o livro. Diariamente, uma funcionária da editora ditava-lhe o alfabeto repetidas vezes. E, apenas com piscar da pálpebra esquerda, ele sinalizava a letra escolhida para compor cada palavra, frase e capítulo do seu livro. O drama vivido não apenas por Jean-Do, mas também por sua incansável soletradora, poderia ter sido “amenizado” com a utilização de uma HCI específica, de requisitos especiais e restritos.

1.2 Objetivos e Metas

Este Trabalho de Conclusão de Curso (TCC) tem como principal objetivo modelar e implementar uma HCI inteligente para escrita, baseada no rastreamento de movimentos oculares. A meta principal é construir uma prova de conceito a partir da modelagem e implementação de um sistema que possibilite a comunicação de pacientes com Síndrome *Locked-in* (ou casos de paralisia semelhantes).

Os objetivos específicos deste trabalho são:

- Revisar literatura sobre HCIs e estudar os fatores humanos considerados no quadro da síndrome;
- Apresentar um módulo de aquisição de dados (DAQ) para obtenção dos sinais de entrada;
- Modelar e implementar um módulo de Processamento Digital de Imagem (PDI) [9] para processamento das imagens obtidas;
- Modelar e aplicar um módulo dotado de Inteligência Artificial (IA), a fim de (i) refinar possíveis ambigüidades no reconhecimento dos sinais de entrada da interface e (ii) permitir a melhoria de desempenho do sistema;
- Considerar a viabilidade de construção de um protótipo experimental, incluindo uma interface com o usuário (GUI).

Como se trata de uma prova de conceito, não faz parte do escopo deste projeto:

- Implementações adicionais em hardware, para captação de sinais em tempo real;
- Considerações neurofisiológicas e psicológicas do quadro do paciente;
- Análise de desempenho e testes reais do sistema final.

1.3 Sistema proposto: Visão geral

Uma visão geral do sistema é apresentada na Figura 1 e descrita brevemente a seguir.

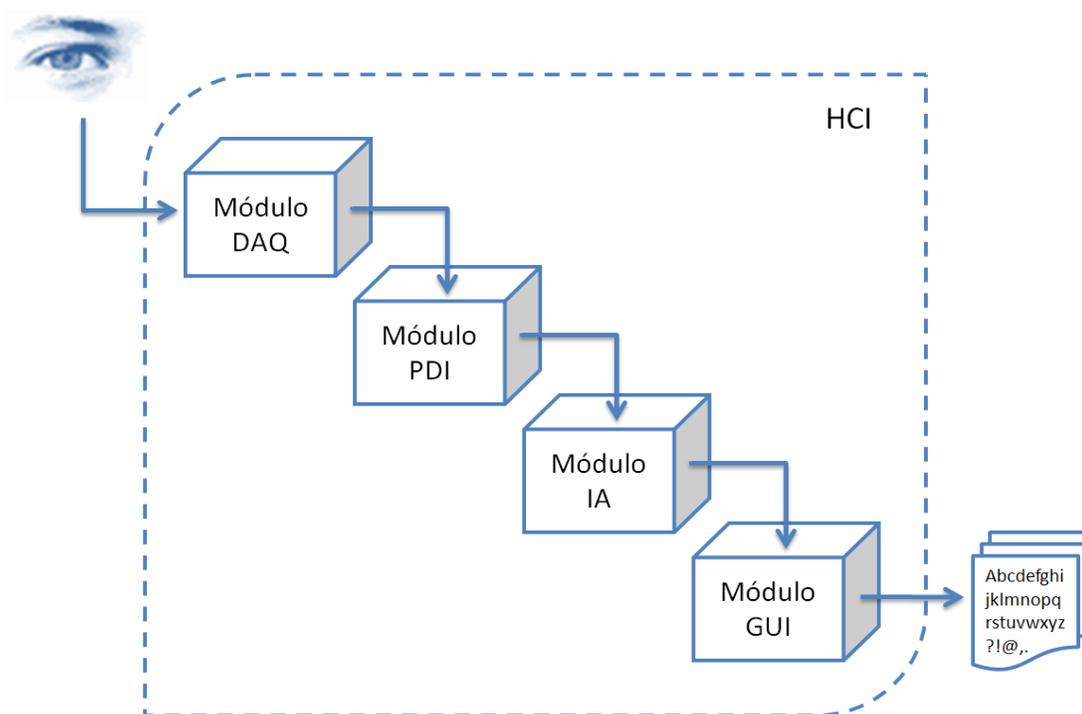


Figura 1. Estrutura geral e organização dos módulos da HCI proposta.

- **Módulo DAQ: Aquisição de Dados**

Define o padrão e o formato de sinais do paciente utilizados para controle do sistema, i.e. imagens dos movimentos dos olhos e a obtenção das mesmas.

- **Módulo PDI: Processamento Digital de Imagem**

Aplica técnicas de Processamento de Imagem e Visão Computacional [10] mais empregadas, tais como Binarização de imagens e Segmentação de Íris.

- **Módulo IA: Inteligência Artificial**

Define a arquitetura e os parâmetros utilizados na aplicação de uma técnica de Inteligência Artificial – a saber, Redes Neurais Artificiais [11,12] – para reconhecimento dos padrões direcionais analisados.

- **Módulo GUI: Interface Gráfica do Usuário**

Apresenta uma Interface Gráfica do Usuário para mapeamento visual dos padrões identificados em caracteres do alfabeto português, permitindo comunicação escrita na tela do computador.

1.4 Estrutura da monografia

A presente monografia foi dividida em 5 capítulos, resumidos a seguir:

- **Capítulo 1: Introdução**

Contém o texto introdutório sobre o trabalho, abordando motivação, problema, objetivos e metas. Também apresenta uma visão inicial do sistema.

- **Capítulo 2: Fundamentação Teórica**

Reúne os principais conceitos necessários para a fundamentação teórica e compreensão do trabalho proposto. Para tal, apresenta a Síndrome *Locked-in* e aborda aspectos sobre Visão Humana e Computacional, Redes Neurais Artificiais e Interfaces Humano-Máquina.

- **Capítulo 3: Uma HCI Inteligente para Escrita de Pacientes com Síndrome *Locked-in***

Descreve o modelo de interface proposto e seus módulos de: Aquisição de Dados, Processamento Digital de Imagem Inteligência Artificial e Interface Gráfica do Usuário.

- **Capítulo 4: Resultados**

Apresenta os resultados experimentais obtidos a partir da prova de conceito proposta. E descreve ainda os melhores resultados encontrados para cada módulo do sistema.

- **Capítulo 5: Conclusões**

Resume o trabalho final, reúne dificuldades encontradas, discussões geradas e as conclusões obtidas. Melhorias e trabalhos futuros também são propostos.

Capítulo 2

Fundamentação Teórica

2.1 Síndrome *Locked-in*

Segundo a “Organização Mundial da Saúde” (OMS), síndrome é “um agrupamento de sinais e sintomas com base em sua frequente co-ocorrência, que pode sugerir uma patogênese básica, curso, padrão familiar ou tratamento comuns” [13]. Ou seja, uma síndrome não caracteriza necessariamente uma única doença, mas um grupo de doenças semelhantes em determinado aspecto. O conceito indica ainda algo provisório, de conhecimento incerto. São exemplos populares de síndromes: “Síndrome da Imunodeficiência Adquirida” (SIDA ou AIDS), Síndrome de Down, Síndrome de Estocolmo, Síndrome do Pânico, Síndrome de Peter Pan e Síndrome Pré-Menstrual [13].

A Síndrome *Locked-in* (LIS), também chamada de “Síndrome do Encarceramento”, porém mais conhecida pelo termo em inglês, foi definida em 1966. Mas, desde 1986, passou a ser descrita como qualquer “quadriplegia e anartria com preservação da consciência” [6]. “Quadriplegia” porque o paciente apresenta paralisia de aproximadamente todos os músculos voluntários do corpo. “Anartria” devido à impossibilidade de articular palavras. “Consciência preservada” uma vez que o paciente mantém todas as suas funções cognitivas, i.e., percepção (e.g. visão e audição), atenção, habilidades intelectuais, memória visual e verbal etc., estas que raramente são afetadas [6]. Popularmente, essa condição rende a um paciente portador da síndrome o quadro de “enterrado vivo” [14].

2.1.1 Causas e tipos de LIS

O Sistema Nervoso (SN) humano pode ser dividido em Periférico (SNP) e Central (SNC). Este último divide-se ainda em encéfalo e medula espinhal. A ponte ventral é uma estrutura do tronco encefálico localizada entre essas duas principais porções do SNC. Ela é responsável pelo envio de informação sensorial ao cérebro [15]. A Figura 2 ilustra sua localização no SNC.

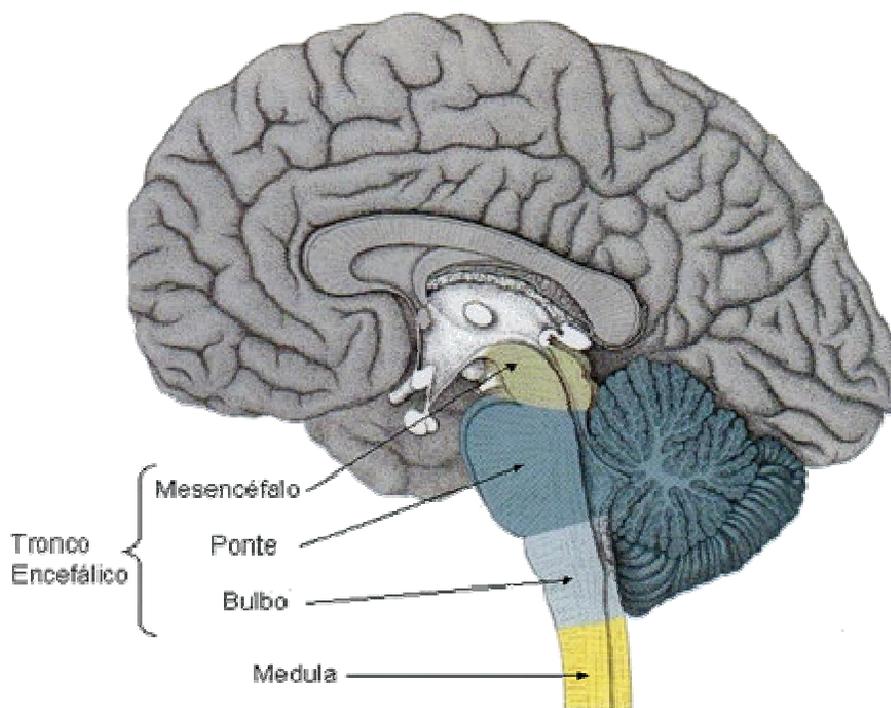


Figura 2. Sistema Nervoso Central, com destaque para as estruturas inferiores [16].

A principal causa de quadros de LIS são danos na ponte, por exemplo, decorrentes de um AVC ou aneurisma. Porém, outros danos cerebrais podem resultar no mesmo quadro [6,7]. Como é considerada síndrome, a LIS pode ser desenvolvida a partir de diferentes etiologias, i.e., possui diversos quadros e doenças de origem. A Tabela 1 ilustra as principais causas de 139 casos de LIS reunidos em 1986.

A LIS tem sido classificada em três categorias [6]:

1. **Clássica:** é caracterizada por quadriplegia e anartria com consciência e visão preservadas;
2. **Completa:** engloba a categoria anterior, contudo, o paciente exibe também vestígios de movimentos voluntários;
3. **Total:** apresenta total imobilidade e inabilidade para comunicação, porém mantém a consciência intacta.

Em muitos casos de LIS total, o quadro pode ser confundido com estado de coma [17] ou ainda, morte encefálica [18]. Entretanto, exames como imagens de ressonância magnética podem indicar o dano na ponte e atividade cerebral normal

referente às funções cognitivas [15]. Aconselha-se ainda que o quadro de pacientes em coma seja constantemente reavaliado, pois há casos de pacientes que despertaram apresentando LIS [19]. Contudo, um diagnóstico correto de que já haviam despertado do coma e que apresentavam plena atividade cognitiva demorou até mesmo anos [7].

Tabela 1. Causas de LIS em 139 casos relatados até 1986, adaptada de [7].

Causa	Nº de pacientes	Sexo M/F	Nº de mortes	Idade média no início da LIS
I. Vascular	105	61/44	70	56
A. Enfarte da ponte base	82	48/34	53	56
B. Hemorragia da ponte	14	7/7	11	55
C. Fenômeno embólico	3	2/1	2	36
D. Enfarte do mesencéfalo	4	2/2	4	70
E. Ataque isquêmico transiente	2	2/0	0	74
II. Não-Vascular	34	28/8	14	40
A. Trauma	9	8/1	3	32
B. Mielinólise pontina central	7	5/2	3	43
C. Tumor	3	2/1	3	47
D. Encefalite	3	2/1	0	20
E. Manipulação cervical	2	1/1	1	43
F. Síndrome de Behçet	2	-	2	-
G. Esclerose múltipla	1	0/1	1	42
H. Abscesso de ponte	1	1/0	1	40
I. Inflamação arterial	1	0/1	0	45
J. Polineuropatia pós-infecciosa	1	1/0	0	45
K. Embolismo de ar	1	1/0	0	54
L. Toxidade por diazepam	1	1/0	0	77
M. Toxinas	1	1/0	0	6
N. Abuso de heroína	1	1/0	0	36

2.1.2 Casos, reabilitação e comunicação

Não há estatísticas precisas sobre o número total de casos de LIS no mundo inteiro. Estima-se que existam milhares somente na Europa [20], muitos ainda não

diagnosticados. Os casos mais conhecidos são o da americana Julia Tavalaro, do jornalista francês Jean-Dominique Bauby e do físico inglês Stephen Hawking.

Aos 32 anos, Julia Tavalaro foi internada em um hospital em Nova Iorque após dois AVCs graves em 1996. Durante 6 anos, acreditou-se que Julia encontrava-se em estado vegetativo. Até que um parente identificou uma tentativa de sorriso de Julia, após contar uma piada dentro de seu quarto. Identificado o caso de LIS, iniciou-se seu tratamento e tentativa de reabilitação. Julia se comunicava piscando os olhos, para escolher uma letra entre o alfabeto que lhe era apontado em uma placa. Conseguiu recuperar parte do movimento da cabeça, com o qual controlava uma cadeira de rodas computadorizada. E, com a co-autoria do poeta Richard Tayson, escreveu sua autobiografia [19]. Morreu de pneumonia aos 68 anos de idade em 2003.

Outro paciente célebre tornou a síndrome conhecida ao escrever um livro [8] sobre suas memórias e seu estado psicológico durante o “encarceramento”. Em 1995, Jean-Dominique Bauby, redator-chefe da revista de moda francesa “Elle”, sofreu um derrame cerebral e teve o corpo inteiro paralisado, com exceção do seu olho esquerdo. O caso de Jean-Do tornou-se ainda mais comovente pela maneira como ele escreveu o livro. Um pequeno “protocolo de comunicação” fora estabelecido para que Jean-Do se comunicasse com enfermeiros e familiares. Diariamente, uma funcionária da editora ditava-lhe o alfabeto repetidas vezes, ilustrado na Figura 3. E, apenas com piscar da pálpebra esquerda, ele sinalizava a letra escolhida para compor cada palavra, frase e capítulo do seu livro. Jean-Do também morreu de pneumonia, em 1997, dias após publicar seu livro. Em 2007, foi lançado um filme baseado em sua vida e obra [21].



Figura 3. Placa de comunicação usada no filme “O Escafandro e a Borboleta” [21].

De todos os casos, Stephen Hawking é talvez aquele de maior sobre-vida. Por volta de 1965, o renomado físico teórico inglês foi diagnosticado com ELA - Esclerose Lateral Amiotrófica [22], uma doença neurodegenerativa progressiva e fatal, que destrói os neurônios motores que controlam os movimentos voluntários dos músculos, mas sem atacar funções cognitivas. Seu quadro avançou rapidamente nos dois anos seguintes, levando-o à paralisia de quase todos os seus movimentos e à perda da fala natural, isto é, a um quadro de LIS. Porém, estabilizou-se surpreendentemente desde então. Atualmente, Hawking usa uma cadeira de rodas e um sintetizador de voz para locomover-se e comunicar-se. Aos 67 anos, e cerca de 45 anos após o diagnóstico de ELA, continua a dar palestras, publicar artigos e receber prêmios por suas descobertas científicas. Também escreveu um livro [23] e autorizou produção de um filme baseados em suas obras e biografia [24].

2.2 O olho humano

Em pacientes portadores de LIS clássica, os únicos movimentos voluntários restantes são, em geral, apenas os dos olhos [6,14]. Assim, julga-se necessária uma revisão sobre o olho humano, em especial, sua anatomia, fisiologia e dinâmica.

2.2.1 Anatomia e diversidade

O globo ocular humano é composto pelas seguintes regiões, ilustradas na Figura 4 [25,26]:

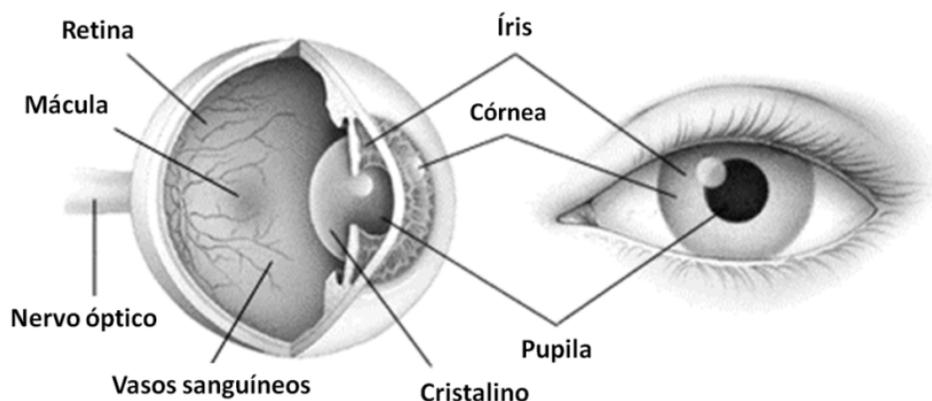


Figura 4. Diagrama esquemático do olho humano [27].

- **Pupila e íris:** controlam a intensidade da passagem da luz visível do meio exterior (cujo espectro de comprimento de onda encontra-se entre 380 nm e 780 nm aproximadamente [28]) até os órgãos sensoriais da retina. Para tal, a pupila possui diâmetro ajustável. A íris é a porção colorida e mais visível do olho e pode conferir, a cada ser humano, um padrão único de identidade.
- **Mácula:** absorve o excesso de luz que atravessa o olho humano, agindo como uma proteção natural.
- **Córnea e cristalino:** têm a função de focar a luz através da pupila até a retina, como lentes fixas.
- **Retina:** é a região do olho humano onde os estímulos visuais são convertidos em imagens invertidas, retidas para leitura do cérebro.
- **Nervo óptico:** transmite as informações visuais da retina até o cérebro.
- **Vasos sanguíneos:** são responsáveis pela irrigação sanguínea do globo ocular.

A Figura 5, extraída de [29], ilustra a diversidade do olho humano.



Figura 5. Diversidade de cores, formas e contornos do olho humano [29].

Embora o olho humano, em geral, apresente todos os elementos básicos citados aqui, a diversidade de formas e contornos – e não apenas de cores – varia significativamente, de acordo com características genéticas, ambientais e comportamentais dos indivíduos [30].

2.2.2 Movimentos oculares

Além dos movimentos de contração e dilatação da pupila, para regulação da entrada de radiação visível, o globo ocular apresenta também movimentos em torno de seu eixo. Essa dinâmica é importante pois: (i) leva as imagens selecionadas até a retina; (ii) as mantém lá até que o objeto seja reconhecido; (iii) estabiliza as imagens contra movimentos da cabeça e do corpo; e (iv) alinha as imagens das retinas nos dois olhos, permitindo uma visão única e tridimensional [31].

Os quatro principais tipos de movimentos do olho humano são [26,31,32]:

1. **Sacádicos** (*saccades*): para reter rapidamente, no centro da retina, novos estímulos diferentes. São movimentos rápidos (como saltos) usados para deslocar o olhar para qualquer objeto selecionado. Na leitura, ocorrem pequenos sacádicos tipicamente três vezes por segundo, geralmente para olhar de uma palavra para outra.
2. **Busca suave**: para rastrear estímulos visuais de movimento. Mantém a linha da visão no objeto selecionado durante os intervalos entre os sacádicos;
3. **Fixação**: para fixar estímulos visuais estacionários.
4. **Piscado**: não é um movimento do olho, mas das pálpebras, a fim de distribuir a lágrima produzida constantemente pelas glândulas lacrimais, umedecendo, nutrindo e limpando a córnea. Dura, em média, 0,3 a 0,4 segundos. Um adulto costuma piscar a cada 2 ou até 10 segundos; porém, durante a leitura, pode piscar somente 3 a 4 vezes por minuto.

A Figura 6 ilustra a sequência de movimentos vertical e horizontal dos olhos durante uma leitura [32]. As mudanças abruptas no movimento dos olhos são movimentos sacádicos.

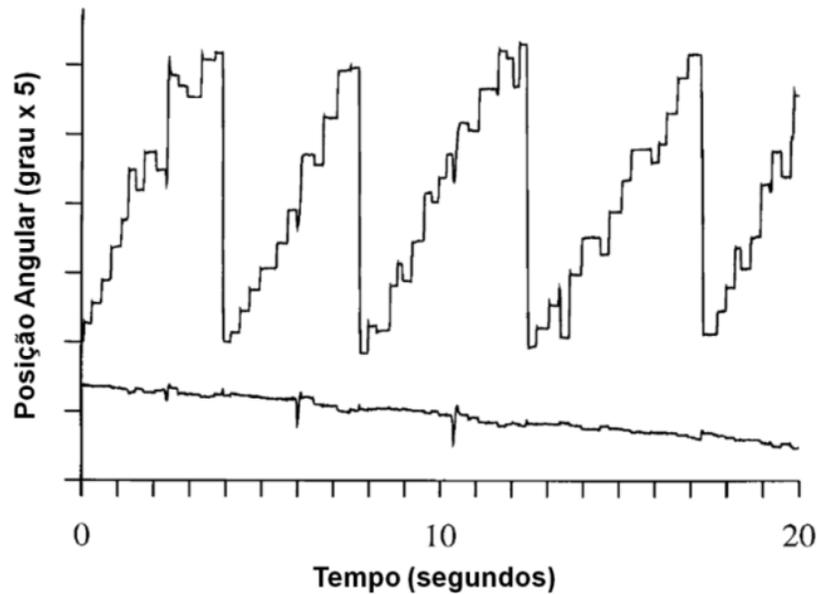


Figura 6. Sequência de movimentos dos olhos durante uma leitura [32].

2.3 Aspectos de Visão Computacional e Imagem Digital

As relações entre Visão Computacional [10] e áreas afins são tão próximas que chegam a ser confundidas. O diagrama da Figura 7, adaptada de [33,34] resume algumas dessas correlações. Uma imagem digital (ou apenas “imagem”, deste ponto em diante) é a materialização e base de grande parte dos processos que envolvem essas áreas [33]. E a finalidade da área de Visão Computacional é obter, a partir de uma imagem digital de entrada, informações geométricas, topológicas ou físicas sobre o cenário que originou essa imagem.

Ainda na área de Visão Computacional, a Segmentação de Imagens [35] estuda como fragmentar uma imagem digital em regiões ou objetos (em geral homogêneos) de forma a modificá-la para facilitar sua análise. As técnicas de Segmentação de Imagens [36,37] podem ser baseadas em características pontuais da imagem (cor, intensidade) ou no formato dos objetos (i.e., contornos, pontos, linhas, descontinuidades etc.). Suas aplicações são importantes para identificação, medição ou quantificação de objetos em uma cena, por exemplo.

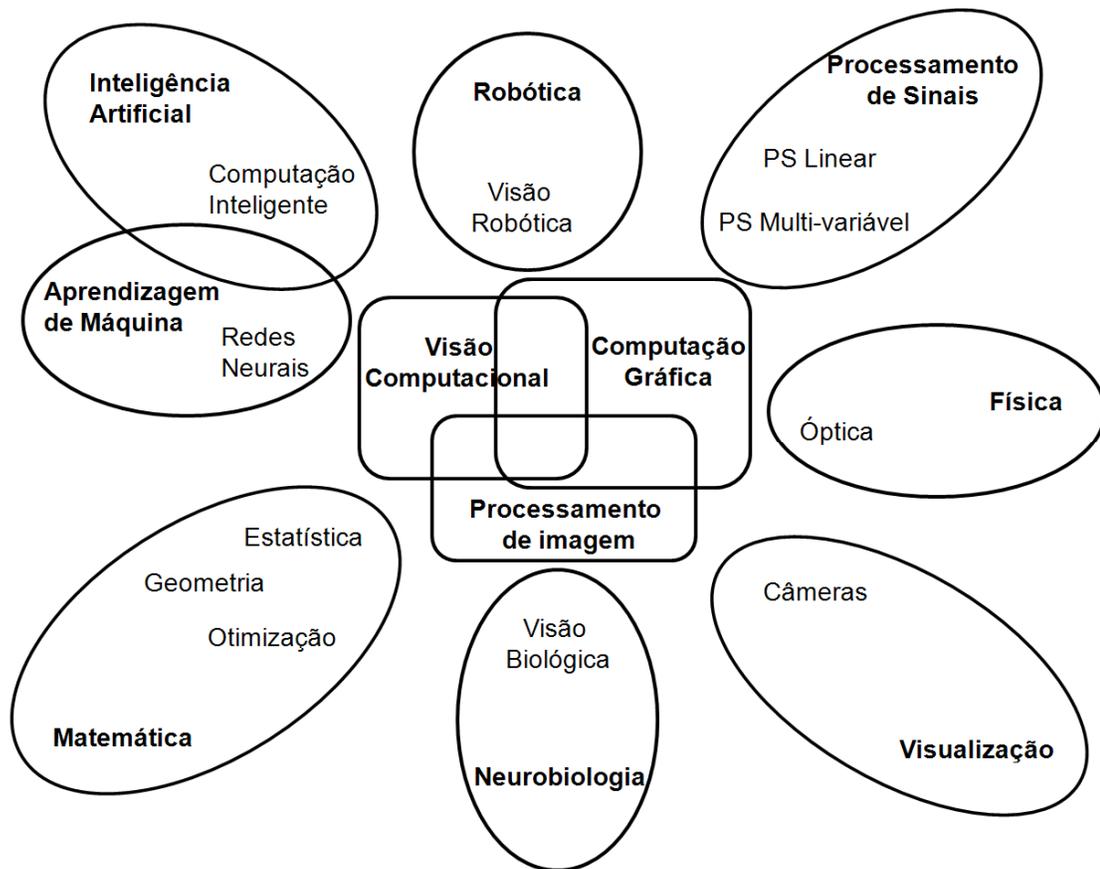


Figura 7. Relação entre áreas da visão computacional e afins [33,34].

2.3.1 Representação de imagens digitais

Uma imagem real é composta por um conjunto de pontos, cada um visualizado por meio de um impulso luminoso que define a sua cor. Assim, uma imagem pode ser descrita matematicamente, conforme a Equação 1 [33], como uma função definida em uma superfície bidimensional, cujos valores estão dentro de um espaço de cores. Isto é:

$$f: U \subset \mathbb{R}^2 \rightarrow C \quad (1)$$

onde:

- U é um subconjunto plano;

- $C = R^n$ é um espaço de cores, onde cada banda espectral se caracteriza pela presença de frequências em uma determinada faixa do espectro;
- f é a função imagem, cujo conjunto de valores é denominado de “conjunto de valores ou cores da imagem”.

Se $C = R^1$, diz-se que a imagem é “monocromática”. Ou ainda, se $n > 1$, a imagem é dita “colorida”. Assim, $C = R^3$ pode ser representado, por exemplo, pelo sistema de cores RGB (Vermelho, Verde e Azul) [9].

A representação matricial pode ser utilizada para realizar a discretização espacial da imagem, considerando o domínio U como um retângulo, conforme a Equação 2 [33]:

$$U = [a, b] \times [c, d] = \{(x, y) \in R^2; a \leq x \leq b \text{ e } c \leq y \leq d\}. \quad (2)$$

Se $(a = 0)$, $(b = M - 1)$ e $(d = N - 1)$, temos a matriz $I_{m \times n}$ da Equação 3. Cada ponto $f(x, y)$ é chamado de elemento da imagem ou *pixel* e possui uma “cor” (valor) associada a ele. A origem da imagem pode ser representada pelo *pixel* central da matriz [38,39,40].

$$f(x, y) = \begin{bmatrix} f(0,0) & \cdots & f(0, N - 1) \\ \vdots & \ddots & \vdots \\ f(M - 1, 0) & \cdots & f(M - 1, N - 1) \end{bmatrix} \quad (3)$$

2.3.2 Resoluções de uma imagem

Ao número de linhas M da matriz de *pixels* $I_{m \times n}$ dá-se a denominação de “resolução vertical”; ao número de colunas N , “resolução horizontal”. O produto $M \times N$, i.e., a multiplicação dos números de *pixels* da base pela altura, é chamado de “resolução espacial” da representação matricial e define a dimensão de uma imagem [33]. O padrão para vídeo VGA (*Video Graphics Array*), por exemplo, apresenta dimensão de 640×480 *pixels*, i.e., 307.200 *pixels* por quadro de imagem. Já a “densidade de resolução” de uma imagem indica o número de *pixels* por unidade linear de medida, em ppi (pixels por polegada) ou dpi (pontos por polegada).

Por último, é chamado de “resolução de cor” o número de *bits* utilizado para armazenar o valor de cor de cada *pixel* da imagem. Para representar uma imagem monocromática (ou “binarizada”), é necessário apenas 1 *bit* por *pixel*, que indica preto ($bit=0$) ou branco ($bit=1$), por exemplo. Para imagens coloridas no sistema de

cor RGB, em geral, é suficiente utilizar 24 *bits*, i.e., 8 *bits* para cada componente de cor do espaço RGB [33]. Uma correspondência entre os números de *bits*, de cores e imagens é dada a seguir:

- Imagens monocromáticas: 1 bit = $2^1 = 2$ cores
- Imagens em tons de cinza: 8 bits = $2^8 = 256$ cores
- Imagens coloridas em RGB: 24 bits = $2^{24} = 16.777.216$ cores (256x256x256)

2.3.3 Operações com imagens e filtros

Operações com imagens desempenham um importante papel na Visão Computacional, pois podem proporcionar mudanças das características de uma imagem, para melhoria ou busca de informações. As operações no espaço de imagem utilizando filtros são chamadas de filtragem [9,33]. É através da aplicação de filtros que, por exemplo, a Segmentação de Imagens funciona.

De forma matricial, uma operação de filtragem espacial sobre uma imagem pode ser descrita como uma convolução discreta entre as imagens, i.e., entre as matrizes do filtro e da imagem. Ou ainda, diz-se também que a imagem é varrida por uma máscara, *pixel a pixel*, conforme ilustrado na Figura 8, extraída de [41].

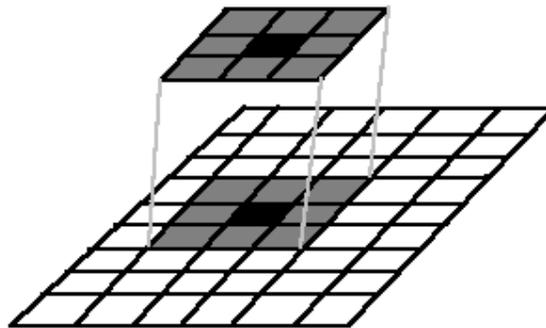


Figura 8. Filtragem espacial de imagens através de máscaras [41].

2.4 Redes Neurais Artificiais

Ilustradas na Figura 7, RNAs [11,12] são uma técnica de Inteligência Artificial [3] inspirada e fundamentada em diversas disciplinas: Neurociências, Matemática, Física etc. Suas aplicações são igualmente amplas, inclusive em Processamento de

Imagens [42], Visão Computacional [43] e Segmentação de Imagens [44], devido à capacidade de aprendizagem e generalização de tais redes.

Em RNA, um conjunto de unidades simples de processamento – neurônios – são conectadas em rede, que passa a apresentar habilidades de processamento não linear, geralmente com potencialidade de resolver problemas computacionais complexos.

2.4.1 O neurônio artificial

O neurônio artificial mais comum é baseado no modelo matemático de McCulloch e Pitts [45], que está ilustrado na Figura 9, adaptada de [12].

O neurônio recebe os sinais de entrada x_i , que são ponderados pelos pesos, w_{ij} , numa operação tipicamente linear, e avaliados por uma função de ativação φ , geralmente não-linear. Se o valor final for maior do que o limiar θ_k , a saída y_j , do neurônio é disparada, ou seja, propaga um sinal não nulo.

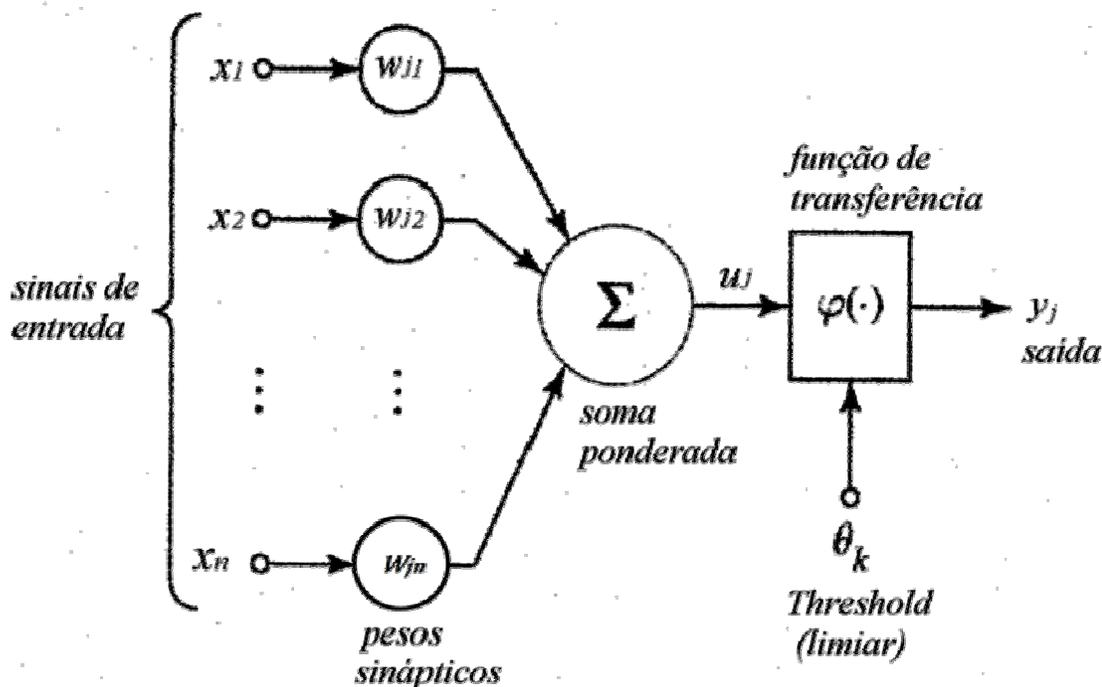


Figura 9. Representação de um neurônio artificial [12].

2.4.2 Arquiteturas de RNAs

Existem incontáveis modelos, arquiteturas e algoritmos de aprendizagem para RNAs. Os mais conhecidos são, de longe, as redes *Perceptron* e MLP – *Multilayer Perceptron* – treinadas com o algoritmo de treinamento *Backpropagation* [11,12].

As redes MLP são compostas de camadas de neurônios conectados entre si, apresentam mais de uma camada intermediária e possuem grande poder computacional se comparadas às redes sem camadas escondidas [11,12]. A Figura 10 – adaptada de [12] - ilustra uma típica MLP, com duas camadas ocultas, *feedforward* e *fully connected*, i.e., todas as saídas dos neurônios de uma camada conectadas a todos os neurônios apenas da camada seguinte.

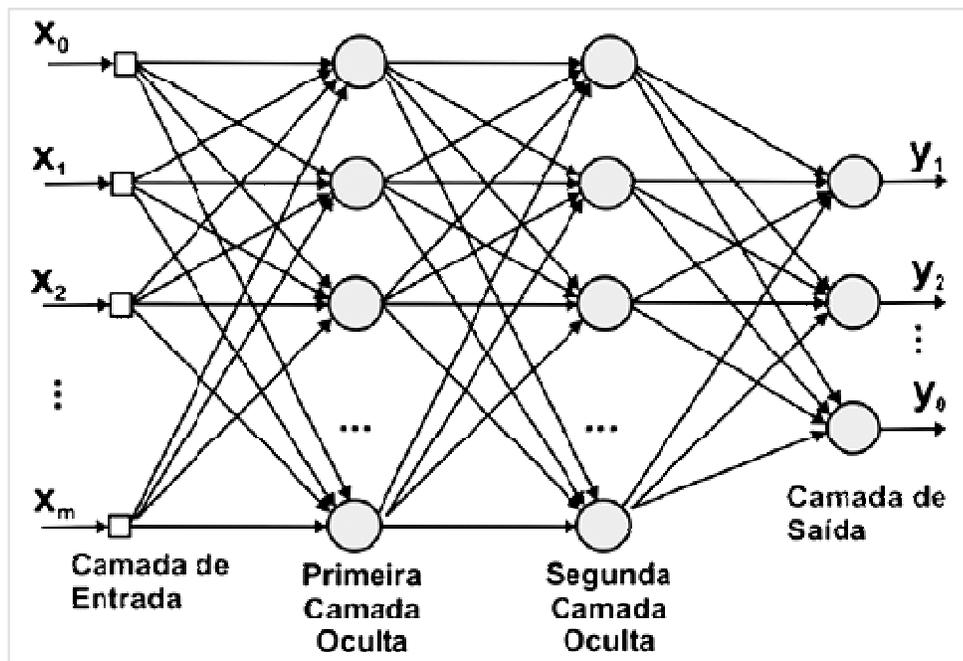


Figura 10. Exemplo de arquitetura de uma rede MLP [12].

O algoritmo *Backpropagation* e suas variações são os mais utilizados para treinamento de redes MLP. Ele é um algoritmo do tipo “supervisionado”, i.e., as entradas e saídas desejadas são apresentadas por um “supervisor” externo durante o treinamento, afim de ajustar os pesos da rede e encontrar uma relação entre os pares entrada-saída, para generalização de novos dados.

2.4.3 Projeto de uma RNA

As decisões de projeto de uma RNA ficam a cargo do projetista da rede e dependem bastante dos problemas a serem resolvidos. Contudo, algumas fases devem ser seguidas cuidadosamente [46]:

1. **Coleta e seleção de dados:** para treinamento, validação e teste da rede;
2. **Configuração da rede:** para escolha de modelo, topologia, tipo de função de transferência, parâmetros (e.g., taxa de aprendizado) etc.
3. **Treinamento:** para ajuste dos pesos da rede;
4. **Validação:** para avaliar a rede ainda durante o treinamento;
5. **Teste:** para avaliar a generalização da rede final.

2.5 Estado da arte em Interfaces Humano-Máquina

Uma interface para usuário é um grupo de recursos com o qual o usuário interage com um sistema. O termo "Interface Humano-Máquina" (IHM), do inglês "*Human-Computer Interface/Interaction*" (HCI), é mais utilizado no contexto de Computação e Engenharia. Novas tendências e potenciais grupos de usuários surgem constantemente [5]. Em especial, HCIs podem ser utilizadas para beneficiar pacientes com capacidade de comunicação restrita ou comprometida, como pacientes com LIS.

Uma HCI deve prover recursos de (i) entrada: que permitam aos usuários manipular o sistema, e também de (ii) saída: que permitam ao sistema indicar os efeitos da manipulação pelo usuário. Usabilidade, acessibilidade e ergonomia são algumas características que devem ser levadas em consideração.

São diversos os tipos de HCIs [1,47], de acordo com a forma de contato, ambiente utilizado ou sistema. O usuário alvo é, contudo, o parâmetro mais importante para definir uma HCI. É possível utilizar sensores adicionais, como câmeras ou eletrodos, para obter mais informações sobre o estado do usuário e reagir de acordo com suas ações sem comandos específicos.

Algumas técnicas de Computação podem ser empregadas a fim de melhorar o desempenho de uma HCI. Técnicas de Reconhecimento de Padrões [2], Inteligência Artificial [3] e Engenharia de Software [4] são frequentemente aplicadas no estado da arte das HCIs.

2.5.1 Trabalhos relacionados

O estudo e desenvolvimento de HCIs que não exijam o contato físico do usuário para interação têm crescido bastante na última década como uma alternativa promissora de comunicação no futuro. Existem incontáveis produtos comerciais de HCIs com características técnicas que dispensam o contato físico do usuário para interação. Contudo, apenas poucos deles podem ser usados por pacientes com LIS. Em geral, tais equipamentos são caros, e os requisitos de acessibilidade, usabilidade e segurança demandados pelos pacientes são bastante específicos, devido às suas limitações de interação.

Os tipos de HCIs mais utilizados em pacientes com LIS podem ser classificados de acordo com a técnica de entrada de dados utilizada [48]:

1. **HCIs baseadas em Eletromiografia (EMG):** que monitoram os impulsos e demais sinais das membranas de músculos esqueléticos motores;
2. **HCIs baseadas em Eletroencefalograma (EEG):** que monitoram as correntes elétricas do encéfalo, i.e., a atividade cerebral do indivíduo;
3. **HCIs baseadas no rastreamento de movimentos oculares (*eye-tracking*):** que monitoram a posição da pupila do indivíduo.

Das técnicas listadas, o rastreamento dos movimentos dos olhos tem sido a mais promissora para casos de LIS não total, uma vez que é possível ser desenvolvida a baixo custo [48,49]. Nessa linha, muitos trabalhos têm sido desenvolvidos com o uso de algoritmos convencionais e técnicas clássicas de Visão Computacional [48,49,50], de técnicas alternativas de Inteligência Artificial, tais como RNAs [51,52,53,54,55,56,57] e alguns apresentam ainda um sistema para escrita baseada no rastreamento do movimento dos olhos [54,58,59,60]. A Figura 11 ilustra os principais trabalhos relacionados a esse projeto.

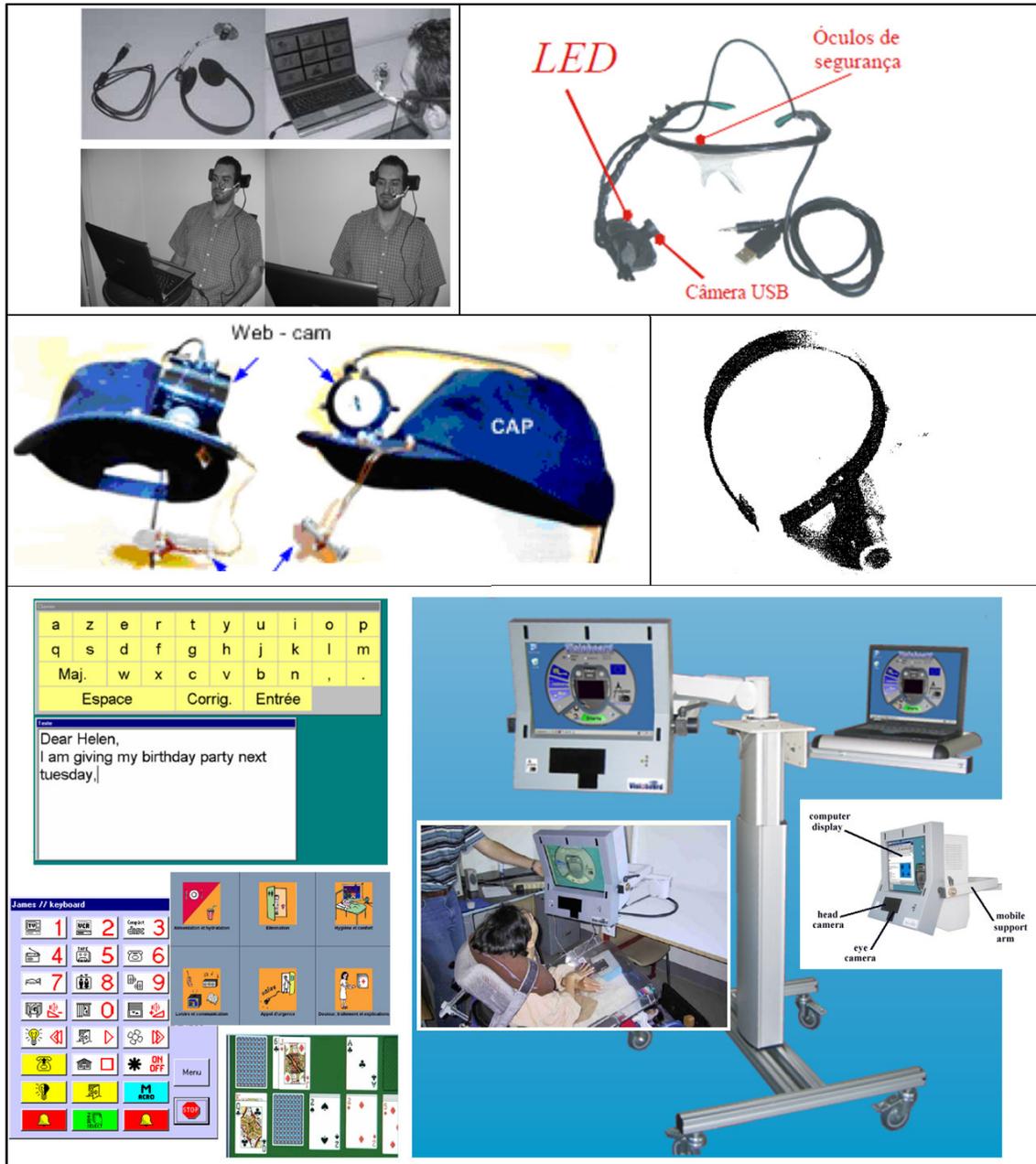


Figura 11. Ilustrações dos trabalhos relacionados. Da esquerda para a direita, de cima para baixo: [48,49,50,54,60].

Capítulo 3

Uma HCI Inteligente para Escrita de Pacientes com Síndrome *Locked-in*

Este capítulo descreve a prova de conceito proposta neste trabalho: uma Interface Humano-Máquina para pacientes com LIS. O modelo proposto é baseado no rastreamento do movimento dos olhos, através de técnicas de Processamento Digital de Imagem e Inteligência Artificial, para possibilitar a comunicação escrita de pacientes portadores de LIS, via computadores pessoais.

3.1 Estrutura da HCI proposta

A estrutura da HCI proposta está ilustrada na Figura 12 e detalhada, em sequência, nas seções adiante.

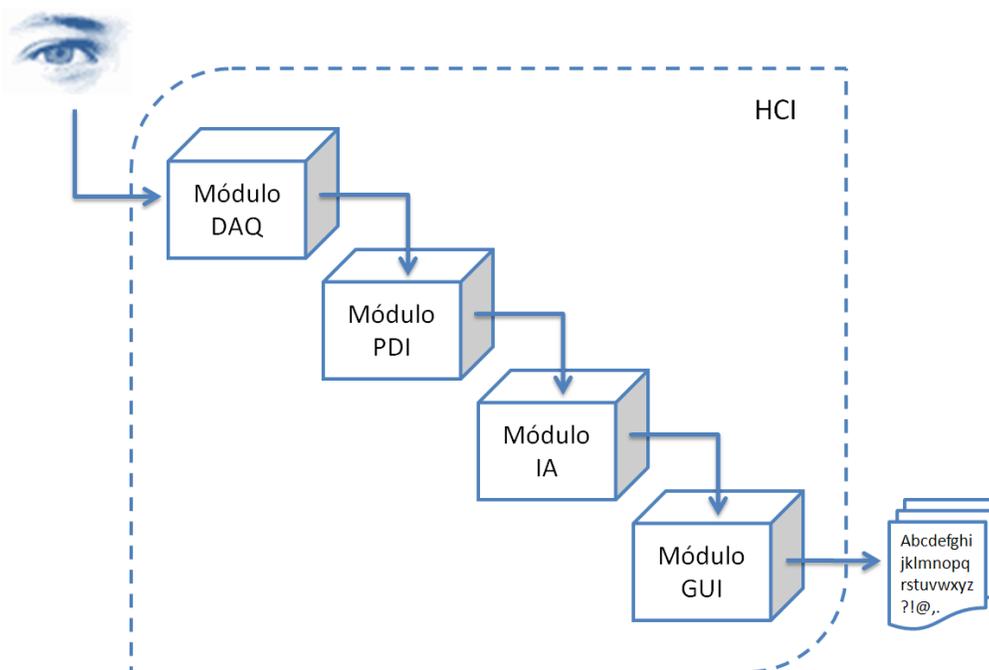


Figura 12. Arquitetura do modelo proposto.

3.2 Módulo DAQ: Aquisição de Dados

Um módulo DAQ tem a finalidade de adquirir dados de entrada para processamento em um sistema. Em geral, possui sensores apropriados que monitoram os sinais a serem adquiridos.

Conforme listado na seção 1.2 do Capítulo 1, não faz parte do escopo desta proposta o desenvolvimento de um dispositivo para aquisição dos dados, i.e., para captação de vídeo em tempo real. Diversos trabalhos propõem uma implementação física para tal finalidade [48,49,50,54].

3.2.1 Base de imagens necessárias para a prova de conceito

Apesar da não implementação de um dispositivo real para o módulo DAQ, foi necessária a utilização de uma base de dados para a prova de conceito ora proposta. Nos módulos a seguir, essa base deve ser processada pelo módulo PDI e utilizada para treinamento, validação e teste do módulo IA, cujo processamento se baseia em Redes Neurais Artificiais.

Conforme descrito mais adiante, na seção 3.4, a RNA deve ser capaz de associar os pares imagem-ponto apresentados. Para tal, a base de imagens selecionada deve ilustrar:

- i. olhos humanos com razoável resolução: o suficiente para identificar íris e pupila;
- ii. movimentos oculares: imagens com diferentes localizações da íris e pupila dentro do espaço reservado ao globo ocular entre-pálpebras;
- iii. as coordenadas dos respectivos pontos observados durante a aquisição de cada imagem.

3.2.2 Avaliação de bases de imagens

Inicialmente, cinco bases de imagens de olhos, disponíveis na Internet, foram analisadas para utilização nesta prova de conceito. Uma breve avaliação dessas bases é incluída ao final da descrição de cada uma, a seguir.

3.2.2.1 ISL - Eye Image Database [61]

- **Instituto de origem:** *Intelligent Systems Lab, Rensselaer Polytechnic Institute* – Nova Iorque, EUA.
- **Número de arquivos:** Aprox. 140000 (7,85 GB)
- **Status:** Não adquirida (parcialmente disponível sob requisição).
- **Descrição:** Contém sequências as imagens capturadas a partir de câmeras em infravermelho. Cada sequência é decomposta em uma sequência de imagens da pupila escura e em outra sequência de imagens com o brilho correspondente da pupila.
- **Avaliação:** Negativa - além de ser parcialmente proprietária, a base não apresenta os pontos observados durante a aquisição de cada imagem.

3.2.2.2 BioID – Face and Eye Position Databases [62]

- **Instituto de origem:** *BioID* - Sachseln, Suíça.
- **Número de arquivos:** 2 x 1521 (124 MB e 5,94 MB)
- **Status:** Adquiridas parcialmente (disponível para *download* livremente).
- **Descrição:** Disponibiliza em dois subconjuntos de bases: (i) 1521 imagens de faces em tons de cinza e (ii) as respectivas coordenadas das posições dos olhos em cada imagem. A Figura 13 exhibe amostras de imagens do primeiro subconjunto da base *BioID*.
- **Avaliação:** Negativa – as coordenadas das posições dos olhos não indicam pontos observados.

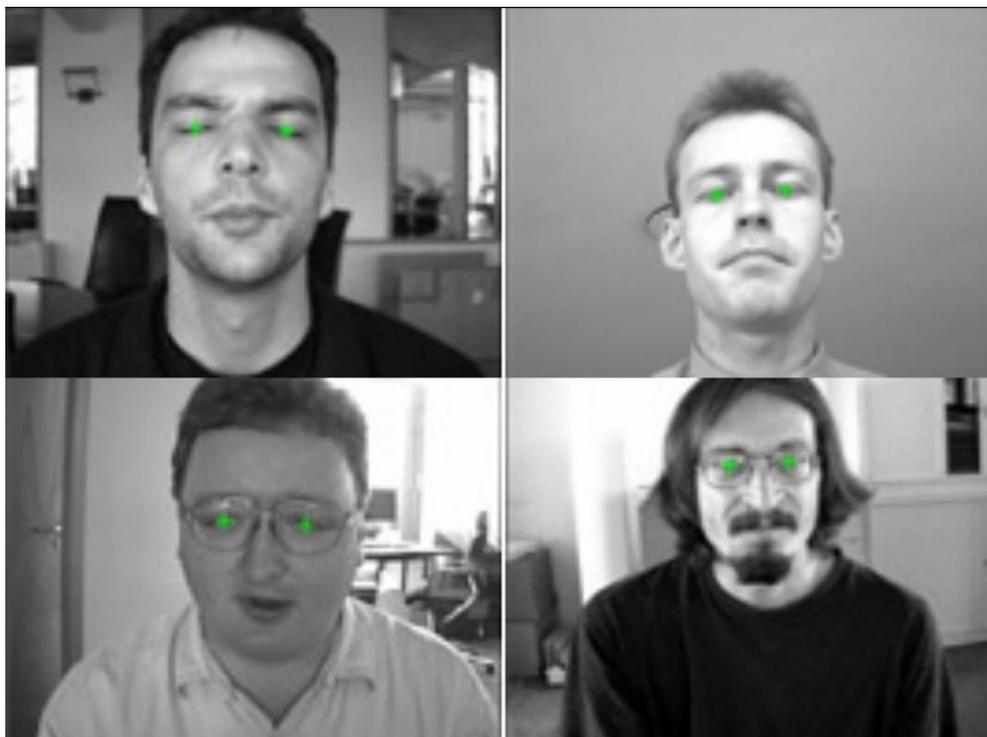


Figura 13. Amostras de imagens da “BiID - Face Database” [62].

3.2.2.3 IrisBase - Smart Sensors Iris Image Database [63]

- **Instituto de origem:** *Smart Sensors Ltda*, Reino Unido.
- **Número de arquivos:** 1000 (74,2 MB)
- **Status:** Adquirida (disponível para *download* sob requisição).
- **Descrição:** Apresenta imagens de íris de 800 indivíduos (1600 olhos), sendo 20 imagens de cada olho esquerdo e direito, em alta resolução (1280 x 960 pixels). Seu módulo DAQ utiliza uma câmera com iluminação de LEDs infravermelhos. A base é dividida ainda em um grupo de imagens ideais e outro não-ideal, ilustradas na Figura 14, respectivamente.
- **Avaliação:** Negativa - apesar da excelente qualidade das imagens, todas apresentam a pupila centralizada, i.e., aproximadamente na mesma posição, ilustrando pouco ou nenhum movimento dos olhos.

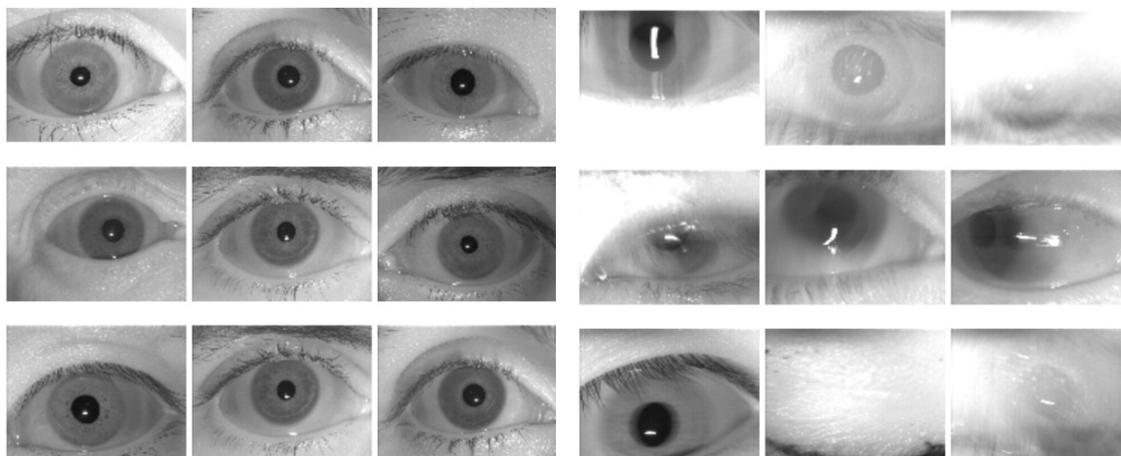


Figura 14. Amostras de imagens da "IrisBase - *Smart Sensors Iris Image Database*" [63].

3.2.2.4 DOVES - A Database of Visual Eye Movements [64]

- **Instituto de origem:** *Laboratory for Image & Video Engineering (LIVE), University of Texas at Austin, EUA.*
- **Número de arquivos:** 264 (284 MB)
- **Status:** Adquirida (disponível para *download* sob requisição).
- **Descrição:** É uma coleção de 101 imagens naturais (em tons de cinza e 1024 x 768 pixels), contendo as coordenadas dos movimentos dos olhos de 29 observadores. Ao todo, há cerca de 30000 pontos de alta precisão, fixados durante tais observações. Também inclui diversas funções de manipulação de imagens para *Matlab®* [65]. A Figura 15 ilustra uma planta e, em seguida, os pontos percorridos pelo olho humano durante análise da mesma.
- **Avaliação:** Negativa – embora seja única base avaliada que ilustra explicitamente as coordenadas dos pontos observados, não descreve as posições dos olhos no momento das fixações dos pontos.

3.2.2.5 CASIA - Iris Image Database [66]

- **Instituto de origem:** CBSR - *Center for Biometrics and Security Research* - Beijing, China.
- **Número de imagens:** 26277 (2,08 GB)
- **Status:** Adquirida (disponível para *download* sob requisição).
- **Descrição:** Inclui três subconjuntos de bases, cujas imagens variam de acordo com o sensor e a iluminação utilizados, resolução espacial etc. Ao todo, reúne mais de 700 indivíduos chineses e 1500 olhos. A Figura 16 ilustra olhos orientais com iluminação diferenciada no olho da imagem à direita.
- **Avaliação:** Negativa – a base também não informa as coordenadas dos pontos observados durante a aquisição das imagens.



Figura 15. Amostras de imagens da “DOVES - *Database of Visual Eye Movements*” [64].

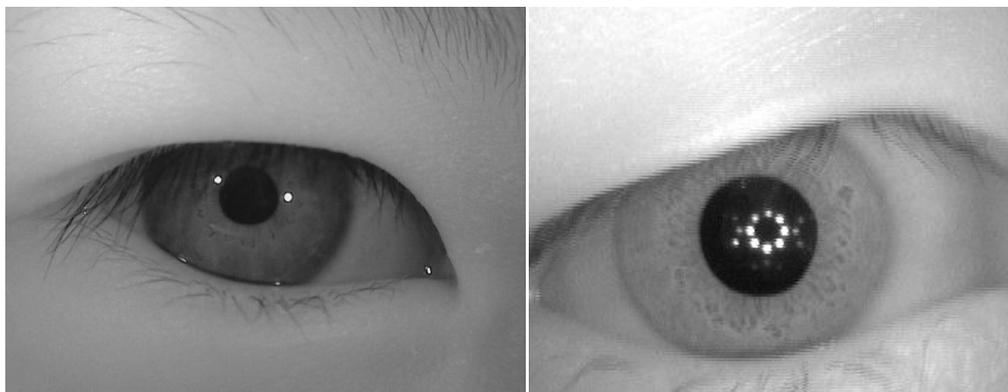


Figura 16. Amostras de imagens da “CASIA - *Iris Image Database*” [66].

3.2.3 Aquisição de dados

Dadas as avaliações negativas de todas as bases analisadas, a seguinte decisão de projeto foi tomada: adquirir os próprios dados necessários para gerar a base de imagens da prova de conceito do sistema. O dispositivo selecionado para aquisição de dados para geração de uma base particular de imagens é descrito a seguir.

3.2.3.1 Dispositivo de rastreamento de movimentos oculares baseado em *webcam* e iluminação com LED infravermelho

Para o módulo de DAQ deste trabalho, foi utilizado o dispositivo proposto durante o mestrado do engenheiro eletrônico Tiago Vieira, no Programa de Pós Graduação em Engenharia Elétrica da UFPE [67] e desenvolvido em 2008 [49]. A Figura 17 ilustra o equipamento.

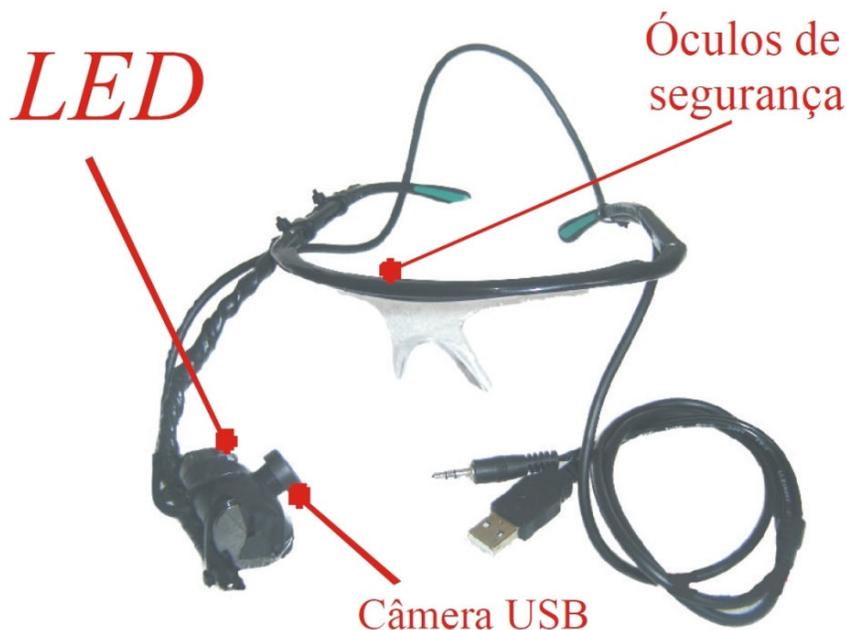


Figura 17. Dispositivo de rastreamento de movimentos oculares baseado em *webcam* e iluminação com LED infravermelho [49].

Os componentes utilizados foram [49]:

- Câmera USB - Vimicro ZC0-301PLH;

- Haste de suporte;
- Óculos de segurança;
- LED infravermelho;
- Filtro óptico;
- Laptop configurado com *Windows® XP*, processador *Intel® Core 2 Duo* 1,6 GHz e 2GB de memória RAM.

A haste e os óculos de segurança são utilizados para prover suporte ajustável e estável aos equipamentos da Figura 16. Eles mantêm a câmera em uso a uma distância de 6 a 10 cm do olho direito do usuário. Em especial, as lentes dos óculos podem ser removidas, a fim de permitir melhor visibilidade e a utilização do dispositivo por pacientes usuários de óculos de grau.

Vale destacar ainda que a câmera utilizada possui 6 LEDs infravermelhos – cujo comprimento de onda permite melhor visualização da íris. *Webcams* simples também podem ser utilizadas, desde que removido o filtro de bloqueio infravermelho que as câmeras comuns possuem. A radiação emitida pelos 6 LEDs, entretanto, não ultrapassa o limite estabelecido para exposição segura do olho humano a esse comprimento de onda [49].

3.3 Módulo PDI: Processamento Digital de Imagem

O módulo PDI tem por objetivo o processamento das imagens obtidas, a fim de utilizá-las como dados na RNA.

A ferramenta utilizada para processamento de imagens foi o *Matlab®* versão 7-R2006a, que provê apoio no desenvolvimento de rotinas para processamento de sinais e cálculos com matrizes. O programa possibilita também a utilização de diversas extensões (chamadas de *toolboxes*), que dispõem de muitos recursos para processamento de imagens [68]. “*Image Processing Toolbox*”, versão 5.2 (R2006a), foi utilizada neste módulo.

A Figura 18 ilustra as fases da sequência interna de processamentos do módulo PDI, listada a seguir:

1. Extração & Classificação dos quadros;
2. Pré-processamento das imagens;
3. Segmentação da pupila e íris;
4. Redução da densidade de resolução & Conversão;

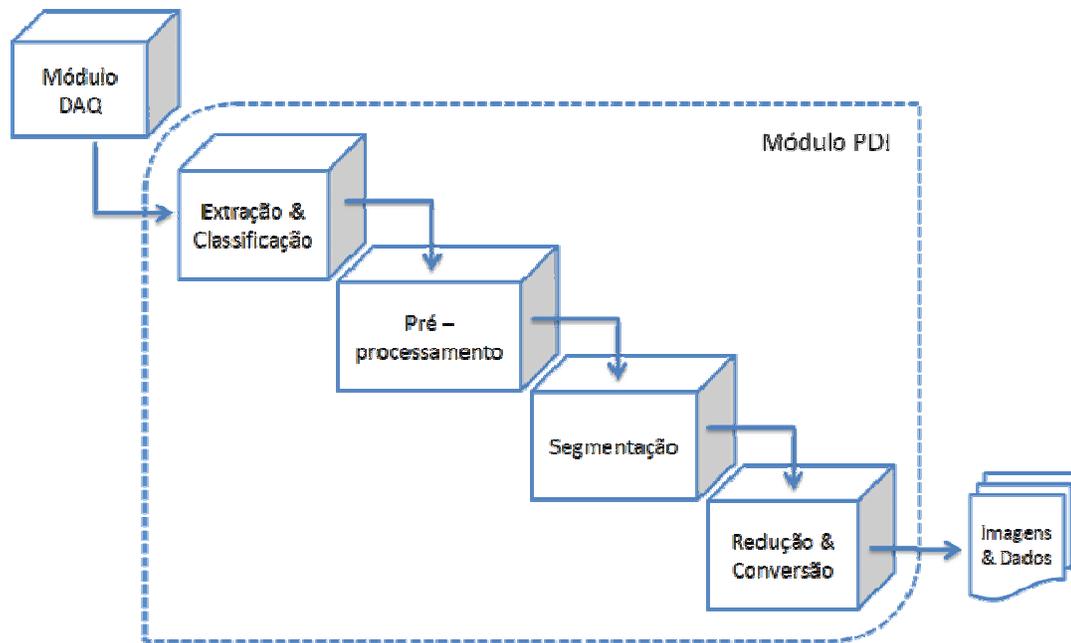


Figura 18. Sequência interna de processamentos do módulo PDI.

3.3.1 Extração & Classificação dos quadros

Para extração de todos os quadros dos vídeos gerados, também foi utilizado um *software* de edição de vídeos: Corel® VideoStudio Pro X2 [69].

Foi implementada uma função ‘framesFromAvi’ que utiliza a função de leitura de vídeos ‘aviread’ do *Matlab*®, entre outras. Essa última, contudo, não permite a leitura de vídeos maiores do que aproximadamente 1024 MB. Dessa forma, os vídeos foram separados em blocos iguais de 900 quadros (aproximadamente 700 MB), utilizando o Corel®, que permite a edição não-destrutiva de vídeos.

3.3.2 Pré-processamento das imagens

As imagens geradas necessitam passar por um pré-processamento, de modo a auxiliar o processamento de fato, mais adiante.

1. **Rotação:** a função `'imrotate'` do *Matlab*® foi utilizada em uma rotina para rotacionar a imagem, de forma a compensar a inclinação da câmera e manter os olhos em posição horizontal em relação à imagem. Como método de interpolação utilizado na rotação, foi o clássico `'nearest'` – interpolação por vizinho mais próximo. Outro método de interpolação disponível nessa função, o `'bicubic'`, poderia gerar valores de *pixels* fora do espaço de cores utilizado [70].
2. **Redução da resolução espacial:** foi implementada uma rotina com a função `'imcrop'` do *Matlab*® para redução da resolução espacial das imagens, mantendo apenas a região do globo ocular em cada quadro.
3. **Conversão para tons de cinza:** foi utilizada a função `'rgb2gray'` do *Matlab*® para converter as imagens em sistema RGB para tons de cinza. A conversão em tons de cinza possibilita a utilização de alguns filtros a seguir.

3.3.3 Segmentação da pupila e íris

Inúmeros trabalhos propõem, analisam ou aplicam técnicas de segmentação e identificação da pupila e íris [71,72,73,74,75,76]. Neste trabalho, a segmentação da íris e pupila foi parcialmente baseada em trabalho de Carr et al. [75].

A segmentação da pupila e íris pode ser resumida a um problema de detecção de bordas. Para tal, utilizou-se filtros passa-alta [33], i.e., que permitem a passagem apenas das altas frequências de uma imagem (como aquelas presentes em contrastes de bordas). A ordem das operações realizadas é descrita a seguir:

1. Detecção de bordas & Binarização:

Na detecção da pupila e íris, foram analisados os seguintes métodos da função `'edge'` de detecção de borda do *Matlab*®: *Sobel*, *Prewitt*, *Roberts*, *Laplaciano da Gaussiana (LoG)*, *Zero-cross*, *Canny* [68]. Essa função deve ser utilizada apenas com imagens em tons de cinza e retorna uma imagem binarizada, após filtragem usando o método selecionado.

2. Dilatação:

Dilatação é uma operação binária que, quando realizada sobre imagens negativas (i.e. com muitos *pixels* com valor 0), permite a expansão das áreas brancas (*pixels* com valor 1). Para isso, é utilizado um “elemento estruturante”, que define a morfologia da dilatação [68].

A dilatação de uma imagem binarizada pode ser realizada em *Matlab*® através da função ‘*imdilate*’. E o método ‘*disk*’, com raio variável, foi o elemento estruturante da função ‘*strel*’ utilizados para dilatação da íris.

3.3.4 Redução da densidade de resolução & Conversão

A última fase do módulo PDI consiste em operações que impactam diretamente na entrada de dados da RNA do módulo IA.

A função ‘*imresize*’ do *Matlab*® foi utilizada com o método ‘*nearest*’ para redimensionar as imagens para uma fração da imagem original, i.e., como miniaturas destas, e, assim, diminuir o número de *pixels* de entrada na rede, com baixa perda de informação.

Por fim, as imagens resultantes foram exportadas do *Matlab*® em arquivos binários, através da função ‘*fprintf*’. Para viabilizar a leitura das imagens pela RNA do módulo IA, os dados foram gravados em formato *.arfff*, descrito no Apêndice A.

3.4 Módulo IA: Inteligência Artificial

Este módulo foi desenvolvido com o objetivo de reconhecer os padrões (de fixação em um ponto ou piscado) nas imagens finais e mapeá-las para as respectivas classes dos movimentos (i.e., em coordenadas de observação quando houver).

3.4.1 *Multilayer Perceptron*

A técnica de IA selecionada foi Redes Neurais Artificiais [11,12]. E o modelo de RNA escolhido foi a arquitetura *Multilayer Perceptron*, um classificador neural bastante conhecido e utilizado [11,12].

A seleção da arquitetura e parâmetros da rede MLP empregada (e.g., número de neurônios na camada escondida, taxa de aprendizado etc.) foi baseada nos resultados obtidos *ah hoc* da execução de fases da Figura 19.

O treinamento da rede é iniciado e segue monitorado pelo teste da validação. Após validada, a rede é testada com um novo conjunto de dados. Se os resultados obtidos forem satisfatórios, a arquitetura é escolhida; senão, o ciclo de reconfiguração, treinamento e testes é reiniciado.

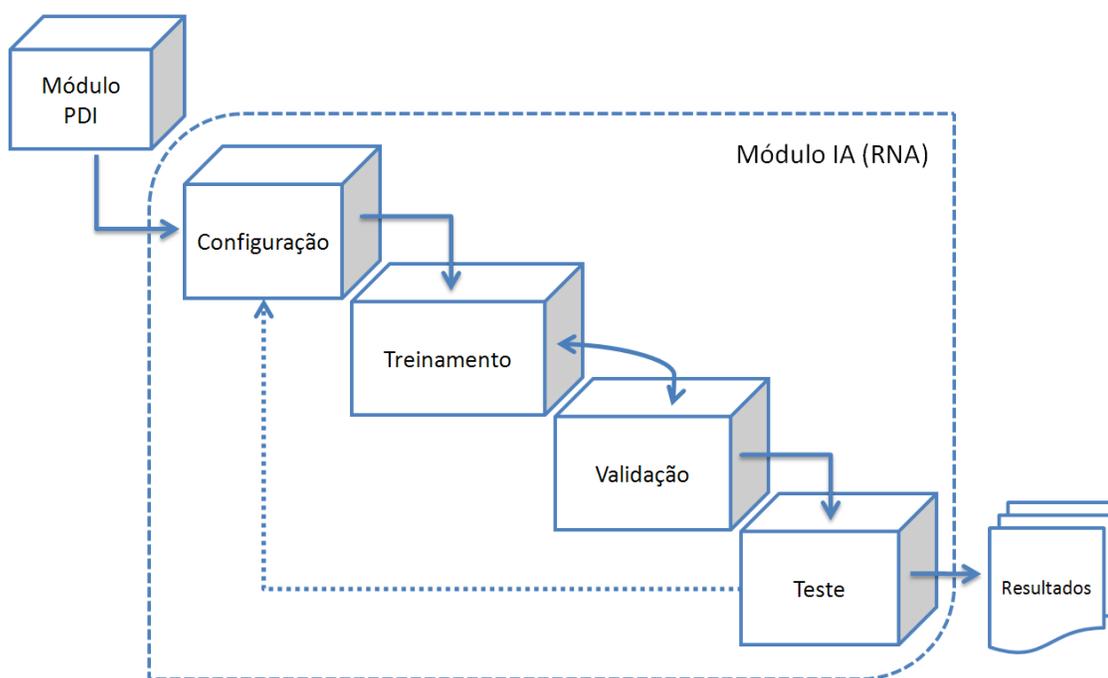


Figura 19. Sequência de modelagem do módulo IA (RNA).

3.4.2 WEKA®

Para realizar a configuração, treinamento, validação e teste da rede, foi utilizado o módulo *Experimenter* da ferramenta *WEKA®*, versão 3 [77] – um pacote de *softwares* com algoritmos de Aprendizagem de Máquina em linguagem de programação *Java* [78]. Além de dar suporte a redes MLP, o *WEKA®* é um *software* muito utilizado pela comunidade científica na aplicação e testes de RNAs.

Os seguintes parâmetros de RNAs [11,12] foram investigados e configurados no WEKA®:

- Porcentagem de dados da base para treinamento, validação e teste;
- Número de neurônios da camada escondida;
- Taxa de aprendizado;
- *Momentum*;
- Número máximo de épocas de treinamento;
- *Threshold* de validação;
- Número de repetições dos experimentos.

A configuração final encontrada para a rede MLP está apresentada na seção 4.3, bem como seus resultados.

3.5 Módulo GUI: Interface Gráfica do Usuário

O módulo GUI tem o objetivo de mapear graficamente os padrões identificados (pelo módulo IA) em caracteres do alfabeto português, permitindo comunicação escrita na tela do computador.

3.5.1 Dimensões

Genericamente, o módulo GUI pode ser descrito como um *grid* de dimensões 4x7, em resolução de aproximadamente 1024x768 *pixels*, contendo 28 quadrados, e um campo superior retangular

Os quadrados definem as classes utilizadas para mapeamento dos movimentos oculares e devem conter caracteres alfabéticos para possibilitar a escrita. A Figura 20 ilustra apenas a estrutura da interface e suas classes.

O campo superior retangular é reservado para exibição dos caracteres selecionados pelo usuário para escrita.

A dimensão 4x7 (28 quadrados) foi definida pois pode representar o número de letras do alfabeto português básico (23 letras) e alguns caracteres extras (5 ao todo). Outras dimensões com valores aproximados (e.g. 5x7) foram descartadas,

porque não maximizavam a área de cada classe – característica que auxilia na diferenciação das coordenadas das classes.

1x1	1x2	1x3	1x4	1x5	1x6	1x7
2x1	2x2	2x3	2x4	2x5	2x6	2x7
3x1	3x2	3x3	3x4	3x5	3x6	3x7
4x1	4x2	4x3	4x4	4x5	4x6	4x7

Figura 20. Grid 4x7 de classes do módulo GUI.

Capítulo 4

Resultados

Este capítulo descreve as configurações, testes e resultados obtidos a partir da modelagem, implementação e aplicação dos módulos descritos no Capítulo 3.

4.1 Resultados do Módulo DAQ

4.1.1 Dados para base de imagens “Alfabeto”

Um arquivo de vídeo pode ser descrito com uma sequência de quadros, i.e, de imagens em movimento. Assim, para a aquisição de dados, a utilização do dispositivo selecionado limitou-se à gravação de amostras de vídeos de movimentos de olhos, e não à utilização do sistema completo proposto em [49,67].

A obtenção dos vídeos deu-se da seguinte forma:

1. Um *grid* de dimensões 4x7, contendo caracteres alfabéticos em cada quadrado, foi exibido na tela de 15' do laptop a uma resolução de 1280 x 960.
2. O dispositivo em uso foi acoplado devidamente à cabeça do usuário, cuja cabeça manteve-se fixa (assim como um paciente portador de LIS clássica), à distância de aproximadamente 30 cm da tela do computador.
3. Solicitou-se ao usuário que observasse quadrado a quadrado do *grid* pausada e sucessivamente, percorrendo linha a linha até o final do *grid*.
4. Em cada quadrado, os olhos do usuário realizaram movimentos de busca suave na área interna do quadrado. Esses movimentos possibilitam pequenas variações entre as imagens obtidas a partir de um mesmo quadrado e, assim, maior variedade dos dados para a RNA.
5. Entre quadrados, os olhos realizaram movimentos sacádicos ou de piscado – este último, sinalizando explicitamente a troca de quadrado.

Por fim, os vídeos obtidos apresentam as seguintes características técnicas, listadas na Tabela 2:

Tabela 2. Informações técnicas sobre os vídeos gerados.

PARÂMETROS	VALORES
Formato do arquivo	.avi
Compressão de vídeo	nenhuma
Número de quadros/segundo	30
Tipo da imagem	<i>truecolor</i> (RGB)
Resolução vertical (altura)	480 <i>pixels</i>
Resolução horizontal (largura)	680 <i>pixels</i>

4.2 Resultados do Módulo PDI

4.2.1 Extração & Classificação dos quadros

Após a extração dos quadros, utilizando o Corel® e o Matlab®, o passo seguinte foi classificar as imagens obtidas de acordo com as suas respectivas coordenadas de observação no *grid*, gerando a base inicial de imagens “Alfabeto”.

O número de quadros de cada classe é descrito na Tabela 3. A Figura 21 ilustra amostras da primeira e última classes, respectivamente; e a Figura 22, imagens diferentes de uma mesma classe.

Tabela 3. Classes da base de imagens “Alfabeto”.

Classe	Nº de quadros						
1x1	142	2x1	151	3x1	123	4x1	147
1x2	199	2x2	153	3x2	157	4x2	189
1x3	189	2x3	147	3x3	164	4x3	157
1x4	197	2x4	154	3x4	164	4x4	186
1x5	153	2x5	169	3x5	152	4x5	154
1x6	154	2x6	194	3x6	183	4x6	161
1x7	160	2x7	162	3x7	201	4x7	208



Figura 21. Amostras de imagens das classes 1x1 e 4x7 da base “Alfabeto”.



Figura 22. Amostras de imagens de uma mesma classe: 2x6.

Duas classes extras também foram geradas, segundo a Tabela 4. As estatísticas finais da base de imagens ‘Alfabeto’ são ilustradas na Tabela 5.

Tabela 4. Classes extras da base de imagens “Alfabeto”.

Classes extras	Nº de quadros
Piscado	373
Sacádicos longos	32

1. Classe “**piscado**”: contém todas as imagens em que a pálpebra recobre a pupila.
2. Classe “**sacádicos longos**”: contém as imagens do olho em situações como transição (quebra) de linha e outros movimentos mais bruscos, ou seja, imagens que não puderam ser classificadas dentro do *grid*.

A Figura 23 ilustra imagens das classes extras da base “Alfabeto”. Note o formato levemente elipsoidal da íris na imagem à direita, devido a um movimento sacádico longo na quebra de linha do quadrado 1x7 para o quadrado 2x1.



Figura 23. Amostras de imagens das classes “piscado” e “sacádicos longos”.

Tabela 5. Informações técnicas sobre a base de imagens “Alfabeto”.

PARÂMETROS	VALORES
Formato do arquivo	.bmp
Compressão de imagem	nenhuma
Tipo da imagem	<i>truecolor</i> (RGB)
Resolução vertical (altura)	480 <i>pixels</i>
Resolução horizontal (largura)	680 <i>pixels</i>
Número de classes	30
Número total de imagens (aprox.)	5040
Número médio de imagens por classe	168

4.2.2 Pré-processamento das imagens

Os resultados do pré-processamento são ilustrados a seguir:

1. **Rotação:** O melhor ângulo de rotação encontrado foi 25° . A Figura 24 ilustra o resultado da rotação: uma imagem de dimensão 785×709 pixels.
2. **Redução da resolução espacial:** Em relação à imagem rotacionada, os valores ideais de corte encontrados foram: $X_{min} = 10$, $Y_{min} = 250$, Largura = 400 e Altura = 170. A Figura 25 exhibe o resultado da redução espacial.
3. **Conversão para tons de cinza:** a resolução de cores das imagens foi reduzida de 24 bits, i.e., 16.777.216 cores em RGB, para 8 bits, i.e., apenas 256 cores em tons de cinza. A Figura 26 ilustra o resultado final deste pré-processamento e o histograma da amostra.



Figura 24. Imagem original rotacionada em 25° .



Figura 25. Imagem com resolução espacial reduzida.

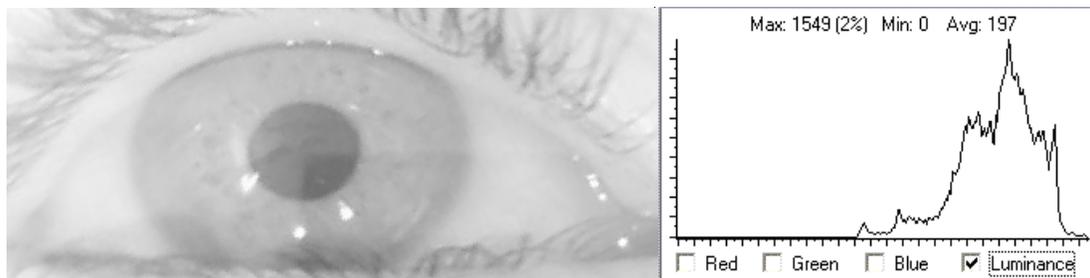


Figura 26. Imagem final em tons de cinza e seu histograma à direita.

4.2.3 Segmentação da pupila e íris

1. Detecção de bordas & Binarização:

Dentre os filtros de detecção de borda examinados, o melhor resultado foi obtido com o filtro *Canny*, com *threshold* (valor de sensibilidade ao contraste) de aproximadamente 0,5. A Figura 27 ilustra o resultado dos filtros *Sobel* (segundo melhor resultado, à esquerda) e *Canny* (à direita), para uma mesma imagem. A binarização é realizada automaticamente durante a chamada da função.

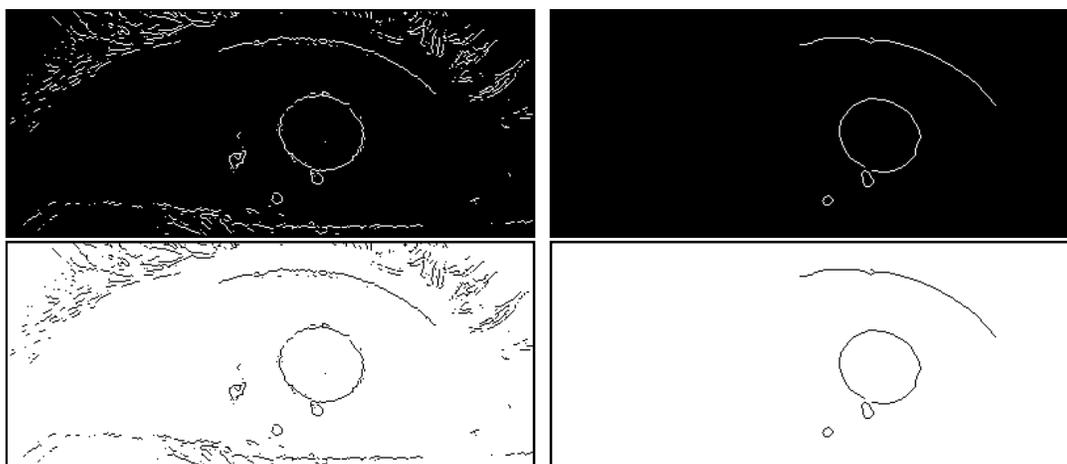


Figura 27. Amostras de segmentação da pupila e íris e binarizações, realizadas pelos métodos *Sobel* e *Canny*, da esquerda para a direita.

2. Dilatação:

A Figura 28 ilustra a imagem com filtro *Canny* após dilatada. A dilatação é necessária para evitar perda de informação significativa durante a etapa de redução a seguir.



Figura 28. Imagem dilatada.

4.2.4 Redução da densidade de resolução

A redução foi realizada para diminuir o número de entradas de dados para o módulo IA. A Figura 29 ilustra a imagem dilata (401 x 701 *pixels*) com resolução reduzida para 10% e 15% respectivamente. Comparando com a imagem reduzida em 85%, a matriz de *pixels* da imagem original, que possuía 326.400 *pixels* (680x480), agora apresenta apenas 1.500 elementos (60x25 *pixels*), preservando as informações de posição da pupila e da íris em relação à imagem.

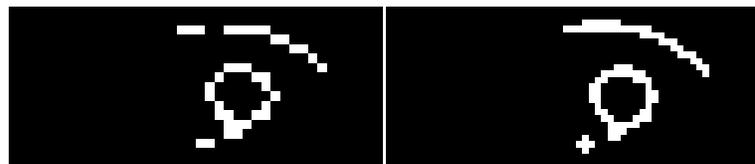


Figura 29. Imagens com resolução reduzida. Da esquerda para a direita: 40x17 e 60x25 *pixels*, respectivamente.

4.2.5 Abordagem básica

Embora as técnicas clássicas de processamento e segmentação de imagens tenham sido eficazes apenas no problema de detecção da íris, o resultado final das imagens não foi satisfatório para o módulo IA. A quantidade de padrões utilizados na entrada da rede (60 x 25 *pixels* = 1500 atributos binários) ainda tornaria o processamento lento.

Em virtude disso, a seguinte decisão de projeto foi tomada: utilizar uma nova sequência de técnicas básicas de processamento de imagem e parâmetros para detectar as informações mais relevantes quanto à posição da íris. O objetivo desta proposta foi buscar obter imagens finais com menor resolução possível e auxiliar ainda mais o processamento do módulo IA.

A seqüência das etapas manteve-se a mesma, contudo houve mudanças de valores (definidos experimentalmente) nas seguintes etapas:

- Novo valor de redução da resolução espacial: $X_{min} = 250$, $Y_{min} = 250$, Largura = 310 e Altura = 90;
- Não utilização de filtros de detecção de bordas;
- Binarização realizada através da função do *Matlab*® 'im2bw', com limiar de 0,1, seguida de negatização (inversão dos *bits*) da imagem.
- Novo fator final de redução da densidade de resolução: 10%.

As Figura 30 e 31 ilustram os novos resultados das operações realizadas.



Figura 30. Novos resultados das operações de binarização, dilatação, redução espacial e inversão, da esquerda para a direita, respectivamente.

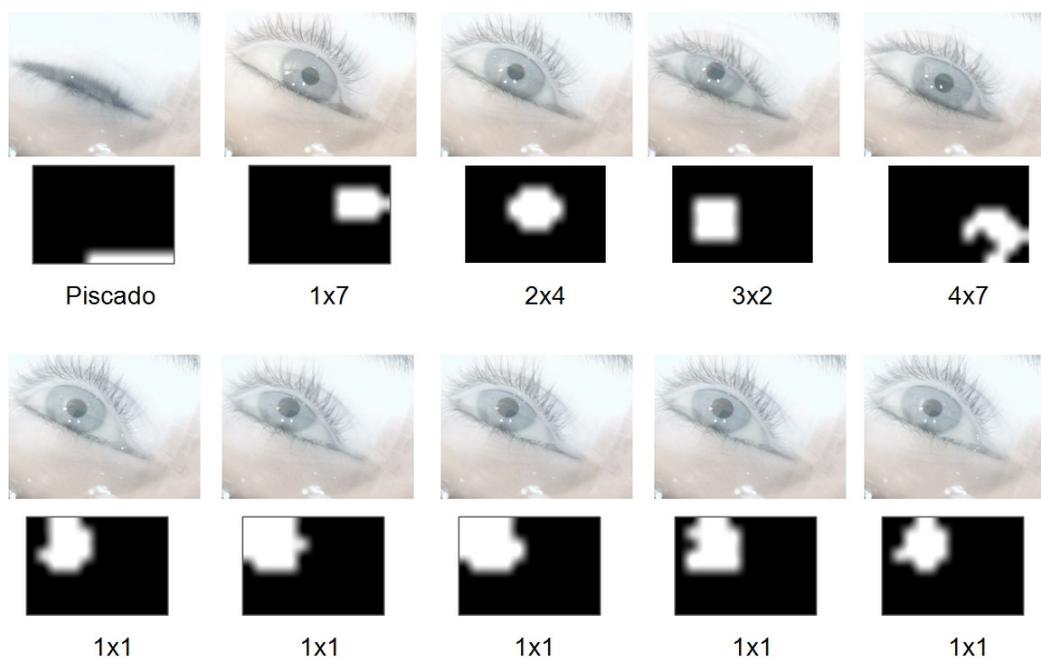


Figura 31. Comparação dos resultados finais para classes diferentes (acima) e para uma mesma base (abaixo).

Uma comparação final das imagens geradas a partir do módulo PDI é ilustrada na Figura 32. As dimensões exibidas são meramente ilustrativas. A imagem original (superior) possui 680 x 480 *pixels*; a imagem final negativada (abaixo, à direita), 13 x 9 (117) *pixels*.

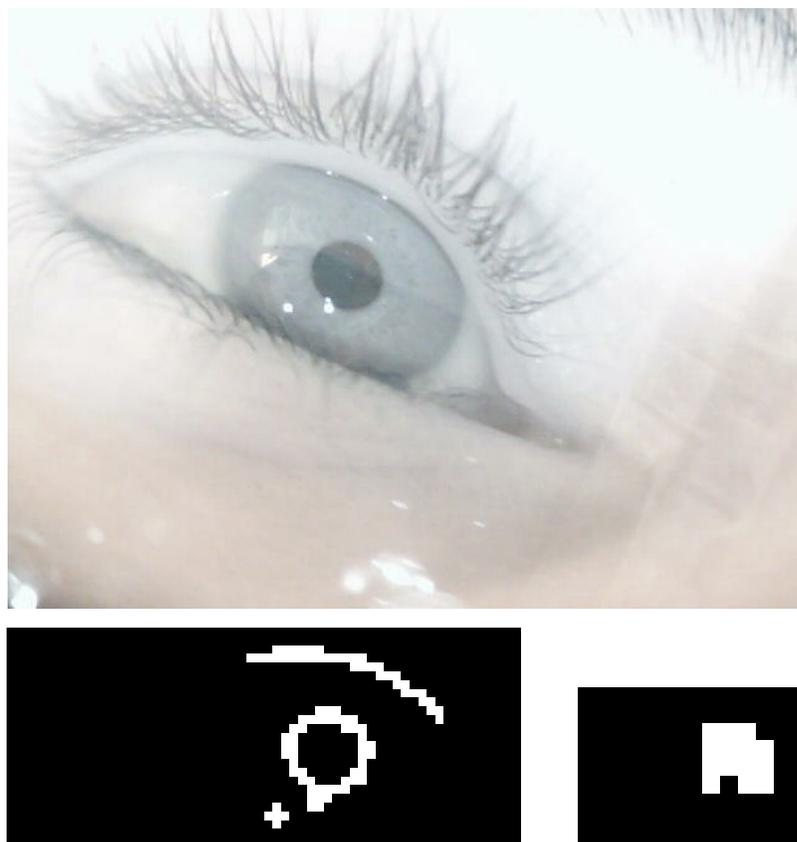


Figura 32. Comparação entre os diferentes padrões de imagens gerados pelo módulo PDI: imagem original acima; segmentada à esq.; básica à dir.

4.2.6 Conversão das imagens

As matrizes das imagens finais foram convertidas para formato `'.arff'`, conforme a seção 3.3.4.

Convencionou-se a inversão dos bits das matrizes antes da gravação dos arquivos binários (conforme descrito na abordagem básica da seção 4.2.5), para que os *pixels* brancos (*bit*=1) pudessem representar a pupila com uma região ativa para a RNA do módulo a seguir. A Figura 30 ilustra a região da íris ativa em 1 (branco).

4.2.7 Geração da base binária 'alfabeto.arff'

Após a conversão de todas as imagens para formato binário, apenas cerca de 300 matrizes de imagens foram selecionadas para compor a base 'alfabeto.arff'. Esse número indica exatamente 10 imagens para cada classe do limite 4x7 e 20 imagens de piscados. A classe "sacádicos longos" não foi inclusa, uma vez que não permitia mapeamento no *grid* do sistema.

A Tabela 6 exibe as estatísticas básicas da nova base. O Apêndice A exibe mais detalhes sobre o seu formato.

Tabela 6. Informações técnicas sobre a base binária "Alfabeto".

PARÂMETROS	VALORES
Nome da base (<i>relation</i>)	alfabeto
Formato da base	.arff
Número de atributos	13x9 (117) <i>pixels</i>
Número de classes	29 (4x7+1)
Número de instâncias	300
Ordenação das instâncias	Não (aleatória)

4.3 Resultados do Módulo IA

4.3.1 Treinamentos e Testes

A melhor combinação de parâmetros obtida experimentalmente para a rede MLP, em simulações realizadas no módulo *Experimenter* do WEKA®, utilizando a base 'alfabeto.arff', é listada na Tabela 7.

Tabela 7. Melhor configuração de parâmetros obtidos para a MLP.

PARÂMETROS	VALORES
Dados da base para:	
• treinamento	230
• validação	40
• teste	30
Neurônios da camada de entrada	117 (<i>pixels</i>)
Neurônios da camada de saída	29 (classes)
Neurônios da camada escondida	29 (prop. à saída)
Taxa de aprendizado	0,3
<i>Momentum</i>	0,2
Épocas de treinamento	60 (máximo)
<i>Threshold</i> de validação	10
Repetições dos experimentos	30 (vezes)

Esses parâmetros foram selecionados da rede que obteve os melhores resultados nos teste finais. A Tabela 8 ilustra resultados obtidos a partir de 30 simulações para essa rede. As simulações ocorreram da seguinte forma:

1. O subconjunto de instâncias da base reservado para o treinamento foi apresentado à rede durante essa fase, que realizou o ajuste dos pesos das conexões entre os neurônios.
2. Ainda na fase de treinamento, a validação foi iniciada, utilizando um outro subconjunto da base, a fim de avaliar a rede ainda durante o treinamento.

3. Atingida a condição de parada do treinamento (i.e. número máximo de épocas de treinamento e *threshold* de validação), a simulação seguiu a fase de teste, utilizando o último subconjunto da base, para avaliar a generalização da rede.

Tabela 8. Melhores resultados da rede MLP configurada.

	Média	Desvio padrão	Mediana
Número de acertos	29,13	1,06	29,00
Porcentagem de acerto	96,13	3,53	96,66
Número de erros	1,17	1,06	0,00
Porcentagem de erro	3,86	3,53	0,00
Erro Absoluto Médio (MAE)	0,02	0,00	0,02
Raíz do Erro Quadrático Médio (RMSE)	0,06	0,00	0,06
Erro Relativo Absoluto (RAE)	30,28	2,97	30,09
Raíz do Erro Quadrático Relativo (RRSE)	36,35	4,80	36,12
Duração do treinamento (segundos)	6,46	3,71	4,73

4.4 Resultados do Módulo GUI

A interface gráfica final foi desenvolvida em linguagem de programação *Java* [78], a mesma linguagem de desenvolvimento do *WEKA*®, e recebe as saídas da rede. Também é possível realizar a integração com o *Matlab*® através de *Java*.

A disposição final dos caracteres na interface gráfica é ilustrada no *grid* da Figura 33. O caractere ‘<’ representa a operação de *backspace* em um teclado normal; e o quadrado em branco, um espaço em branco. O símbolo ‘@’ foi adicionado após relato de um paciente, solicitando que incluíssem o sinal, para que pudesse escrever endereços de e-mail ao se comunicar [79].

Em especial, a disposição das letras difere da Figura 3, por exemplo, seguindo a ordem alfabética. Isso se deve porque, nas placas contendo o alfabeto ditado ao paciente, este deve ser ordenado pela freqüência descendente de uso de suas letras. Isto é, da mais empregada até a menos utilizada, agilizando, assim, a comunicação. Porém, nesta interface controlada pelo usuário, esse requisito não é necessário.

A	B	C	D	E	F	G
H	I	J	L	M	N	O
P	Q	R	S	T	U	V
X	Z	@	?	<	.	

Figura 33. Disposição final dos caracteres da interface gráfica.

4.4.1 Protocolo de comunicação

Como resultado do módulo GUI, foi desenvolvido também um pequeno protocolo de comunicação para utilização do sistema.

O usuário deve confirmar o caractere desejado olhando fixamente para seu quadrado e piscando por aproximadamente 1 segundo. À taxa de captação de vídeo de 10 quadros/segundo, a detecção de mais de 10 imagens da classe “piscado” consecutivamente efetua a seleção do último sinal de imagem não pertencente à classe “piscado”, para escrita desta classe. Essa abordagem reduz os erros de escrita acidental, decorrentes do movimento natural de piscado do olho humano.

Capítulo 5

Conclusão e Trabalhos Futuros

5.1 Discussões e Dificuldades

Este Trabalho de Conclusão de Curso apresentou uma prova de conceito, propondo uma Interface Humano-Máquina inteligente para comunicação escrita de pacientes com Síndrome *Locked-in*.

Pacientes com Síndrome *Locked-in* apresentam um quadro clínico de paralisia rara e grave. “Apesar da profunda inabilidade, eles não querem necessariamente morrer.” [6] Muitos desejam retornar aos seus lares, e até mesmo ao trabalho e apresentam qualidade de vida superior a de pacientes com outros quadros. A barreira da comunicação é, contudo, o maior obstáculo na reabilitação e no retorno desses pacientes.

HCI's baseadas no rastreamento dos movimentos oculares podem ser uma alternativa de comunicação para muitos dos casos de Síndrome *Locked-in* clássica. Os requisitos especiais desses pacientes tanto podem facilitar o projeto de tais sistemas, como tornar seu desenvolvimento uma tarefa peculiar. Em particular, neste projeto, uma vantagem foi desconsiderar movimentos adicionais realizados pela cabeça do usuário durante o rastreamento dos olhos. Um desafio, contudo, foi adequar o projeto a um tipo básico de interação que esse usuário em geral pode apresentar: o piscar dos olhos.

Em especial, técnicas de Visão Computacional e Inteligência Artificial, tais como Segmentação de imagens e Redes Neurais Artificiais, podem garantir HCI's dedicadas e robustas, até mesmo a um custo mais baixo [48,49]. A combinação desses técnicas não necessariamente precisa incluir algoritmos novos, em estado da arte, para garantir resultados satisfatórios. Exemplo disso foi a utilização eficaz, neste projeto, de técnicas como *Multilayer Perceptron* e Binarização – duas técnicas bastante conhecidas e antigas em suas áreas. A utilização de uma câmera simples, devidamente acoplada à cabeça do usuário, também se mostrou eficaz,

dispensando a utilização de imagens finais de alta resolução ou coordenadas tridimensionais no globo ocular.

A utilização de Redes Neurais Artificiais teve grande importância no projeto, em particular, no reconhecimento dos diferentes padrões obtidos. As taxas de acertos das redes simuladas apresentaram resultados muito bons, próximos de 100% nos casos de teste final. E a curta duração dessas simulações implica na possibilidade de utilização, em tempo real, de tais redes ou até mesmo de re-treinamento sempre que necessário.

Como dificuldade, pode-se citar ainda o número razoável de parâmetros obtidos experimentalmente e ajustados até a configuração final do sistema, alguns deles: valor de limiarização das imagens obtidas e número de neurônios da camada escondida da rede.

A escassez de dados disponíveis para mapeamento dos movimentos dos olhos em coordenadas observadas também foi outro desafio. Este, contudo, motivou à criação de uma base de dados própria para testar o modelo proposto. Esta tarefa não estava incluída no planejamento inicial, mas se apresentou desafiadora o bastante para aumentar o interesse pelo tema, ao mesmo tempo que se mostrou ser possível a manutenção das premissas iniciais: a interface ser construída a um custo baixo (mesmo incluindo a geração de dados de treinamento *'in-house'*).

5.2 Melhorias e Trabalhos futuros

Ao final deste trabalho, algumas melhorias e trabalhos futuros podem ser apontados em relação à prova de conceito desenvolvida. São eles:

5.2.1 Adição de módulo de calibração ao sistema

A utilização de um módulo de calibração do sistema é uma abordagem comum entre algumas interfaces pesquisadas [48,49,52]. A calibração permite um pré-ajuste do sistema em relação a mudanças de coordenadas iniciais da cabeça do usuário, às condições atuais de iluminação do ambiente ou até particularidades dos olhos do usuário. A decisão de projeto de utilizar RNA facilitará essa melhoria [53].

Propõe-se o re-treinamento da rede MLP (ou outra arquitetura de RNA a ser analisada), em tempo real nesta fase, nos casos em que as condições da entrada do sistema variem muito. Essa possibilidade é viável, uma vez que a fase de treinamento da rede selecionada apresentou duração relativamente pequena (i.e. na ordem de alguns segundos).

5.2.2 Configuração automática dos parâmetros

Muitos dos parâmetros ajustados experimentalmente poderiam ser configurados automaticamente pelo sistema. O limiar de corte para imagens binarizadas, por exemplo, pode ser obtido de forma automática através da análise do histograma da imagem [33]. Também há estudos para configuração automática dos parâmetros de uma RNA que podem ser incluídos num futuro próximo [11].

Embora a configuração automática do sistema às vezes implique em maior custo computacional para o mesmo, ela dispensa a dependência de configuração via operador e ainda pode tornar o sistema mais flexível e adaptável.

5.2.1 Realização de testes reais

Novos experimentos devem incluir a apresentação de diferentes indivíduos ao sistema, a fim de testar e aumentar sua capacidade de generalização para padrões desconhecidos. Mais testes reais também devem incluir a variação da distância entre o usuário e a tela do computador, a fim de avaliar o comportamento do sistema.

5.2.2 Implementação de um produto final

Por fim, é sugerida a construção de um protótipo experimental que resulte em um produto final da aplicação.

A substituição das rotinas desenvolvidas em *Matlab*® e no *WEKA*® (ferramentas de código interpretado) por um sistema completo desenvolvido em linguagem de programação compilada (e.g. em linguagem C [80]) deve aumentar consideravelmente a velocidade de processamento do sistema. Outra vantagem é tornar os módulos mais independentes e portáteis; e sua integração, instalação e utilização, mais práticas.

Bibliografia

- [1] A. Sutcliffe, *Human-Computer Interface Design*, Springer-Verlag, 1989.
- [2] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 2006.
- [3] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, 2003.
- [4] R. Pressman, *Software Engineering: A Practitioner's Approach*, Boston, Mass: McGraw-Hill, 2006.
- [5] J. Raskin, *The Humane Interface: New Directions for Designing Interactive Systems*, New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000.
- [6] E. Smith and M. Delargy, "Clinical Review: Locked-in Syndrome," *BMJ (Clinical research ed.)*, vol. 330, 2005, pp. 406-9.
- [7] J. Patterson and M. Grabois, "Locked-In Syndrome: A Review of 139 Cases," *Stroke - Journal of The American Heart Association*, vol. 17, 1986, pp. 758-764.
- [8] J.D. Bauby, *O Escafandro e a Borboleta*, São Paulo, Martins Fontes, 1997.
- [9] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Upper Saddle River, New Jersey: Prentice Hall, 2002.
- [10] D.A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Prentice Hall Professional Technical Reference, 2002.
- [11] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 2008.
- [12] A.D. Braga, T.B. Ludermir, and A.C. Carvalho, *Redes Neurais Artificiais: Teoria e Aplicações*, Editora LTC, 2000.
- [13] All and WHO, *DSM-IV: Diagnostic and Statistical Manual of Mental Disorders*, World Health Organization, American Psychiatric Association, 1994.
- [14] S. Laureys, F. Pellas, P. Vaneeckhout, S. Ghorbel, C. Schnakers, F. Perrin, J. Berre, M. Faymonville, K. Pantke, and F. Damas, "The locked-in syndrome :

- what is it like to be conscious but paralyzed and voiceless?," *The Boundaries of Consciousness: Neurobiology and Neuropathology*, vol. 150, 2005, pp. 495-611.
- [15] E.R. Kandel, J. Schwartz, and T.M. Jessell, *Principles of Neural Science*, McGraw-Hill Medical, 2000.
- [16] G. Thibodeau and K. Patton, *Estrutura e Funções do Corpo Humano*, São Paulo: Ed. Manole, 2002.
- [17] A. Andrade, R. Carvalho, R. Amorim, W. Paiva, E. Figueiredo, and M. Teixeira, "Coma e Outros Estados de Consciência," *Rev Med (São Paulo)*, vol. 86, 2007, pp. 123-131.
- [18] M. Glezer, "Morte Encefálica," *Brain*, vol. 2, 2004, pp. 52-54.
- [19] J. Tavalaro and R. Tayson, *Look up for Yes*, New York, NY: Kodansha America, Inc, 1997.
- [20] Wikipedia, "Locked-in Syndrome," Disponível em: http://en.wikipedia.org/wiki/Locked-in_syndrome Último acesso em 1º Junho, 2009.
- [21] E. Films, *Le Scaphandre et le Papillon*, Pathé Renn Productions, 2007.
- [22] R.H. Brown, V. Meininger, and M. Swash, *Amyotrophic Lateral Sclerosis*, Informa Health Care, 2000.
- [23] S. Hawking, *Guia do Leitor para uma Breve História do Tempo de Stephen Hawking*, Rocco, 1994.
- [24] E. Morris, *A Brief History of Time*, Amblin Entertainment, 1991.
- [25] M.J. Hogan, J.A. Alvarado, and J.E. Weddell, *Histology of the Human Eye: An Atlas and Textbook*, WB Saunders Company, 1971.
- [26] A. Yarbus, *Eye Movements and Vision*, Plenum Press, 1967.
- [27] Inmetro, "Óculos de Sol II," Disponível em: <http://www.inmetro.gov.br/consumidor/produtos/oculos2.asp> Último acesso em 1º Junho, 2009.

- [28] D. Atchison and G. Smith, *Optics of the Human Eye*, Butterworth-Heinemann, 2000.
- [29] "The Official Eye Colour Chart 2008," *Disponível em: http://fc04.deviantart.com/fs29/f/2008/062/7/5/eye_colour_chart_by_delPigeon.jpg Último acesso em 1º Junho, 2009.*
- [30] L. Sara, *Why so many eye colors?*, Indiana, EUA: 1997.
- [31] V.S. Ramachandran, "Eye Movements - Encyclopedia of the Human Brain," USA: Academic Press, Elsevier Science, 2002, p. 2616.
- [32] R. Wilson, *Eye Movements and Visual Attention - The MIT Encyclopedia of the Cognitive Sciences (MITECS)*, The MIT Press, 2001.
- [33] J. Gomes and L. Velho, *Computação Gráfica: Imagem*, Rio de Janeiro, RJ: Instituto de Matemática Pura e Aplicada-IMPA, 2002.
- [34] Wikipedia, "Computer Vision," *Disponível em: http://en.wikipedia.org/wiki/Computer_vision Último acesso em 1º Junho, 2009.*
- [35] K.S. Fu and J.K. Mui, "A Survey on Image Segmentation," *Pattern Recognition*, vol. 13, 1981, pp. 3-16.
- [36] N.R. Pal and S.K. Pal, "A Review on Image Segmentation Techniques," *Pattern recognition*, vol. 26, 1993, pp. 1277-1294.
- [37] R.M. Haralick and L.G. Shapiro, "Image Segmentation Techniques," *Applications of Artificial Intelligence II*, vol. 548, 1985, pp. 2-9.
- [38] H. Pedrini and W.R. Schwartz, *Análise de Imagens Digitais: Princípios, Algoritmos, e Aplicações*, 2008.
- [39] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2003.
- [40] E.R. Dougherty and R.A. Lotufo, *Hands-on Morphological Image Processing*, SPIE Press, 2003.
- [41] C.A. Mello, *Filtros Digitais para Imagens*, Notas de aula na Graduação em Engenharia da Computação - UPE, Poli, DSC, 2008.

- [42] M. Egmont-Petersen, D. De Ridder, and H. Handels, "Image Processing with Neural Networks: A Review," *Pattern Recognition*, vol. 35, 2002, pp. 2279-2301.
- [43] Y.T. Zhou and R. Chellappa, "Artificial Neural Networks for Computer Vision," *Research notes in neural computing*, vol. 5, 1992, pp. 1-162.
- [44] G. Kuntimad, H.S. Ranganath, R. Div, B.N. American, and A.L. Huntsville, "Perfect Image Segmentation Using Pulse Coupled Neural Networks," *IEEE Transactions on Neural Networks*, vol. 10, 1999, pp. 591-598.
- [45] W.S. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," *Bulletin of Mathematical Biology*, vol. 5, 1943, pp. 115-133.
- [46] O. Ludwig Jr. and E. Montgomey, *Redes Neurais : Fundamentos e Aplicações com Programas em C*, Rio de Janeiro, RJ: Editora Ciência Moderna Ltda, 2007.
- [47] B. Laurel and S.J. Mountford, *The art of human-computer interface design*, Inc. Boston, MA, USA: Addison-Wesley Longman Publishing Co., 1990.
- [48] M.J. Mashaalany, H.C. Sleiman, J.D. Heneine, and W.M. Katergi, *Design and Development of a Low-cost Eye Tracking System for the Rehabilitation of the Completely Locked-in Patient*, Beirut, Lebanon: 2006.
- [49] T.F. Vieira and E. Fontana, "Dispositivo de Rastreamento de Movimentos Oculares Baseado em Webcam e Iluminação com LED Infravermelho," *XXI Congresso Brasileiro de Engenharia Biomédica*, Bahia, Salvador: 2008, pp. 669-672.
- [50] T. Ishimatsu, N. Irie, and O. Takami, "Computer Interface Device for Handicapped People Using Head Movement," *Proceedings of the IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing*, vol. 1, 1997, pp. 273-276.
- [51] D. Torricelli, S. Conforto, M. Schmid, and T. D'Alessio, "A Neural-based Remote Eye Gaze Tracker under Natural Head Motion," *Computer methods and programs in biomedicine*, vol. 92, 2008, pp. 66-78.
- [52] A. Campana, A. Dud, O. Gambini, and S. Scarone, "An Artificial Neural Network That Uses Eye-Tracking Performance to Identify Patients With Schizophrenia," *Schizophrenia Bulletin*, vol. 25, 1999, pp. 789-800.

- [53] M. Coughlin, T. Cutmore, and T. Hine, "Automated Eye Tracking System Calibration Using Artificial Neural Networks," *Computer Methods and Programs in Biomedicine*, vol. 76, 2004, p. 13.
- [54] V. Raudonis and G. Narvydas, "Discrete Eye Tracking Based on PCA and Neural Classifier," *Human System Interactions, 2008 Conference on*, 2008, pp. 237-241.
- [55] G. Marcone, G. Martinelli, and L. Lancetti, "Eye Tracking in Image Sequences by Competitive Neural Networks," *Neural Processing Letters*, vol. 7, 1998, pp. 133-138.
- [56] I. Processing and C.E. Smith, "Eye Tracking Using Active Deformable Models," *Proceedings of the III Indian Conference on Computer Vision, Graphics and Image Processing*, 2002.
- [57] S. Amarnag, R.S. Kumaran, and J.N. Gowdy, "Real Time Eye Tracking for Human Computer Interfaces," *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*, 2003, pp. 2-5.
- [58] J.R. Charlier, C. Buquet, J.P. Huguex, and B. Degroc, "Visioboard: A New Gaze Command System for Handicapped Subjects," *Medical & Biological Engineering*, vol. 35, 1997, pp. 2-3.
- [59] Visioboard, "Gaze Communication Device for Severely Handicapped Subjects," *Review Literature And Arts Of The Americas*, 2001.
- [60] Metrovision, "Visioboard - Gaze Communication System," *Disponível em: <http://www.metrovision.fr/mv-vi-notice-us.html> Último acesso em 1º Junho, 2009*, 1997.
- [61] I.S. Lab, "ISL - Eye Image Database," *Troy, New York, EUA. Disponível em: http://www.ecse.rpi.edu/~cvrl/database/ISL_IR_Eye_Database.htm Último acesso em 1º Junho, 2009*.
- [62] "BioID - Eye Position Database," *Sachseln, Suíça. Disponível em: <http://www.bioid.com/downloads/facedb/index.php> Último acesso em 1º Junho, 2009*.

- [63] S.S. Ltd., "IrisBase - Smart Sensors Iris Image Database," *Reino Unido, UK*.
Disponível em: <http://irisbase.com/> Último acesso em 1º Junho, 2009.
- [64] LIVE, "DOVES - A Database of Visual Eye Movements," *Texas, EUA*.
Disponível em: <http://live.ece.utexas.edu/research/doves/> Último acesso em: 1º Junho, 2009.
- [65] I. The MathWorks, *MATLAB - The Language Of Technical Computing*.
- [66] CBSR, "CASIA - Iris Image Database," *Beijing, China*. *Disponível em: <http://www.cbsr.ia.ac.cn/IrisDatabase.htm> Último acesso em: 1º Junho, 2009.*
- [67] T.F. Vieira, *Dispositivo de Tecnologia Assistiva Baseado em Processamento em Tempo Real de Imagens do Globo Ocular*, Programa de Pós Graduação em Engenharia Elétrica, UFPE, 2009.
- [68] R.C. Gonzalez, R.E. Woods, and S.L. Eddins, *Digital Image Processing Using Matlab*.
- [69] C. Corporation, "Corel® VideoStudio Pro X2," *Disponível em: <http://www.corel.com/servlet/Satellite/us/en/Product/1175714228541#versionTabview=tab1&tabview=tab0> Último acesso em: 1º Junho, 2009, 2009.*
- [70] T. Lehmann, C. Gonner, and K. Spitzer, "Survey: Interpolation Methods in Medical Image Processing," *IEEE transactions on medical imaging*, 1999.
- [71] J. Sebag, J. Arnaud, G. Lelievre, J.L. Nieto, and E.L. Coarer, "High-resolution Imaging Using Pupil Segmentation," *Journal of the Optical Society of America A*, vol. 7, 1990, pp. 1237-1242.
- [72] J. Huang, Y. Wang, T. Tan, and J. Cui, "A New Iris Segmentation Method for Recognition," *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, 2004.
- [73] H. Proença and L.A. Alexandre, "UBIRIS: A noisy iris image database," *Lecture Notes in Computer Science*, vol. 3617, 2005, p. 970.
- [74] S. Dey and D. Samanta, "An Efficient Approach for Pupil Detection in Iris Images," *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, 2007, pp. 382-389.

- [75] D. Carr, D. Khabashesku, B. Lipinski, and P. Robichaux, "Iris Recognition," 2008.
- [76] K.R. Park, H. Park, B.J. Kang, E.C. Lee, and D.S. Jeong, "A Study on Iris Localization and Recognition on Mobile Phones," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, 2008, pp. 1-13.
- [77] I.H. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, San Francisco: Morgan Kaufmann, 2005.
- [78] "Java 6," *Disponível em: java.sun.com/ Último acesso em: Junho, 2009.*
- [79] A.N. Bamji, A. Primavera, F. Bandini, E.R. Ramsey, H.F. Al-Ahmar, M. Longo, and L.R. Kumar, *Rapid Responses to: Locked-in Syndrome*, British Medical Journal - BMJ Publishing Group Ltd., 2006.
- [80] H. Schildt and R.C. Mayer, *C Completo e Total*, Makron, 1996.

Apêndice A:

Formato da base binária

'alfabeto.arff'

Entre os formatos de arquivo mais utilizados para bases de dados em RNAs, e suportado pelo WEKA®, está o '.arff'. É um arquivo de texto em ASCII que descreve uma lista de instâncias com atributos (padrões) relacionados. E é dividido em duas partes: cabeçalho e dados.

Para a base de dados "Alfabeto", foi definido o seguinte *template* de arquivo '.arff' o WEKA®:

```
% Universidade de Pernambuco
% Escola Politécnica de Pernambuco
% Departamento de Sistemas Computacionais
% "Uma HCI Inteligente para escrita de pacientes com LIS"
% Coordenador: Prof. Dr Fernando Buarque de Lima Neto
% Autor/Data: Amanda Leonel Nascimento / Junho-2009

@relation alfabeto

@attribute 'ELEMENT 1x1' { 1, 0}
...
@attribute 'ELEMENT NXM' { 1, 0}
@attribute 'Class' { 1x1, ..., 4x7 }

@data
0, 1, ..., 1x1
...
```