

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
CÂMPUS CORNÉLIO PROCÓPIO  
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL  
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

EDUARDA MAGANHA DE ALMEIDA

**SISTEMA INTERATIVO PARA TRATAMENTO DE DISTÚRBIOS DA  
LINGUAGEM**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO

2015

EDUARDA MAGANHA DE ALMEIDA

## **SISTEMA INTERATIVO PARA TRATAMENTO DE DISTÚRBIOS DA LINGUAGEM**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Diplomação, do curso de Tecnologia em Análise e Desenvolvimento de Sistema da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Dr. Eduardo Filgueiras Damasceno

CORNÉLIO PROCÓPIO

2015

Dedico este trabalho aos meus pais, Edilson e Lúcia, por todo apoio, compreensão e incentivo em todos os momentos da minha vida.

## **AGRADECIMENTOS**

Certamente por meio destes parágrafos, não será possível agradecer a todas as pessoas, nas quais de algum modo colaboraram ou fizeram parte desta grande realização pessoal, por isso antes de começar peço desculpas aquelas que não estão presentes entre essas palavras e asseguro todo respeito e gratidão ao que fizeram por mim.

Agradeço primeiramente a Deus, por ter concedido a vida e iluminar meus caminhos nessa trajetória.

Agradeço aos meus pais, Edilson e Lúcia, que sem o apoio deles seria praticamente impossível vencer esse desafio.

Agradeço ao Professor Dr. Eduardo Filgueiras Damasceno pela orientação deste trabalho, pela oportunidade de trabalhar com um excelente profissional e pela amizade e confiança adquirida durante esse período de orientação.

Agradeço aos meus tios, Elson e Luiza, pela confiança, incentivo e grande apoio.

Agradeço aos meus avós, Edson e Conceição, pelo incentivo em todas as etapas da minha vida.

Agradeço a todos os professores do curso, que foram importantes para minha vida acadêmica e pessoal.

Enfim, a todos os que por algum motivo contribuíram para a realização deste trabalho.

“Tente uma, duas, três vezes e se possível tente a quarta, a quinta e quantas vezes for necessário. Só não desista nas primeiras tentativas, a persistência é amiga da conquista. Se você quer chegar aonde a maioria não chega, faça o que a maioria não faz.” (GATES, Bill)

## RESUMO

ALMEIDA, Eduarda Maganha. **Sistema Interativo para Tratamento de Distúrbios da Linguagem**. 2015. 74 p. Trabalho de Diplomação – Tecnologia em Análise e Desenvolvimento de Sistemas. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2015.

Este trabalho apresenta uma ferramenta desenvolvida com tecnologia de visão computacional para rastrear movimentos dos dedos de usuários portadores de algum tipo de distúrbio de aprendizagem. A ferramenta possui uma característica de inteligência artificial, implementada com um algoritmo de redes neurais. Neste foi avaliado a potencialidade da aplicação da ferramenta bem como o índice de acertos e erros. Tem como finalidade ser uma ferramenta de auxílio no tratamento dos distúrbios da linguagem.

**Palavras-chave:** Captura de movimento; movimento gestual; distúrbio da linguagem; ferramenta.

## ABSTRACT

ALMEIDA, Eduarda Maganha . Interactive System for Language Disorders Treatment. 2015 74 p. Graduation work - Technology Analysis and Systems Development. Federal Technological University of Paraná. Cornélio Procópio, 2015.

This work presents a tool developed with computer vision technology to track the movements of the holders of users fingers of some kind of learning disability. The tool has a feature artificial intelligence algorithm implemented with a neural network. We evaluated the application potential and the roster of hits and misses. The tool aims to be a tool to aid in the treatment of language disorders.

**Keywords:** Motion Capture; gestural movement; language disorder; application;

## LISTA DE FIGURAS

Figura 1: Interface Natural de Comunicação (Damasceno, 2013).....	14
Figura 2: Captura por Coloração (BURKE et al.,2008).....	15
Figura 3: Captura por coloração em games (CAMERAO et al., 2010) .....	15
Figura 4: Marcadores Reflexivos (Damasceno, 2013) .....	16
Figura 5: Marcadores Fiduciais em sistemas de reabilitação .....	17
Figura 6: Comunicação entre neurônios (BUCKLAND, 2010).....	18
Figura 7: Modelo de um neurônio perceptron (MEDEIROS ,2006) .....	19
Figura 8: Mesa adaptada .....	22
Figura 9: Diagrama de Implantação .....	24
Figura 10: Diagrama de Classe - Módulo PRCM .....	26
Figura 11: Eventos Associados .....	27
Figura 12: Coordenadas do marcador no instante T1 .....	28
Figura 13: Coordenadas do marcador no instante T2 .....	29
Figura 14: Coordenadas do marcador no instante T3.....	29
Figura 15: Coordenadas do marcador no instante T4.....	30
Figura 16: Coordenadas do marcador no instante T5 .....	30
Figura 17: Coordenadas do marcador no instante T6.....	30
Figura 18: Obtendo coordenadas da vogal em x e y.....	32
Figura 19: Acoplamento na horizontal para a vogal 'a'.....	33
Figura 20: Sentido do preenchimento das linhas em bmp2 .....	33
Figura 21: Diagrama de Classes do módulo RNA.....	35
Figura 22: Modelo de um Neurônio Artificial.....	37
Figura 23: Modelo de um Perceptron de camada única com alimentação para frente (feed-forward).....	38
Figura 24: Tela de Ajuda .....	41
Figura 25: Fases do processo (PRESSMAN, 2005).....	42
Figura 26: Módulos do Sistema.....	43
Figura 27: Arquitetura do Sistema.....	44
Figura 28: Fluxograma do Sistema .....	46
Figura 29: Taxa de Acerto do Algoritmo.....	49
Figura 30: Taxa de acertos por letra do algoritmo.....	49
Figura 31: Apresentação das vogais aos usuários.....	50

## LISTA DE GRÁFICOS

Gráfico 1: Tempo de Execução .....	50
------------------------------------	----

## LISTA DE TABELAS

Tabela 1: Funcionalidades do Sistema .....	15
Tabela 2: Saídas alvo para as vogais manuscritas reconhecidas .....	40
Tabela 3: Cronograma .....	52

## LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

3D	Tridimensional
API	<i>Application Programming Interface</i>
ASCII	<i>American Standard Code for Information Interchange</i>
C#	Linguagem de Programação C#
I. A.	Inteligência Artificial
NUI	<i>Natural Interface User</i>
PRCM	Programa de Reconhecimento de Caracteres Manuscritos
OpenAL	<i>Open Audio Library</i>
OpenGL	<i>Open Graphics Library</i>
OpenCv	<i>Open Source Computer Vision Library</i>
R. N.	Rede Neural
UML	<i>Unified Modeling Language</i>
UTFPR	Universidade Tecnológica Federal do Paraná

## SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	<b>14</b>
1.1 JUSTIFICATIVA .....	14
1.2 OBJETIVOS .....	14
1.3 ÁREA DE PESQUISA .....	16
1.4 ORGANIZAÇÃO DO TRABALHO .....	17
<b>2. MÉTODO DE PESQUISA</b> .....	<b>18</b>
2.1 DISTÚRBIOS DA LINGUAGEM .....	18
2.2 ESTADO DA ARTE .....	19
2.2.1 Práxis de Alfabetização .....	19
2.2.2 Tecnologia Educacional para distúrbios da aprendizagem .....	20
2.3 INTERFACE NATURAL DE COMUNICAÇÃO .....	13
2.3.1 Interação Gestual e Visual .....	13
2.4 CAPTURA DE MOVIMENTO COM MARCADORES .....	14
2.4.1 Marcadores Coloridos .....	14
2.4.2 Marcadores Reflexivos .....	16
2.4.3 Marcadores Fiduciais .....	16
2.5 REDE NEURAL .....	17
<b>3. TECNOLOGIAS E FERRAMENTAS</b> .....	<b>20</b>
3.1 TECNOLOGIAS .....	20
3.2 FERRAMENTAS .....	21
3.2.1 Dispositivos .....	21
<b>4. DESENVOLVIMENTO</b> .....	<b>23</b>
4.1 DEFINIÇÃO DAS CARACTERÍSTICAS .....	23
4.2 DIAGRAMA DE IMPLANTAÇÃO .....	23
4.3 MÓDULO PRCM .....	26
4.3.1 Marcador .....	28
4.3.2 Reconhecimento da Vogal .....	31
4.4 MÓDULO RNA .....	34
4.5 MÓDULO HELP .....	40
4.6 METODOLOGIA DE DESENVOLVIMENTO .....	41
4.7 MÓDULOS DO SISTEMA .....	43
4.8 ARQUITETURA DO SISTEMA.....	44

4.9 FLUXOGRAMA DO SISTEMA .....	45
<b>5. RESULTADOS E DISCUSSÕES .....</b>	<b>48</b>
<b>6. CRONOGRAMA.....</b>	<b>52</b>
<b>7. CONCLUSÃO.....</b>	<b>54</b>
<b>8. TRABALHOS FUTUROS.....</b>	<b>55</b>
<b>9. REFERÊNCIAS .....</b>	<b>56</b>

## **1. INTRODUÇÃO**

O cenário atual vem sendo determinado pelo uso de tecnologias, informações e conhecimento. Neste contexto globalizado, a educação geral enfrenta sérios problemas no processo de ensino e aprendizagem. Crianças e jovens deparam-se com a escassez de recursos e a ausência de acesso às tecnologias que auxiliem na aprendizagem. (LUCZYNSK, 2002)

O uso de novas ferramentas tecnológicas aliados à medicina provocam mudanças em todos os setores da sociedade. Por exemplo, os novos olhares da educação e saúde permitem adaptar tradicionais formas de ensino e do diagnóstico como novos modelos fortemente baseados em recursos computacionais, utilizando ferramenta de apoio tanto nas atividades de ensino como no diagnóstico médico.

Desta maneira, as gerações atuais crescem em um ambiente cada vez mais dependente de tecnologias computacionais. Ainda assim, os principais assuntos acadêmicos (leitura e escrita) devem ser dominados pelas crianças para adquirir conhecimentos de todos os tipos de mídia, leitura e escrita (REGTVOORT; LEIJ, 2007). Um dos problemas apresentados nessa fase é a dificuldade de aprendizagem.

Visando este escopo, os processos de aprendizagem da leitura e escrita podem ser melhorados significativamente, visto que o próprio aluno irá buscar cada vez mais a descoberta de novos ambiente e principalmente novas maneiras de interação com objetos virtuais.

### **1.1 JUSTIFICATIVA**

Acredita-se que por meio de um software multimídia é possível realizar a combinação dos sentidos tátil, visual e motor, ampliando o processo educacional.

### **1.2 OBJETIVOS**

O objetivo principal é desenvolver uma aplicação multimídia que combine o artefato visual, a interação gestual e a animação.

Dado este contexto, é possível desenvolver uma aplicação que realize a captura dos movimentos gestuais e utilizá-los como entrada de dados em um sistema multimídia.

A Tabela 1 apresenta as funcionalidades do sistema, juntamente com sua prioridade, onde a prioridade essencial significa que a não execução do requisito/objetivo implica diretamente no funcionamento do sistema, já a prioridade desejável não implica no funcionamento do sistema, ocorrendo ou não o sistema continuará funcionando normal.

Tabela 1: Funcionalidades do Sistema

<b>ID</b>	<b>OBJETIVO</b>	<b>PRIORIDADE</b>
<b>F1</b>	Reconhecer marcador	Essencial
<b>F2</b>	Reconhecer padrão da letra	Essencial
<b>F3</b>	Exibir imagem da letra	Essencial
<b>F4</b>	Emitir som da letra	Essencial
<b>F5</b>	Permitir a configuração do sistema	Desejável
<b>F6</b>	Exibir Instruções de Uso	Desejável

#### F1 – Reconhecer Marcador

Para conseguir capturar o movimento do usuário é necessário que o mesmo possua um marcador colorido, e este deve ser reconhecido pelo sistema.

#### F2 – Reconhecer Padrão da Letra

Após o reconhecimento do marcador é realizado o desenho da letra, e esta será reconhecido ou não, obedecendo a um padrão que está armazenado em sua rede neural.

### F3 e F4 – Exibir Imagem da Letra e Emitir Som

Após o reconhecimento de letra desenhada pelo usuário, será exibido na a letra desenhada de maneira lúdica (acompanhada por um desenho) e simultaneamente será emitido o som da letra.

### F5 – Permitir a configuração do Sistema

Esse requisito é de prioridade desejável, pois não influenciaria na execução do sistema. Esse botão de configuração foi adicionado ao sistema devido aos problemas de iluminação, de reconhecimento da câmera e entre outros. Com isso o usuário passa a ter gerenciamento do sistema, pode adequar a execução do sistema em qualquer ambiente que estiver sendo necessária apenas a configuração do sistema.

### F6 – Exibir Instruções de Uso

Esse requisito também é prioridade desejável, pois não influenciaria na execução do sistema. Esse botão de instrução foi adicionado como parte extra do projeto, pois exibe uma página .html que mostra como deve ser utilizado o sistema.

## **1.3 ÁREA DE PESQUISA**

Em primeiro momento para o desenvolvimento da ferramenta para auxílio no tratamento de distúrbios da linguagem, foi necessário um estudo de conceitos de como as doenças de distúrbios da linguagem afetam na aprendizagem, os sintomas, como são realizado os diagnósticos, tratamentos, através de inúmeros artigos, sites, leis, entre outros processos. A partir desses estudos e entendimentos foram escolhidas as alternativas tecnológicas que poderiam auxiliar os profissionais da saúde e/ou ensino no tratamento desses distúrbios.

Para esta área de pesquisa foi adotada uma abordagem qualitativa, através de artigos disponíveis online, livros, e softwares relacionados já existentes.

Para construção da ferramenta, foi realizada uma pesquisa de estudos de conceitos e técnicas através do artigo (DAMASCENO, 2013), que utiliza novas tecnologias e ferramentas associada as “velhas” tecnologias e ferramentas, e também sobre redes neurais artificiais (BUCKLAND, 2015). A metodologia foi

implementada utilizando a linguagem de programação C#, na qual foi escolhida pelo fato da afinidade e documentação completa disponível para uso, e o modelo de prototipação, utiliza a captura de movimento agregado a um gerenciamento de tela, que identifica o movimento.

E por fim, para validação e testes do sistema, foram selecionados 10 usuários desconsiderando gênero, idade e classe social, considerando apenas se os mesmos eram ou não alfabetizados. E dentre esses 10, foi possível selecionar apenas os portadores de algum dos distúrbios da linguagem e então apresentar os resultados do sistema.

## **1.4 ORGANIZAÇÃO DO TRABALHO**

Este trabalho está organizado da seguinte maneira, na seção 2 será o método de pesquisa onde é apresentado um contexto sobre os distúrbios da linguagem o estado da arte, a captura de movimento, a rede neural e interface natural de comunicação. Na seção 3 serão apresentadas as Tecnologias e Ferramentas que foram utilizadas para o desenvolvimento; na seção 4, é apresentado o desenvolvimento do sistema, o processo, os diagramas, os módulos, arquitetura, a metodologia para criação da ferramenta. Na Seção 5, será apresentado os resultado e discussões obtidos; a seção 6 apresenta o cronograma para a elaboração; a seção 7 é apresentado a conclusão realizada após o término do trabalho; a seção 8 apresenta um breve contexto dos trabalhos que poderão ser realizados no futuro. E por a seção 8 são as referências utilizadas para a construção deste Trabalho de Conclusão de Curso.

## **2. MÉTODO DE PESQUISA**

Nesta seção serão apresentados os conceitos e técnicas que foram estudados para o desenvolvimento da ferramenta.

### **2.1 DISTÚRBIOS DA LINGUAGEM**

Os distúrbios da linguagem são variados e podem ocorrer em qualquer idade, podem interferir na capacidade do indivíduo de entender, de expressar seus pensamentos, entre outros. As causas são variadas, e podem estar presentes desde o nascimento, ou podem ocorrer na infância ou até mesmo na vida adulta.

As pessoas com distúrbios da linguagem podem ter dificuldade para entender palavras faladas e/ou escritas (NASCENTE, 2010).

Os distúrbios da linguagem são problemas relacionados às habilidades linguísticas, são elas fala, leitura, escrita e audição, especialmente na fase de formação escolar. Esses distúrbios assim como outros distúrbios não são considerados doença trata-se de dificuldades que podem e devem ser tratadas com acompanhamento adequado, atendendo as especificações de cada caso.

Segundo NASCENTE (2010), os problemas na área da linguagem podem ser retardo na fala ou emissões das primeiras palavras (demora a falar e se expressar), deficiência na formação de frases (fala frases de maneira incompleta ou não consegue termina-las), trocas de fonemas, troca de palavras, entre outros fatores.

Os distúrbios que interferem no desenvolvimento e comunicação do indivíduo podem estar relacionados à fala, à linguagem, à audição e/ou à voz. A seguir serão apresentados três tipos de distúrbios da linguagem, a dislalia, a discalculia e a dislexia no qual teve um estudo mais ampliado.

Os distúrbios da linguagem são analisados como uma dificuldade para decodificar tanto a palavra escrita como a oral, o que prejudica a compreensão do material lido e/ou ouvido, alterando o comprometimento e aproveitamento escolar (MOUSINHO; GOLDFELD, 2003). Estes devem ser diagnosticados, preferencialmente, por especialistas, como psicólogos, fonoaudiólogos, psicopedagogos e alguns casos um neuropediatra (SHAYWITZ; FIGUEIRA, 2006).

Existem diversos métodos empíricos que apoiam o tratamento de tais transtornos, todavia, os mais aceitos métodos de ensino são o fônico e o

multissensorial (CAPOVILLA et al., 2004). O método fônico utiliza os fonemas das letras associando-as a desenhos lúdicos para que o aluno faça a associação correta (SOARES, 2003). No entanto o método multissensorial o aluno observa as letras e/ou palavras impressas em uma folha de papel e pronuncia e/ou desenha o movimento de sua escrita (SEBRA; DIAS, 2011). Os dois métodos não dispensam a atenção do terapeuta ou psicopedagogo, que muitas vezes exige uma melhor especificação para a execução do processo e da tecnologia para promovê-lo.

## **2.2 ESTADO DA ARTE**

O estado da arte, no caso a dislexia visa identificar os alguns softwares encontrados e disponíveis para disléxicos desde 2004 até os dias atuais juntamente com a sua abordagem, e também como é o praxis de ensino/alfabetização de disléxicos. E também como deverá ser o manuseio da ferramenta PRCM.

### **2.2.1 Praxis de Alfabetização**

O processo de alfabetização pode sofrer interferências de vários fatores, dessa forma, a dislexia reporta a criança a uma situação de incapacidade na função de ler e escrever com facilidade, afetando assim a compreensão de outras áreas de conhecimento que dependem da leitura e escrita (Fernandes; Penna, 2008).

É muito comum encontrar educadores que praticam o método fônico nas escolas com abordagens que incentivam os educadores a desenvolver habilidades de rima, segmentação fonêmica, discriminação de sons e ensinam as relações entre as letras e os sons (MEDEIROS, OLIVEIRA, 2008).

Diferente método e muito praticado no cotidiano dos educadores é o da escrita multissensorial, uma combinação de distintas modalidades sensoriais no ensino da linguagem escrita. Para Capovilla, 2004, este método facilita a leitura e escrita a partir da conexão entre aspectos visuais, a desenhos ortográficos da letra ou das palavras; os aspectos auditivos, a configuração fonológica; os aspectos sinestésicos da articulação, movimentos e posições para pronunciar sons e palavras.

No Brasil, um procedimento multissensorial bastante difundido e adotado é o fonovisuoarticulatório, conhecido como método da boquinha (SILVA, 2014). Este método é indicado desde a alfabetização de crianças até na reabilitação dos distúrbios da linguagem.

### **2.2.2 Tecnologia Educacional para distúrbios da aprendizagem**

No universo da educação não podemos deixar de citar a tecnologia como um marco de progresso e avanço, aplicada a métodos de ensino e aprendizagem. A cada dia novas tecnologias aplicadas à educação têm influenciado as novas formas de aprendizagem e ampliado oportunidades aos alunos que sofrem de distúrbios da linguagem.

Um exemplo é o método das boquinhas, aprovado como tecnologia educacional pelo MEC, em dezembro de 2009, uma proposta multissensorial, uma nova oportunidade oferecida ao aluno no processo de aprendizagem (ANDRÉ, 2009). A tecnologia educacional vem transformando a tecnologia em recursos de aprendizagem, disponibilizando aos alunos em sala de aula, redes sociais, games, plataformas e aplicativos interativos.

Os softwares interativos de ensino como o CBE (Computer Based Education), permite desenvolver um processo de aprendizagem mais interessante por parte do aluno e a transmissão de conhecimentos por parte do professor (DUARTE et. al, 2013). Esses softwares são desenvolvidos na maioria das vezes utilizando tecnologias multimídia, que ganham cada vez mais espaço no ambiente educacional. Para RAMOS et.al (2007), essas plataformas interativas são ferramentas de apoio, incentivo, motivação e desenvolvimento das habilidades do aluno.

Outros conceitos tecnológicos igualmente estão sendo inseridos no ambiente escolar, como o conceito de gamificação e realidade aumentada. No primeiro caso, estão sendo desenvolvidos jogos educativos que permitem ao aluno realizar diferentes atividades, estimulando o desenvolvimento das habilidades cognitivas e emocionais (ELLWANGER et al. ,2014).

## **2.3 INTERFACE NATURAL DE COMUNICAÇÃO**

Não é necessário olhar muito longe para perceber que a tecnologia está cada vez mais natural e intuitiva. Em um dia típico, muitas pessoas usam toque ou voz para interagir com tecnologia em seus telefones, no caixa eletrônico, no supermercado e em seus carros. A curva de aprendizagem para trabalhar com computadores está se tornando cada vez menor, isso devido a quebra de barreiras nas formas naturais de interação homem-computador graças a formas mais naturais de interação. (CAPOVILLA, 2004)

A Interface Natural de Comunicação, do inglês Natural User Interface, ou simplesmente NUI, é um tipo de interface de usuário que é projetado para sentir o toque, movimento, gesto mais natural do usuário. O objetivo principal do NUI é criar uma interação perfeita entre homem e a máquina, fazendo como que a própria máquina não existisse.

Segundo (DAMASCENO, 2013) Interface natural é a linguagem comum usada por desenvolvedores e designers de interface humano-computador para fazer referência a interface de usuário que é efetivamente invisível, e permanece invisível conforme o usuário evolua utilizando repetidamente a interface, com a evolução do uso de interfaces naturais o usuário chega a algumas interações cada vez mais complexas de maneira autodidata.

### **2.3.1 Interação Gestual e Visual**

A interação gestual localiza os movimentos do usuário e os traduz como instruções. O sistema de jogos de movimento do Nintendo Wii e o PlayStation Move funcionam através de acelerômetros e giroscópios para detectar a inclinação, rotação e aceleração.

Um tipo mais intuitivo de Interface Natural é equipado com uma câmera e um software no dispositivo o qual reconhece específicos gestos e os traduz para ações. O Kinect da Microsoft, por exemplo, é um sensor de movimentos para o console de jogos Xbox 360 que permite os usuários interagirem através de movimentos

corporais, gestos e comandos de fala. O Kinect reconhece corpos e vozes dos jogadores. Reconhecimento gesticular pode também ser usado para interagir com computadores.

Já a Interação visual é a percepção de objetos de interação fora do computador por meio de uma projeção.

Na Figura 2, é apresentando uma visão geral de algumas funcionalidades da tecnologia NUI.



Figura 1: Interface Natural de Comunicação (Damasceno, 2013)

## 2.4 CAPTURA DE MOVIMENTO COM MARCADORES

As técnicas de captura de movimento com marcadores podem ser classificadas em três modelos: com marcadores coloridos (que será utilizada para o desenvolvimento do projeto), com marcadores reflexivos e com marcadores fiduciais (DAMASCENO, 2013).

Nas subseções a seguir será apresentada de maneira individual e sucinta cada uma dessas técnicas.

### 2.4.1 Marcadores Coloridos

A captura de movimentos por coloração é baseada com marcadores coloridos, no qual estes passam por técnicas de processamento de imagens para realizar a análise e variância de cor de determinado conjunto de pixels (DAMASCENO,2013). A identificação de um padrão de cor em formato RGB e a

variação deste padrão na imagem pode ser rastreada. Basicamente é necessária a identificação dos blocos de uma imagem usada como template.

Essa técnica de captura de movimento por coloração tem sido aplicada em sistemas de reabilitação motora dos membros superiores, como exemplifica a Figura 3.

Outro modo muito utilizado para aplicação dessa técnica é em ambiente virtual de um game, como exemplifica a Figura 3, no qual o usuário é estimulado a movimentar os braços no intuito de selecionar ou obstruir o caminho dos atores virtuais animados.

A ferramenta desenvolvida para auxílio no tratamento e distúrbios da linguagem visa estimular o usuário a interagir de maneira lúdica com o sistema, possibilitando que o mesmo meio por meio de luvas coloridas ou coloração nos dedos seja capaz de desenhar a letra pedida pelo profissional.



Figura 2: Captura por Coloração (BURKE et al.,2008)



Figura 3: Captura por coloração em games (CAMERAO et al., 2010)

### 2.4.2 Marcadores Reflexivos

Segundo Damasceno (2013), para avaliação biomecânica e aprimoramento de treinamento esportivo é utilizada uma técnica de captura de movimentos por marcadores reflexivos.

Para exemplificar a aplicação dessa tecnologia, o sistema comercial Peak Motus (RICHARDS, 1999), visto na Figura 4, o qual por meio de marcadores reflexivos e um conjunto de câmeras de luz infravermelha e um sistema de representação em realidade virtual analisa o movimento do usuário para fins de análise biomecânica.

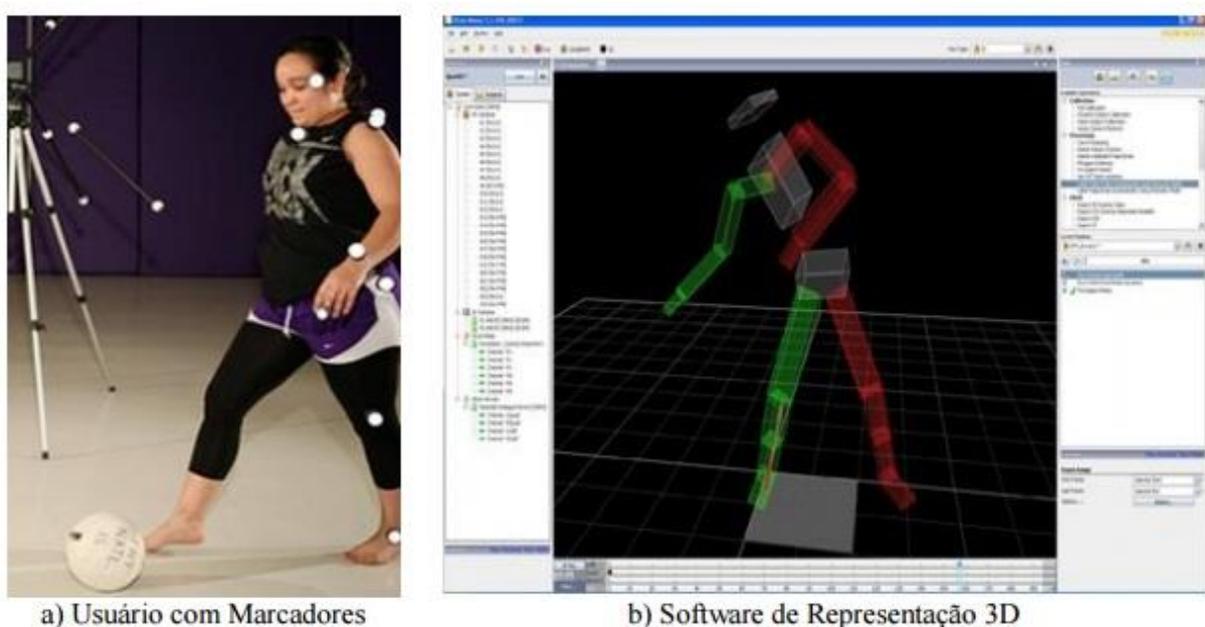


Figura 4: Marcadores Reflexivos (Damasceno, 2013)

A Figura 4 a mostra um usuário realizando um movimento e de qual maneira os marcadores são afixados no corpo. A Figura 5b mostra o funcionamento tridimensional de representação de movimento do sistema de acordo com a movimentação do usuário da Figura 5 a.

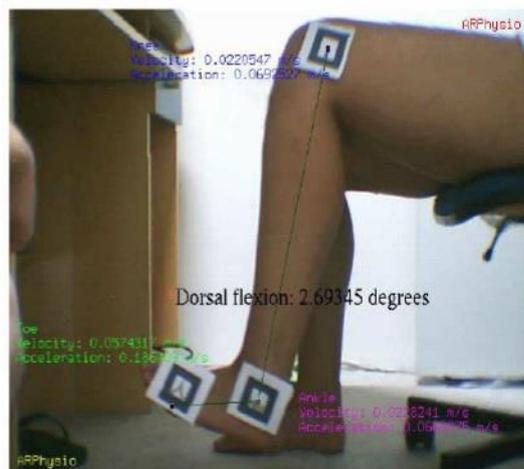
### 2.4.3 Marcadores Fiduciais

O uso de marcadores fiduciais como forma de obter o posicionamento de elementos ou objetos reais em ambientes virtuais é uma técnica que vem ganhando cada vez mais força em sistemas de captura de movimento (WONG,2007).

Por meio da extração das características dos marcadores durante o processo de captura é possível reconhecer o posicionamento tridimensional do usuário no ambiente real (DAMASCENO 2013). Em sistemas de reabilitação essa tecnologia começou a ser aplicada com o intuito de fortalecer a assistência em psicologia e neurologia, promovendo um maior estímulo no controle de desordens cognitivas.

O uso de sistemas de realidade aumentada com marcadores para reabilitação tem se fortalecido na combinação de estímulos motores e cognitivos.

Com as pesquisas realizadas para encontrar trabalhos com marcadores fiduciais foram encontrados os trabalhos de (LIMA et al., 2006) que demonstra o uso de marcadores fiduciais pra indicar a angulação correta e a condução dos exercícios dos pacientes (Figura 6).



a) Sistema de RA para Reabilitação do Tornozelo (LIMA et al., 2006)

Figura 5: Marcadores Fiduciais em sistemas de reabilitação

Segundo Damasceno (2013), por meio de bibliotecas de processamento de imagem que realizam o processamento e o cálculo referencial dos marcadores é possível obter diversos ângulos e por meio da análise do número de quadros capturados, deduzir dados sobre velocidades, aceleração e força aplicada ao movimento.

## 2.5 REDE NEURAL

Explicando rede neural de uma forma simplificada, que dizer uma maneira primitiva de o homem tentar simular o cérebro humano eletronicamente.

Segundo Buckland (2010), nossos cérebros são feitos de aproximadamente 100 bilhões de pequenas unidades chamadas de neurônios. Cada neurônio é conectado a milhares de outros neurônios e se comunica com eles via sinais eletroquímicos. Os sinais que entram nos neurônios são recebidos via junções chamadas de sinapses, essas por sua vez são localizadas no fim dos ramos da célula do neurônio chamados de dendritos. O neurônio continuamente recebe sinais destas entradas e assim desempenha a pequena mágica, ou seja, ela soma as entradas de alguma maneira e então, se o resultado final é maior que algum valor de baliza o neurônio dispara um sinal. A Figura 6 apresenta a comunicação desses neurônios.

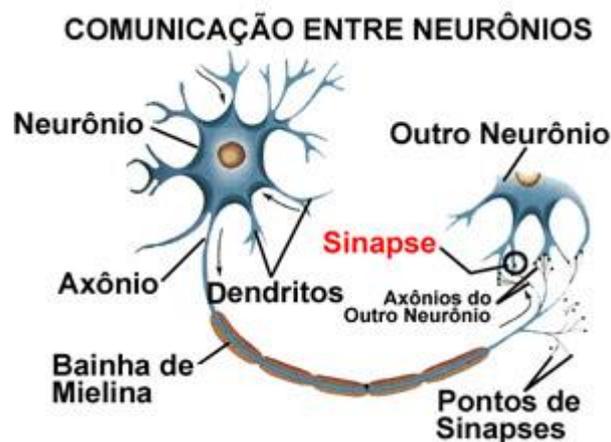


Figura 6: Comunicação entre neurônios (BUCKLAND, 2010)

As redes neurais artificiais são modelos de grafos em que os nós representam os neurônios artificiais e as arestas são conexões de entrada e saída dos neurônios (BUCKLAND, 2008), onde um neurônio tem como função coletar, processar e enviar sinais, a estrutura de uma rede neural simples é composta apenas de um perceptron como pode ser observado na Figura 9.

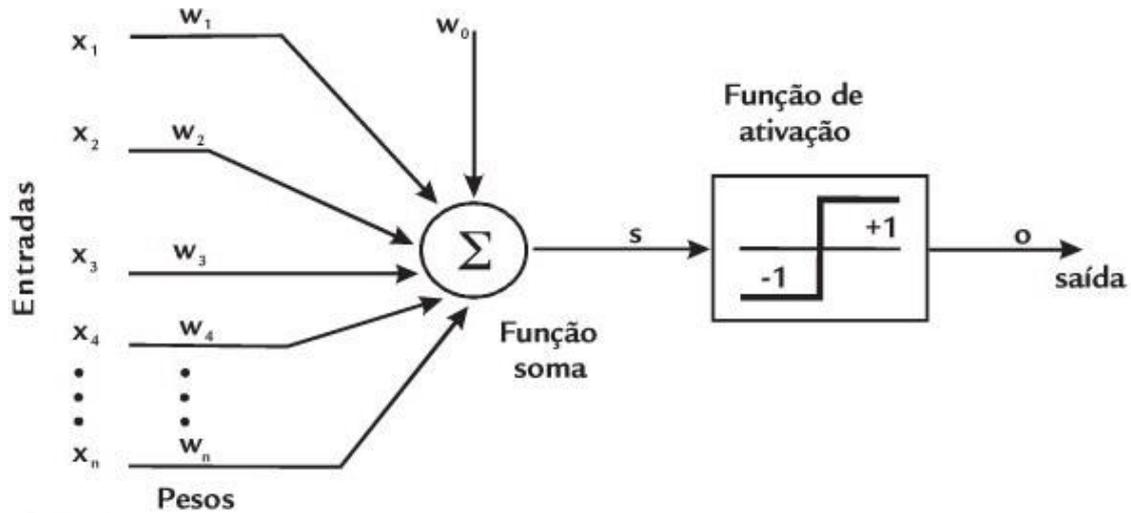


Figura 7: Modelo de um neurônio perceptron (MEDEIROS ,2006)

Os componentes da rede neural são os sinais de entrada  $X_n$ , os quais são elementos do vetor de características, os pesos sinápticos  $W_n$ , a função aditiva, que é um somatório simples das  $N$  entradas já com seus respectivos pesos e a função de ativação na qual se obtém um valor binário denotado de ativo ou não ativo representando a saída.

### 3. TECNOLOGIAS E FERRAMENTAS

Nesta seção serão apresentadas as tecnologias e ferramentas utilizadas para o desenvolvimento da ferramenta PRCM.

#### 3.1 TECNOLOGIAS

Uma das tecnologias utilizadas foram o TouchlessLib, no qual é um projeto SDK rastreamento divertido, webcam multi "toque" objeto. O Touchless encapsula algumas características do processamento de imagens (DAMASCENO, 2013).

Com o uso do Touchless, não é necessário o toque no visor, apenas o efeito do movimento dos dedos capturado por uma webcam. A webcam é responsável por monitorar os eventos de técnicas de rastreamento de uma região de interesse, na qual essa região deve possuir um conjunto de características no formato RGB.

Esse SDK é responsável por transformar a imagem capturada pelo marcador em um bitmap menor, ou seja, este é gerado a partir de um mapeamento feito pela própria biblioteca do touchless da captura do movimento na tela.

A Linguagem de Programação escolhida para a implementação é C# (C-Sharp) devido a disponibilidade de bibliotecas de programação que realizam comunicação com dispositivos.

É uma linguagem orientada a objetos, com segurança de tipos de dados, altamente integrada à plataforma .NET e também por ser uma linguagem com muita flexibilidade.

E a modelagem foi realizada em UML, Linguagem de Modelagem Unificada (do inglês, UML - Unified Modeling Language), que possibilita ao desenvolvedor representar um sistema de uma maneira padronizada.

A UML é uma linguagem visual utilizada para a modelagem de sistemas computacionais utilizando o paradigma de orientação a objetos (GUEDES, 2011, P.19). Foi escolhido a UML devido ao conhecimento da linguagem adquirido ao longo do curso.

## 3.2 FERRAMENTAS

As ferramentas utilizadas para o desenvolvimento foram: Astah, Microsoft Visual Studio e Microsoft Visio.

Astah é uma ferramenta case (ferramenta de auxílio) de criação de diagramas UML, além de outros diagramas, tais como de entidade-relacionamento, diagrama de fluxo de dados e outras funcionalidades úteis à fase de especificação e projeto de um sistema (MARTINS, 2013).

A ferramenta de Astah foi selecionada para uso, pois é de fácil entendimento, o desenvolvedor já possui uma pequena experiência em seu uso, e com esta é possível realizar toda modelagem necessária.

O Microsoft Visual Studio é um ambiente de desenvolvimento e programação das interfaces, especialmente dedicado ao .NET Framework e as linguagens Visual Basic, C, C#, C++, e do ASP.NET.

A Microsoft Visual Studio é uma ferramenta de desenvolvimento de software que auxilia os desenvolvedores a criarem aplicações produtivas e com qualidades destacadas (MICROSOFT, 2013).

Foi escolhida a Microsoft Visual Studio por ser uma ferramenta já conhecida pelo programador e também pelo fato de ser facilmente manuseado.

E por fim, a Microsoft Visio é um aplicativo que serve para criação de diagramas para o ambiente Windows, este foi utilizado apenas para elaboração do diagrama de implantação do sistema (Figura 10).

### 3.2.1 Dispositivos

Os dispositivos utilizados para a construção da ferramenta foram um webcam, um computador e uma mesa adaptada.

O Webcam é uma câmera de vídeo de baixo custo que capta imagens e as transfere para um computador. Pode ser usada para videoconferência, monitoramento de ambientes, produção de vídeo e imagens para edição, captura de objetos, movimentos, entre outras aplicações. Atualmente existem webcams de

baixa ou de alta resolução (acima de 2.0 megapixels) e com ou sem microfones acoplados. Alguns webcams vêm com leds (diodos emissores de luz), que iluminam o ambiente quando há pouca ou nenhuma luz externa.

A maioria dos webcams é ligada ao computador por conexões USB, e a captura de imagem é realizada por um componente eletrônico denominado CCD.

O computador este pode ter inúmeras configurações, desde a mais básica até de elevado padrão, e com distintos sistemas operacionais. A sua utilização no software é de grande importância.

A mesa adaptada nada mais é que, uma mesa comum de madeira, ou outro material, onde o revestimento de sua superfície será uma folha EVA na cor preta, que são folhas do tamanho de uma cartolina revestida por um material macio.



Figura 8: Mesa adaptada

Como pode ser observada na Figura 8, a mesa adaptada precisa ser revestida com um EVA e é necessário o uso de um suporte para ser colocada a webcam.

## 4. DESENVOLVIMENTO

Neste capítulo são descritas quais forma as etapas e decisões necessárias para o desenvolvimento da ferramenta.

### 4.1 DEFINIÇÃO DAS CARACTERÍSTICAS

O trabalho tem como ênfase a captura de movimentos por coloração, nos quais podem ser divididos em 3 módulos: PRCM, onde toda a computação realizada visa à captura do movimento de um marcador dentro da área da imagem da câmera de vídeo e a transformação de sua trajetória no contorno de uma vogal manuscrita que será convertida para um bitmap binarizado (em preto e branco) de dimensões  $20 \times 20$  pixels que será apresentado para o reconhecimento pela rede neural artificial, ou para o treinamento da mesma. O marcador é uma implementação da biblioteca TouchlessLib. Basicamente, ele é algum objeto na imagem capturada pela câmera que contém um conjunto de cores que se destaque do cenário, e que terá os seus movimentos monitorados pelas classes daquela biblioteca. O módulo RNA implementa uma rede neural artificial do tipo **Perceptron**. O processamento da rede neural se dá da seguinte forma: inicialmente a rede deve ser treinada para reconhecer cada vogal manuscrita. Isso se dá inserindo um conjunto apropriado de amostras para cada vogal, e atualizando o aprendizado da rede. Tanto as amostras quanto o aprendizado ficam gravados no Banco de Dados do sistema. Feito isso, ao apresentar um Bitmap para a rede, ela é capaz de verificar a qual das vogais aquele Bitmap corresponde, devolvendo um caractere da tabela ASCII que a represente (a, e, i, o, u). E por fim o Módulo Help gera um pequeno aplicativo que recebe parâmetros externos para abrir um arquivo em formato HTML que compõe o sistema de ajuda para o usuário.

### 4.2 DIAGRAMA DE IMPLANTAÇÃO

A compilação destes três módulos não são os únicos que compõe o projeto, ele gera diversos componentes, como arquivos, bibliotecas, tabelas, executáveis, que podem ser listados no diagrama de Implantação (Figura 10) abaixo:

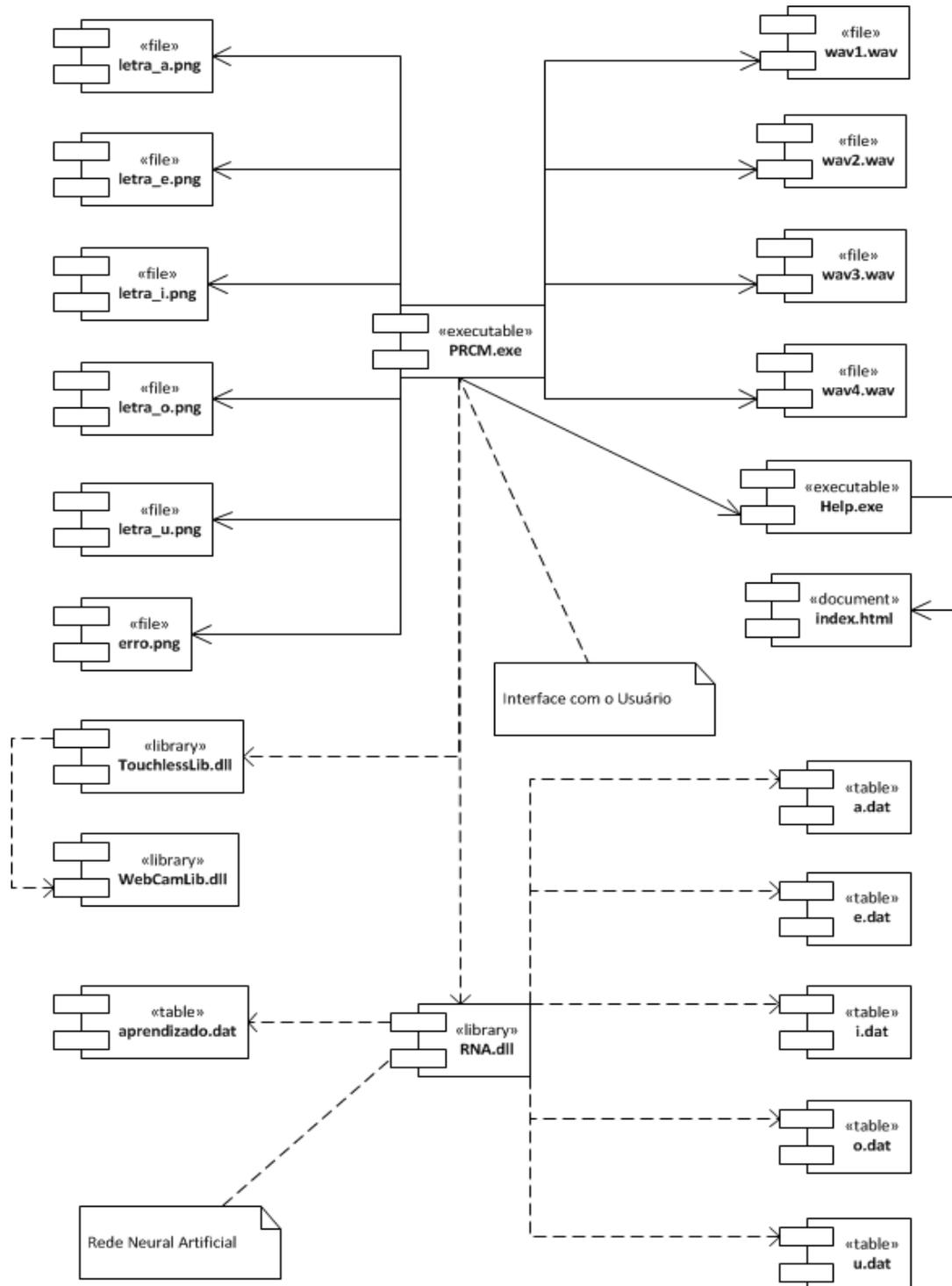


Figura 9: Diagrama de Implantação

A Figura 10 acima apresenta o diagrama de implantação do sistema. Este diagrama tem como objetivo modelar a organização do código fonte, modelar os executáveis, e modelar fisicamente o banco de dados, a seguir é descrito cada componente, o que faz e como estão interligados.

- PRCM.exe: executável principal. Implementa a interface gráfica para a interação com o usuário e faz a captura do movimento do marcador.
- RNA.dll: biblioteca de classes que implementa a rede neural artificial. Essa .dll foi desenvolvida especialmente para o uso deste trabalho, posteriormente será alterada.
- Help.exe: visualizador do arquivo de ajuda do sistema (help).
- TouchlessLib.dll: biblioteca de classes do Touchless. Monitora o movimento do marcador na imagem obtida pela câmera de vídeo.
- WebCamLib.dll: biblioteca requerida por TouchlessLib.dll.
- letra\_a.png, letra\_e.png, letra\_i.png, letra\_o.png, letra\_u.png, erro.png: imagens exibidas na tela do programa PRCM.exe como resultado do processo de reconhecimento da vogal realizado pela rede neural artificial. Cada imagem representa uma vogal, respectivamente a, e, i, o e u. A imagem erro.png será exibida caso a rede neural não tenha identificado a vogal.
- wav1.wav, wav2.wav, wav3.wav, wav4.wav: sons que sinalizam ações no sistema.
- Index.html: arquivo principal do sistema de ajuda.
- a.dat, e.dat, i.dat, o.dat e u.dat: arquivos contendo as amostras para cada vogal, respectivamente, a, e, i, o e u. Cada imagem amostrada no arquivo está convertida num padrão de 1's e 0's, representado os pixels da mesma (1 é branco, 0 é preto).
- aprendizado.dat: aprendizado da rede neural artificial.

### 4.3 MÓDULO PRCM

O Módulo PRCM é a interface principal da ferramenta. Contém todos os elementos de interação com o usuário. Na Figura 11 vemos o diagrama de classes deste módulo.

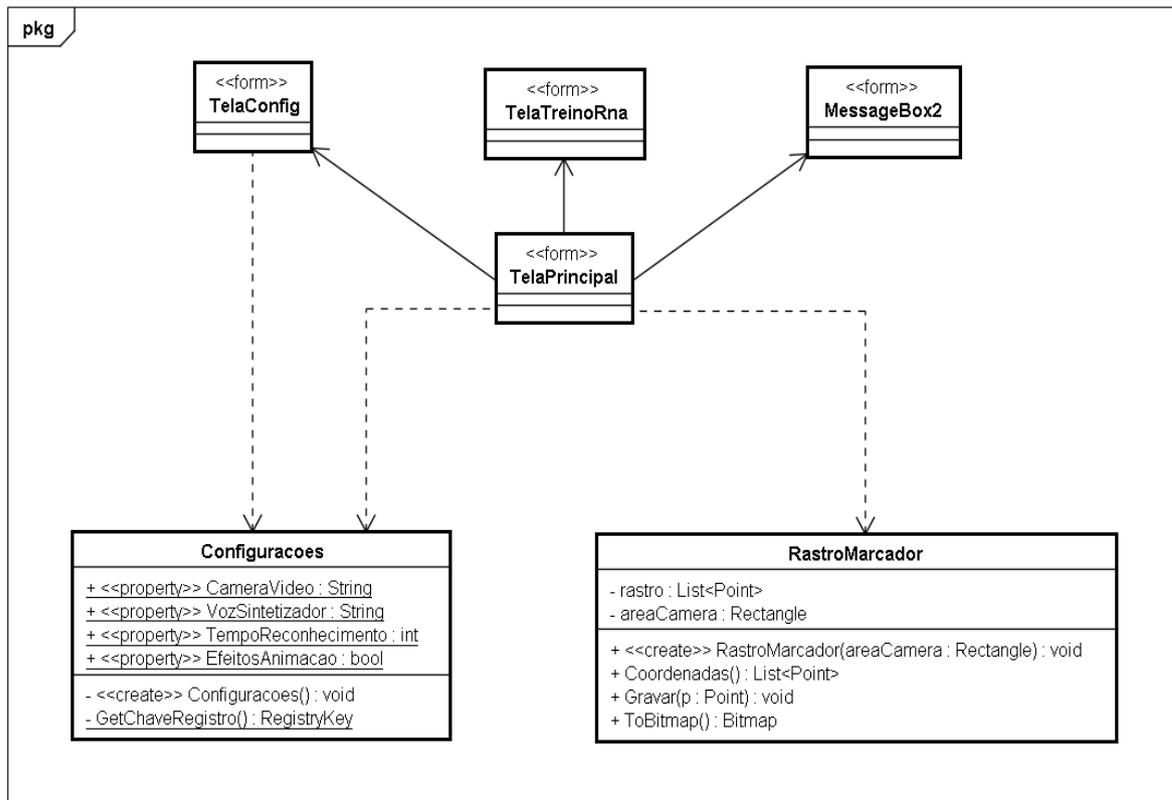


Figura 10: Diagrama de Classe - Módulo PRCM

A Tela Principal fornece a interface gráfica para a interação com o usuário. Toda a funcionalidade do sistema estará disponível através desta tela por meio de controles e da interação com a câmera de vídeo. A Tela Configurações é tela para realizar a configuração do sistema. A Tela TreinoRNA é a tela para sinalizar a obtenção de amostras para o Banco de Dados e para realizar o treinamento da rede neural.

O messageBox2 é a caixa de diálogo para confirmação na inserção de amostras no Banco de Dados. A classe de Configurações serve para a recuperação dos parâmetros de configuração do sistema. O meio pelo qual as configurações são mantidas é o registro do Windows; e por fim a classe RastroMarcador na qual é

utilizada para o registro das coordenadas por onde o marcador passou enquanto se movimentava pela imagem da câmera.

A identificação do marcador é dada pela captura pelo próprio usuário. É necessária a identificação de um padrão de cor RGB e a variação deste padrão na imagem. O rastreamento do marcado pode ser implementado de forma recursiva, necessitando ser inicializado com a posição de origem do objeto rastreado e sua cor. É comum se adotar um detector para identificar o objeto de interesse na imagem de entrada e fornecer a informação referente à inicialização do algoritmo de rastreamento.

De forma a ilustrar este processo, a Figura 12 mostra o sistema em 3 eventos associados.



Figura 11: Eventos Associados

O evento acontece logo na inicialização do sistema e é por meio desse evento que se realiza a marcação do objeto de interesse na imagem. Nota-se que foi utilizada uma fita de cor vermelha para identificar a região da imagem que desejamos associar como um ponto marcador. Já o evento (b), evidencia o ponto da imagem identificado para rastreamento. E por fim o evento (c) mostra o objeto sendo identificado e rastreado. Para o desenvolvimento e testes dos algoritmos, as imagens foram capturadas utilizando o espaço de cores RGB, como resolução 352x288 e com uma taxa de 21 quadros por segundos.

Essa técnica de rastreamento por coloração foi adotada devido a características do sistema e a técnica de ensino abordada (multissensorial), que visa estimular o usuário a interagir com os dedos e sentir a escrita da letra associando com o som lúdico simulador gerado pela multimídia.

A partir do momento em que o marcador é criado, o Timer **tmrMarcador** é iniciado. Ele vai levar 5 segundos para disparar a execução de seu evento. Sua função consiste em dar o tempo necessário para que o usuário posicione corretamente o marcador para desenhar a vogal que ele deseja. Quando esse timer executa seu evento, ele instancia um objeto da classe *RastroMarcador* que vai registrar as coordenadas por onde o marcador vai se mover ao longo da imagem da câmera, também liga o Timer **tmrReconhecimento**, responsável por detectar quando o marcador para de se mover. Este segundo Timer é que dispara o processo de reconhecimento da vogal quando o marcador para de se mover (ou a captura da amostra para o Banco de Dados, se for o caso).

A forma de verificar se o marcador está parado foi descrito na seção 4.3.1.

#### 4.3.1 Marcador

Considerando que o Timer para a captura do movimento esteja configurado para disparar a execução de seu evento a cada 3 segundos (o tempo de execução pode ser configurado, podendo ir de 1 a 5 segundos). Vamos designar cada ciclo de três segundos de T1, T2, T3, ..., Tn.

No primeiro ciclo – T1, o sistema vai registrar o ponto em que o marcador está naquele momento. Devido a ser a primeira vez que o evento é executado, nada mais será verificado.

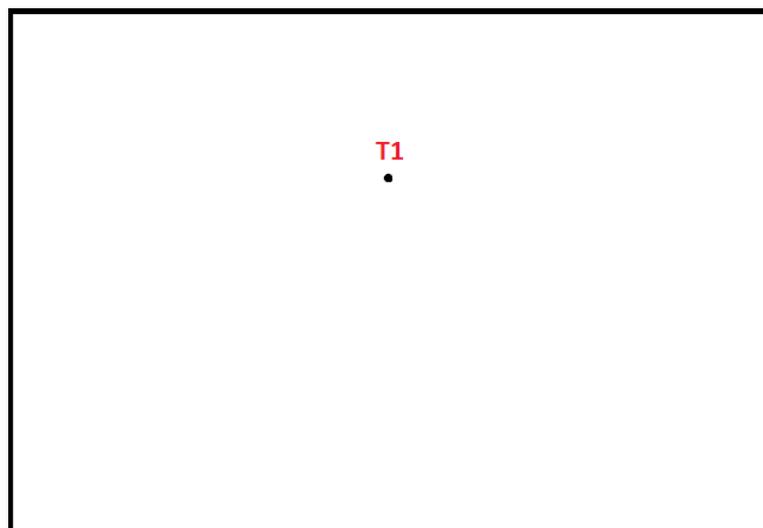


Figura 12: Coordenadas do marcador no instante T1

No segundo ciclo – T2 (Figura 13), ele faz a seguinte verificação: a partir do ponto registrado no ciclo anterior (T1), o marcador em algum momento se moveu para fora de uma pequena área (em verde) delimitada ao redor daquele ponto? Como mostra na Figura 14, ele se moveu. Neste caso, o sistema registra novamente o ponto em que o marcador estava no momento da execução do evento em T2 e prossegue. A cada ciclo seguinte essa mesma verificação será feita. O marcador só vai ter parado de se mover quando ele não tiver saído da pequena área ao redor do ponto registrado no ciclo imediatamente anterior.

Veja nas figuras 13, 14, 15, 16 e 17 a sequência de verificações para o exemplo do desenho da vogal ‘a’.

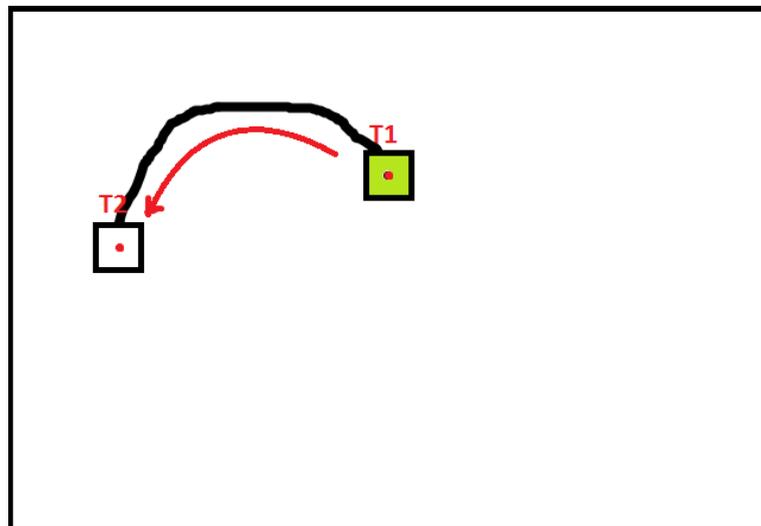


Figura 13: Coordenadas do marcador no instante T2

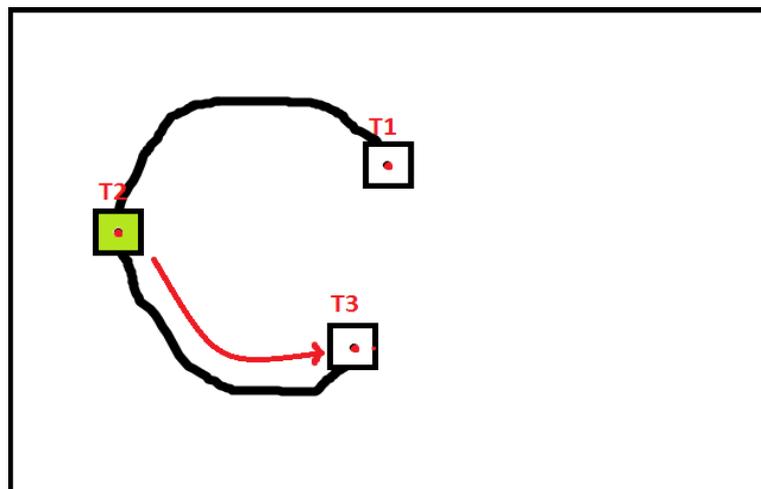


Figura 14: Coordenadas do marcador no instante T3

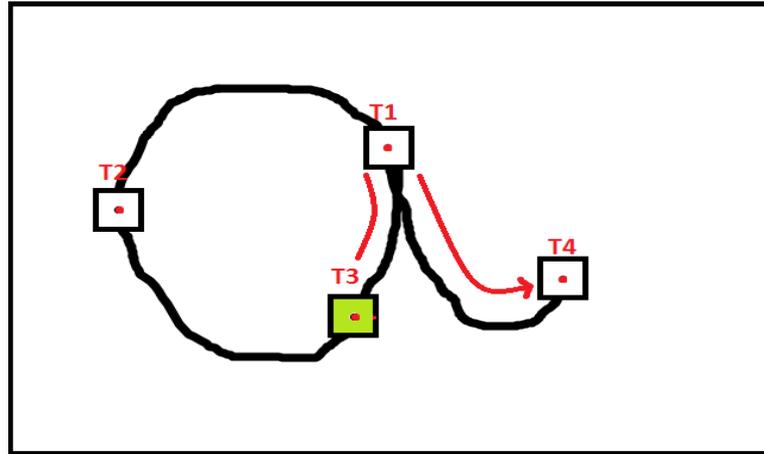


Figura 15: Coordenadas do marcador no instante T4

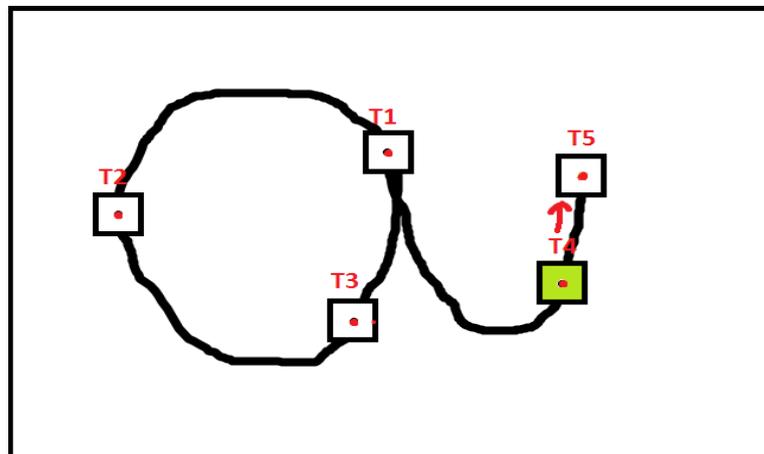


Figura 16: Coordenadas do marcador no instante T5

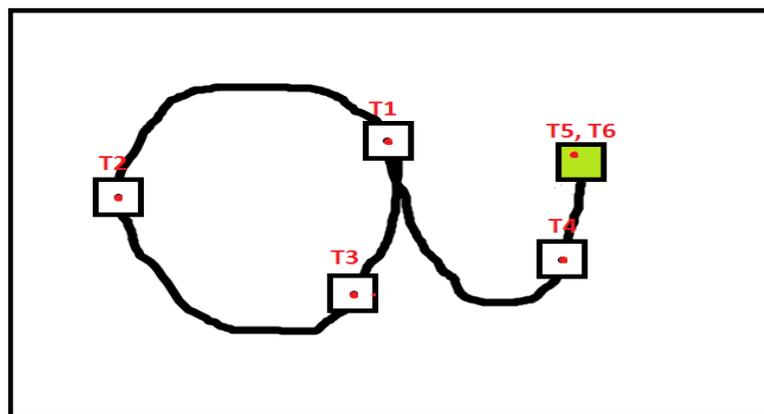


Figura 17: Coordenadas do marcador no instante T6

Nesse exemplo, no ciclo T6, o sistema verificou que o marcador permaneceu dentro da área delimitada ao redor do ponto capturado no ciclo T5. Nesse caso o marcador está parado.

O marcador nunca vai ficar parado no ponto exato em que estava há um ciclo atrás, pois o usuário pode tremer um pouco, e mesmo o próprio componente Touchless gera pequenas oscilações em sua posição. Daí que fazer a verificação de um ponto exato se torna inviável, pois o marcador teria que estar exatamente nele no momento em que o evento do Timer fosse disparado para que se iniciasse o processo de reconhecimento da vogal, o que, devido a essas flutuações aleatórias em sua posição, vai acontecer numa porcentagem ínfima das vezes em que essa verificação for realizada.

#### **4.3.2 Reconhecimento da Vogal**

Para realizar o reconhecimento da letra, é necessário que o marcador esteja parado, assim o sistema irá acionar a rede neural para então reconhecer o que foi desenhado.

Para isso, é obtido o objeto de *RastroMarcador*. Esse objeto tem o registro de todas as coordenadas por onde o marcador passou ao longo do seu trajeto para desenhar a vogal. Essa lista deverá ser convertida numa imagem binarizada (preto e branco) com  $20 \times 20$  pixels que será apresentada à rede neural para reconhecimento. Essa binarização é realizada da seguinte maneira, é gerado uma nova imagem através da captura da webcam, a partir desta é gerado um mapeamento da captura de movimento (neste caso, a imagem gerada é interna é a cópia do marcador), e a partir dessa imagem a rota feita pelo marcador é pintada de preto, ou seja, é necessário circundar para aumenta a rota do caractere. Resumidamente é necessário reduzir uma imagem de determinado tamanho de pixel para um tamanho menor, assim é realizado o rastreamento do movimento para o contorno do caractere.

Com isso será criado um bitmap com as dimensões do retângulo da imagem da câmera e para cada coordenada onde o marcador passou, pinta-se um pixel preto em sua respectiva posição no bitmap. Este pixel será circundado com  $n$  pixels pretos para alargar o contorno da letra.

Feito isso, o próximo passo é isolar as seguintes coordenadas na imagem obtida:

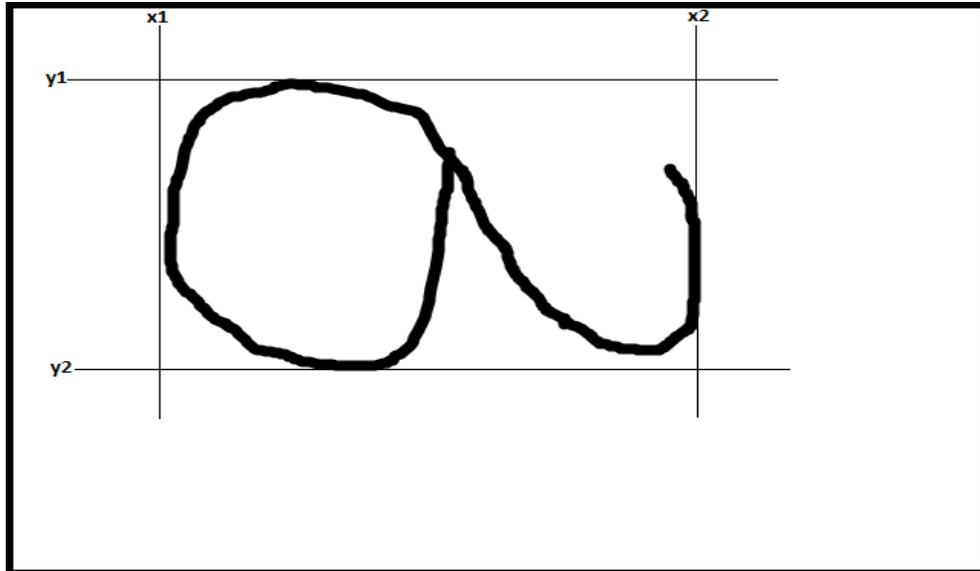


Figura 18: Obtendo coordenadas da vogal em x e y

- **x1** e **x2**: pontos no eixo x aonde foram encontradas, respectivamente, a primeira e a última coluna com um pixel preto. Para obter a dimensão horizontal  $w$  da letra, faz-se:  $w = x2 - x1$ .
- **y1** e **y2**: pontos no eixo y aonde foram encontradas, respectivamente, a primeira e a última linha com um pixel preto. Para obter a dimensão vertical  $h$  da letra, faz-se:  $h = y2 - y1$ .

Feito isso, tem-se as dimensões horizontal ( $w$ ) e vertical ( $h$ ) do desenho da vogal e as coordenadas de onde elas começam e terminam no plano da imagem da câmera.

O próximo passo agora é copiar o desenho da vogal para um segundo bitmap, **bmp2**, com dimensões  $d \times d$  pixels, onde:

$$d = w, \quad w > h$$

$$d = h, \quad h \geq w$$

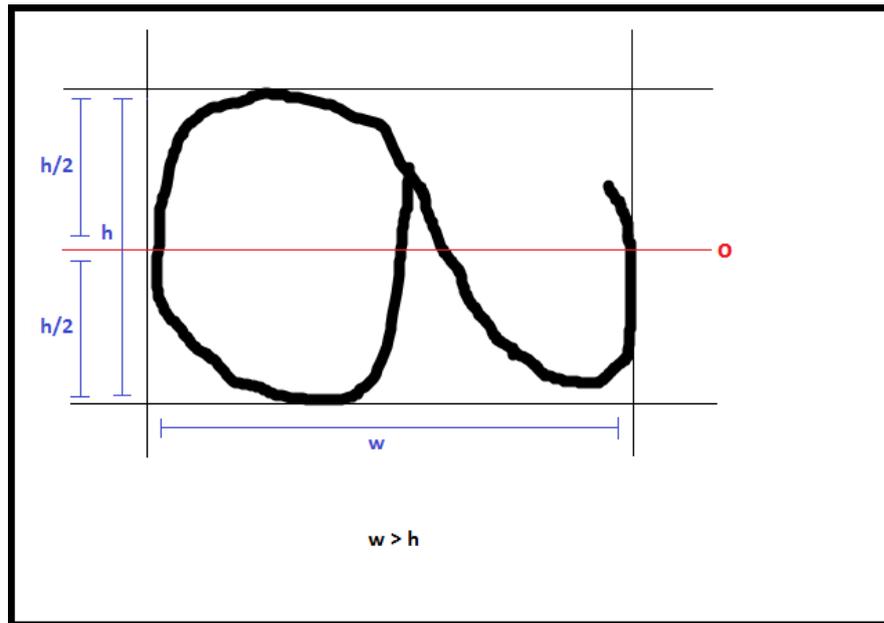


Figura 19: Acoplamento na horizontal para a vogal 'a'

- em `bmp1`, obtém-se a posição no eixo  $y$  do ponto  $o$ , que divide a vogal horizontalmente em duas partes com o mesmo tamanho. Para isso, usa a equação:

$$o = y1 + (h/2)$$

- Lê cada linha (não a linha toda, mas os pontos que vão de  $x1$  a  $x2$ ) abaixo e acima do ponto  $o$  (*inclusive*) em `bmp1` e copia para `bmp2` na respectiva posição. A quantidade de linhas que serão copiadas de ambos os lados do eixo  $o$  em `bmp1` é definida por  $(h/2)$ . Veja na Figura 11 como se dá esse processo de cópia das linhas para `bmp2`.

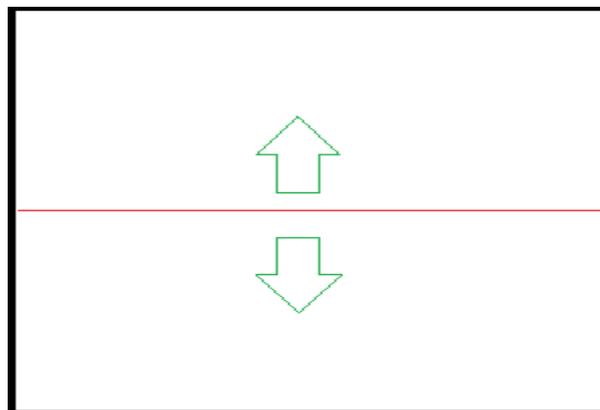


Figura 20: Sentido do preenchimento das linhas em `bmp2`

Pelo fato de a acoplagem ser no plano horizontal, a imagem em **bmp2** estará centralizado neste plano. E como a cópia se dá a partir do centro da vogal em **bmp1** e irradia do centro para as bordas em **bmp2**, ela também estará centralizada na vertical. Ao término deste processo, teremos uma cópia exata em **bmp2** do desenho da vogal em **bmp1**. O próximo passo agora é gerar um terceiro bitmap, **bmp3**, com dimensões  $20 \times 20$  pixels que será apresentado para a rede neural artificial.

O processo de acoplamento no plano vertical ( $h \geq w$ ) funciona exatamente da mesma forma, considerando-se que a cópia será feita tomando como base um eixo que divide **bmp1** ao meio na vertical, e vai irradiar para o lados em **bmp2**.

Gerado o bitmap, ele vai ser apresentado para o processamento pela rede neural artificial. Ela vai processá-lo e devolverá um caractere da tabela ASCII que represente aquela vogal. Caso ela não tenha reconhecido a vogal que se tentou desenhar, vai retornar o caractere de não identificado.

Na Tela Principal, o resultado do processamento da rede neural é apresentado em forma de uma imagem descrevendo a vogal reconhecida (se foi reconhecida), seguida de uma frase descrevendo aquela imagem. Para pronunciar esta frase, o sistema usa da API do Windows, através da importação de **System.Speech.Synthesis**. Caso esteja obtendo amostras para o Banco de Dados, o bitmap será codificado e gravado em arquivo próprio.

#### 4.4 MÓDULO RNA

A Rede neural artificial que fará o reconhecimento dos caracteres manuscritos. Na Figura 22 vemos o diagrama de classes deste módulo:

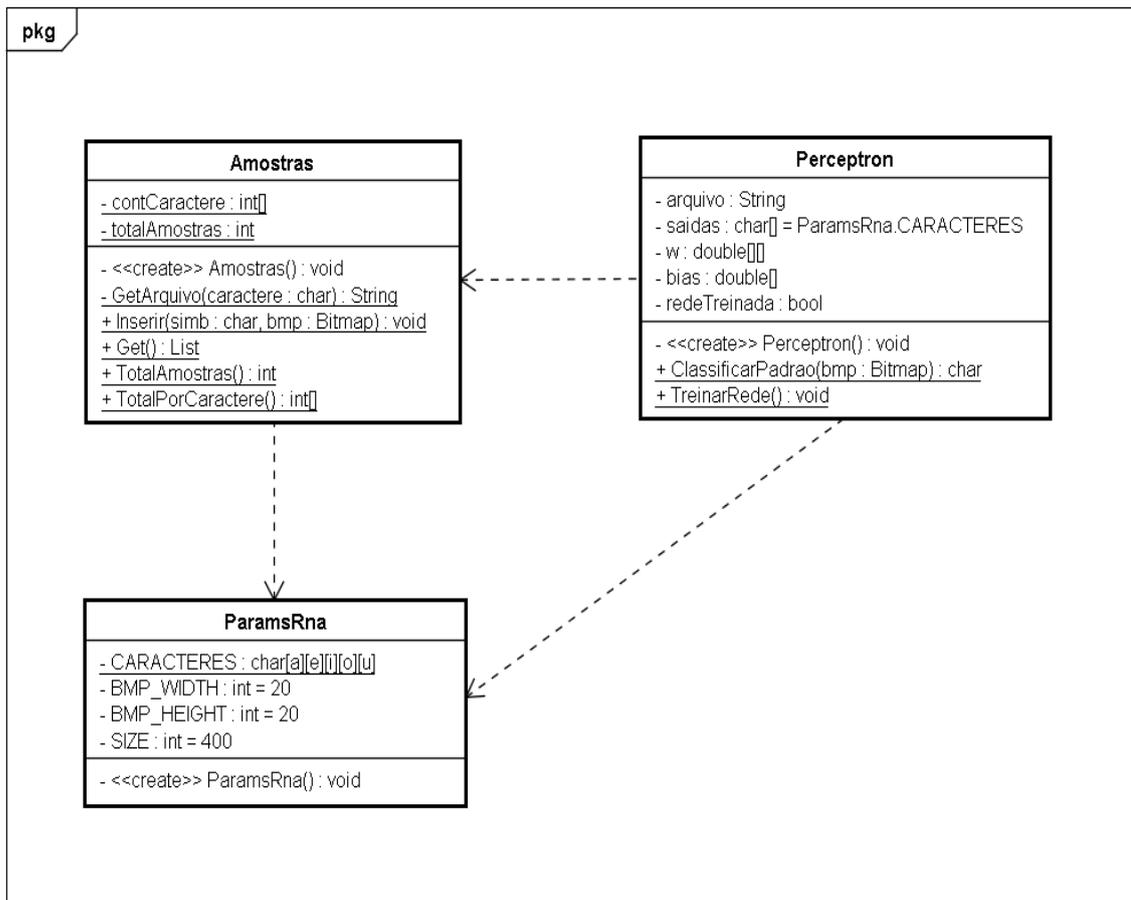


Figura 21: Diagrama de Classes do módulo RNA

A classe Amostra é a manutenção do Banco de Dados de amostras. Quando uma nova amostra é incluída, cabe a esta classe fazer a conversão para o padrão adequado e gravar este padrão no arquivo correto, de acordo com o caractere que está sendo amostrado. A ParamsRna são as constantes que definem os padrões aceitos pela rede neural artificial. Essa classe define quais os caracteres que serão reconhecidos (**a**, **e**, **i**, **o** e **u**), as dimensões do bitmap que será apresentado para o reconhecimento (20 × 20 pixels) e o tamanho do padrão a ser classificado pela rede (400 bits).

A classe Perceptron é a classe que implementa uma rede neural artificial do tipo **Perceptron** de camada única. Ela vai receber um bitmap como entrada para o processamento e devolver um caractere da tabela ASCII que corresponda àquele caractere reconhecido.

Para que a rede neural funcione, ela depende de um Banco de Dados de amostras para o seu treinamento. Esse Banco de Dados é composto de 5 arquivos, todos no formato texto: **a.dat**, **e.dat**, **i.dat**, **o.dat** e **u.dat**, contendo, respectivamente, as amostras para os caracteres a, e, i, o e u. Cada amostra nestes arquivos corresponde a um bitmap binarizado (preto e branco) convertido para um padrão de 1's e 0's, preenchendo uma linha do arquivo cada.

A seguir um exemplo para a vogal a gravada em **a.dat**:

**Bitmap binarizado com 20 × 20 pixels representando a vogal a:**



**Respectivo código binário numa linha do arquivo a.dat:**

```
111111000000011111111111000000000001111111100011111000011111110001
111111100011111100011111111000111110011111111110001111100111111111
1000111110001111111110001111100011111100011111100000000100001111
11110000000000011111111111000000001111111111111110000011111111111
111111000111111111111111100011111111111111100011111111111111111
0001111111111000000000011111111110000000001111111111111100011111
```

O código de 1's e 0's do padrão é um mapa completo de todos os pixels da imagem. Cada pixel preto foi convertido para um dígito 0, e branco num dígito 1. Cada padrão destes tem exatamente 400 dígitos, representando uma imagem de 20 × 20 pixels (20 × 20 = 400).

Juntamente com as amostras, o resultado do aprendizado da rede neural também ficará gravado no Banco de Dados. Nele estão registrados os pesos ajustados das sinapses e outros parâmetros que, quando recuperados, fazem com que a rede neural reconheça os padrões de caracteres manuscritos apresentados.

O modelo de Rede Neural Artificial implementado para a classificação dos caracteres é o **Perceptron** de camada única (*Classe Perceptron*). Essa escolha se deu principalmente pela simplicidade na implementação do modelo e por ele funcionar bem para o problema em questão.

O modo como ele está implementado para a classificação de caracteres segue as seguintes etapas:

- recuperar o aprendizado que foi gravado no Banco de Dados para que os pesos calculados com o treinamento da rede estejam acessíveis (faz isso na primeira tentativa de reconhecimento).
- transformar cada pixel preto do bitmap de entrada em um bit 1 e branco em -1. A ordem de conversão do bitmap é importante, logo, convencionou-se em mapear a imagem coluna a coluna da esquerda para a direita.

Para melhor compreensão, deve-se recorrer ao modelo esquemático de um neurônio artificial mostrado na Figura 22:

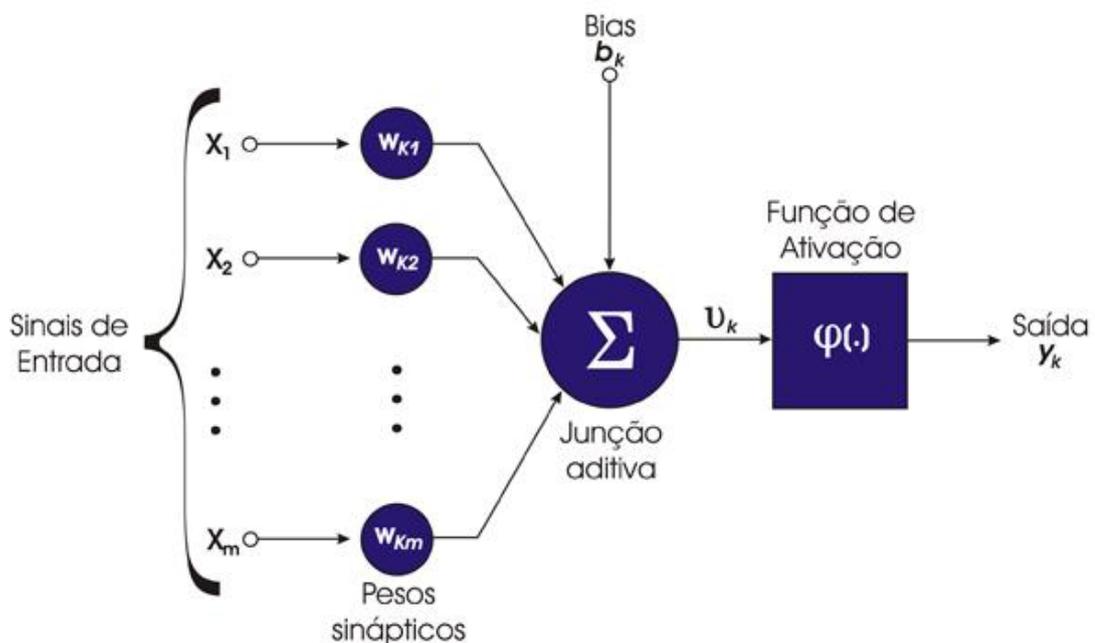


Figura 22: Modelo de um Neurônio Artificial

- ✓  $x_1, x_2, \dots, x_m$ : Sinais de entrada. Esses dados podem ser provenientes do ambiente ou de outros neurônios. Os valores de entrada diferem quanto ao intervalo permitido de acordo com o modelo implementado, mas geralmente são valores do conjunto  $[0, 1]$ ,  $[-1, 1]$ , pois são números da coloração branco e preto, ou números reais quaisquer.

- ✓  $w_{k1}, w_{k2}, \dots, w_{km}$ : Conjunto de Sinapses ou Elos de Conexão, cada caracterizada por um peso ou força própria. O peso sináptico de um neurônio pode estar em um intervalo que inclui tanto valores positivos quanto negativos.
- ✓  $\varphi(\cdot)$ : função de ativação. É usada para restringir a amplitude da saída do neurônio. Esta função calcula o estado final ou de saída de um neurônio, determinando quanto o nível de ativação de um neurônio está abaixo ou acima de um valor limiar. O objetivo do limiar é reproduzir o estado de ativo/inativo de um neurônio biológico.
- ✓  $b_k$ : bias. Tem como finalidade acrescentar ou enfraquecer a entrada líquida da função de ativação, dependendo se ele é positivo ou negativo, concomitantemente.
- ✓  $y_k$ : saída do neurônio. Valor binário no intervalo unitário fechado  $[0, 1]$ , ou alternativamente  $[-1, 1]$ .

Para o problema de classificação de vogais manuscritas é necessário mais do que um neurônio na camada de saída do Perceptron, mais precisamente, são necessários 5, um para cada vogal. Os sinais de entrada ( $x_1, x_2, \dots, x_{400}$ ) obtidos a partir do mapeamento do bitmap apresentado à rede serão propagados para esta camada num esquema denominado **feed-forward** (alimentação para frente). Nela se dará o processamento, gerando as saídas pré-definidas para a vogal reconhecida.

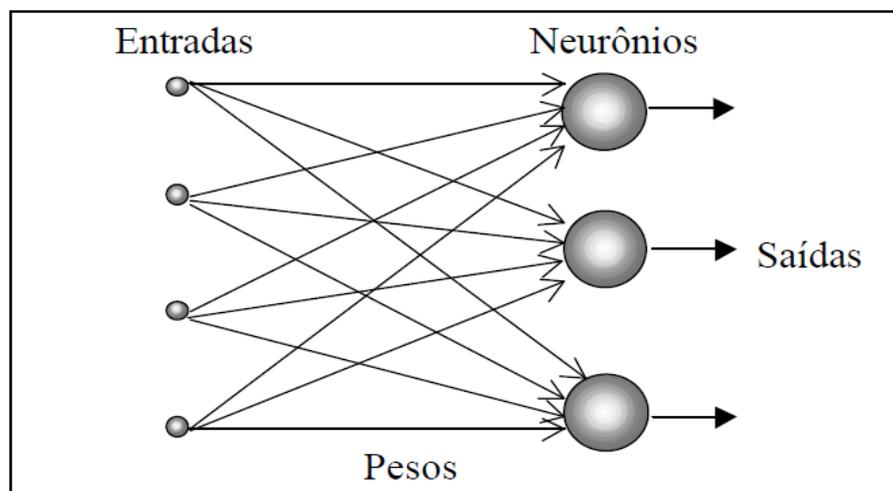


Figura 23: Modelo de um Perceptron de camada única com alimentação para frente (feed-forward)

Na Figura 24 o diagrama de um Perceptron de camada única ilustrando o nosso modelo de rede neural. Sua estrutura é idêntica à de nosso Perceptron, com a diferença de que no nosso, há 400 linhas de entrada (uma para cada pixel do bitmap apresentado) e 5 neurônios na camada de saída (um para cada vogal manuscrita reconhecida).

Em termos de implementação, o processo de classificação do bitmap de entrada pelo Perceptron se dará nos seguintes passos:

**Passo 1:** recupera o aprendizado da rede gravado no arquivo **pesos.dat** (faz isso na primeira tentativa de reconhecimento).

**Passo 2:** converte cada pixel preto do bitmap apresentado em um bit 1 e branco em um bit -1, gerando o vetor de entradas para a rede neural (veja a ordem de leitura do bitmap no diagrama da Figura 14).

**Passo 3:** calcula o valor de ativação  $[-1, 1]$  de cada saída para o padrão apresentado à rede (bitmap).

**Passo 4:** lê os sinais das saídas para verificar qual neurônio está ativado  $[1]$ , retornando para o método chamador o caractere ASCII que o representa ou “?” caso todas as saídas estão desativadas  $[-1]$ .

Para estar apta a realizar a classificação do bitmap de entrada, a rede precisa estar treinada com os padrões de amostras gravados no Banco de Dados do sistema. Cada uma dessas amostras servirá como parâmetro para que ela aprenda o que é um “a”, por exemplo, ajustando os seus *pesos* e *bias* para que ela retorne uma combinação pré-definida de sinais de saída que identifique aquela vogal quando ela for reconhecida.

As saídas esperadas (**saídas-alvo**) para cada vogal manuscrita reconhecida são mostradas na Tabela 2:

Tabela 2: Saídas alvo para as vogais manuscritas reconhecidas

Vogal	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
<b>a</b>	<b>1</b>	-1	-1	-1	-1
<b>e</b>	-1	<b>1</b>	-1	-1	-1
<b>i</b>	-1	-1	<b>1</b>	-1	-1
<b>o</b>	-1	-1	-1	<b>1</b>	-1
<b>u</b>	-1	-1	-1	-1	<b>1</b>

O objetivo do treinamento é fazer com que a rede convirja para alguma dessas combinações sempre que um padrão semelhante a algum amostrado no Banco de Dados for apresentado para a classificação pelo Perceptron. Este tipo de aprendizado é denominado **aprendizado supervisionado**. Apresenta-se o padrão e o resultado esperado para o mesmo, e ela ajusta seus parâmetros livres para chegar àquele resultado quando padrão semelhante for apresentado para classificação.

É de fundamental importância essa habilidade que uma rede neural artificial têm de “aprender” a partir de dados colhidos de seu ambiente. Percebe-se de imediato a sua importância se considerarmos o número de linhas e a complexidade do código necessário para resolver o nosso problema de classificação de vogais por métodos “tradicionais” de computação.

#### 4.5 MÓDULO HELP

E por fim o módulo Help, no qual é um browser criado para exibir a ajuda do sistema. O arquivo de ajuda é no formato, como pode ser observado na Figura 23.

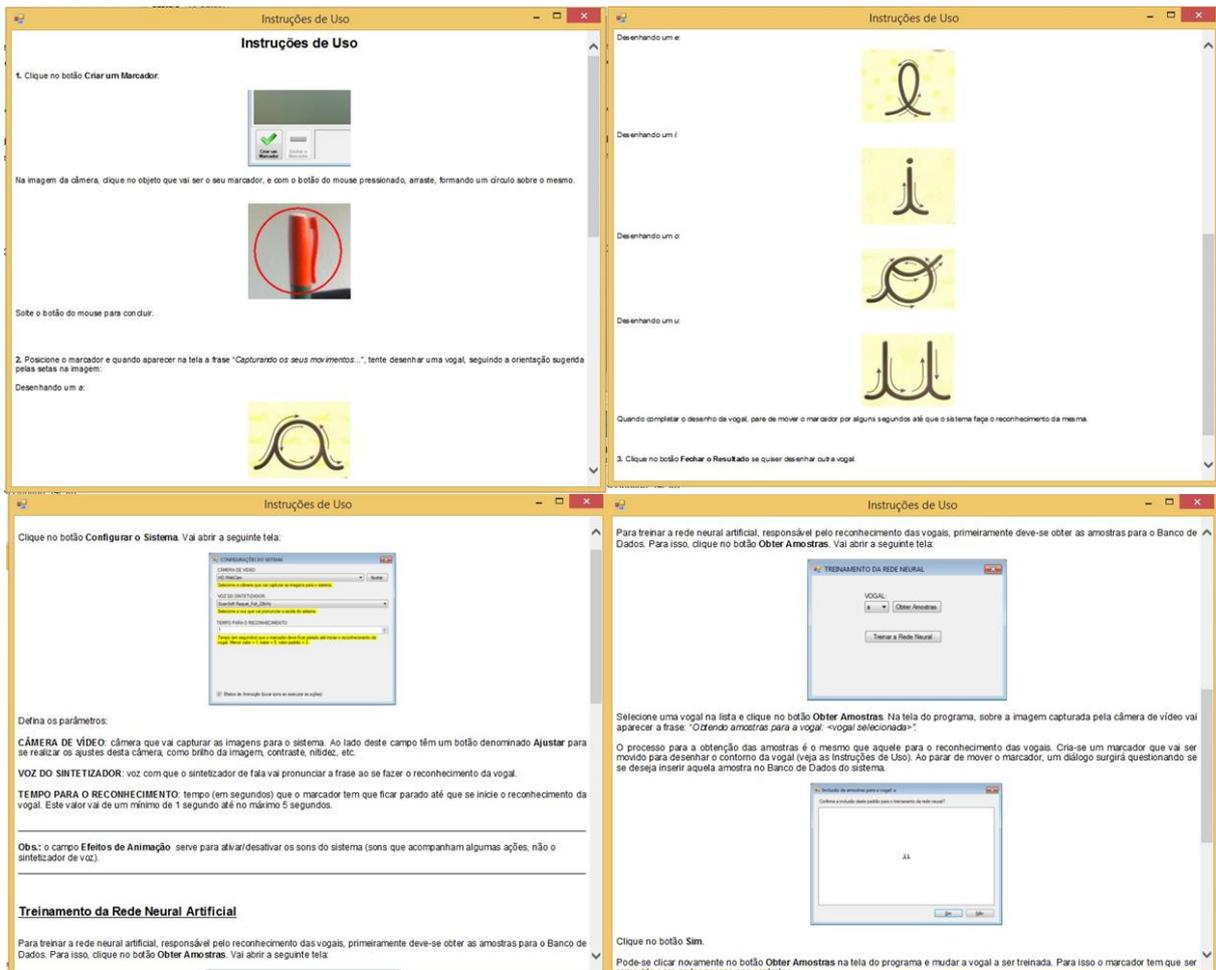


Figura 24: Tela de Ajuda

É o mais simples dentre os três módulos do projeto. Este executável possui apenas uma função: receber como parâmetros de **PRCM.exe** o caminho do arquivo de ajuda e um texto para ser exibido na barra de títulos e carrega este arquivo para a visualização num componente *WebBrowser* do **Visual Studio**.

#### 4.6 METODOLOGIA DE DESENVOLVIMENTO

Para o desenvolvimento deste sistema foi adotado o modelo de Prototipação (PRESSMAN, 2005).

O modelo de desenvolvimento baseado na prototipação procura suprir duas grandes limitações do modelo cascata. De acordo com Christian (2001) a ideia básica deste modelo é que ao invés de manter inalterados os requisitos durante o projeto e codificação, um protótipo é desenvolvido para ajudar no entendimento dos requisitos.

Este desenvolvimento passa por um projeto, codificação e teste, sendo que cada uma destas fases não é executada formalmente. Usando assim os protótipos o cliente pode entender melhor os requisitos do sistema.

Os objetivos do processo de Prototipação são listados abaixo:

- Entender os requisitos, e assim obter uma melhor definição dos requisitos do sistema;
- Possibilita que o desenvolvedor crie um protótipo que deve ser construído;
- É melhorado até atingir o objetivo final;

A figura 25, apresenta as fases do processo, comunicação, plano rápido, modelagem, construção, implantação, entrega e *feedback*.



Figura 25: Fases do processo (PRESSMAN, 2005)

Todavia, Pressman (2005) afirma que se um protótipo de trabalho for construído, o desenvolvedor tentará usar fragmentos de programas existentes ou aplicará ferramentas (por exemplo, geradores de relatórios, gerenciadores de janelas etc.) que possibilitam que programas de trabalho sejam gerados rapidamente.

## 4.7 MÓDULOS DO SISTEMA

A Figura 25 apresenta os módulos utilizados no sistema. O módulo principal é a aplicação do sistema, e dentro deste estão Inteligência Artificial, encapsulando a Rede Neural com respectivo treinamento, e Configuração. O módulo de captura, onde se localiza a webcam e as técnicas de processamento de imagem, ligado ao módulo de captura está o módulo de projeção, e dentro deste a mesa adaptada. O módulo de desenho está diretamente ligada aos módulos de animação e som. E também os módulos isolados, não interligados a nenhum outro, o Help, que apresenta as instruções de uso do sistema, e a Configuração, que possibilita ao usuário configurar todo ambiente de trabalho como ajustes de câmera (luz, brilho, saturação, nitidez entre outros), sintetizar a voz, e tempo para reconhecimento.

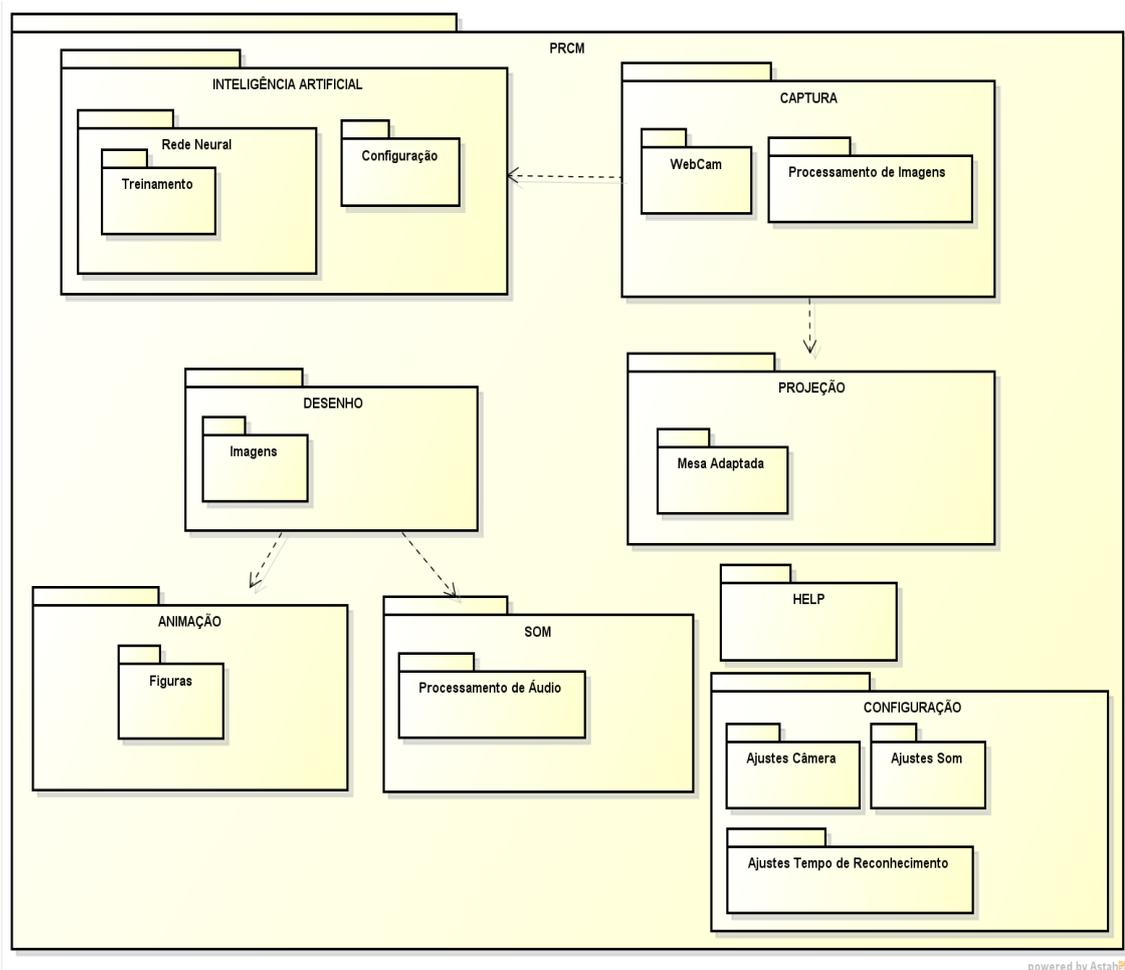


Figura 26: Módulos do Sistema

## 4.8 ARQUITETURA DO SISTEMA

A Figura 26 apresenta a arquitetura do sistema, no qual é responsável por interoperar os dados de informação com os dados de captura de movimento.

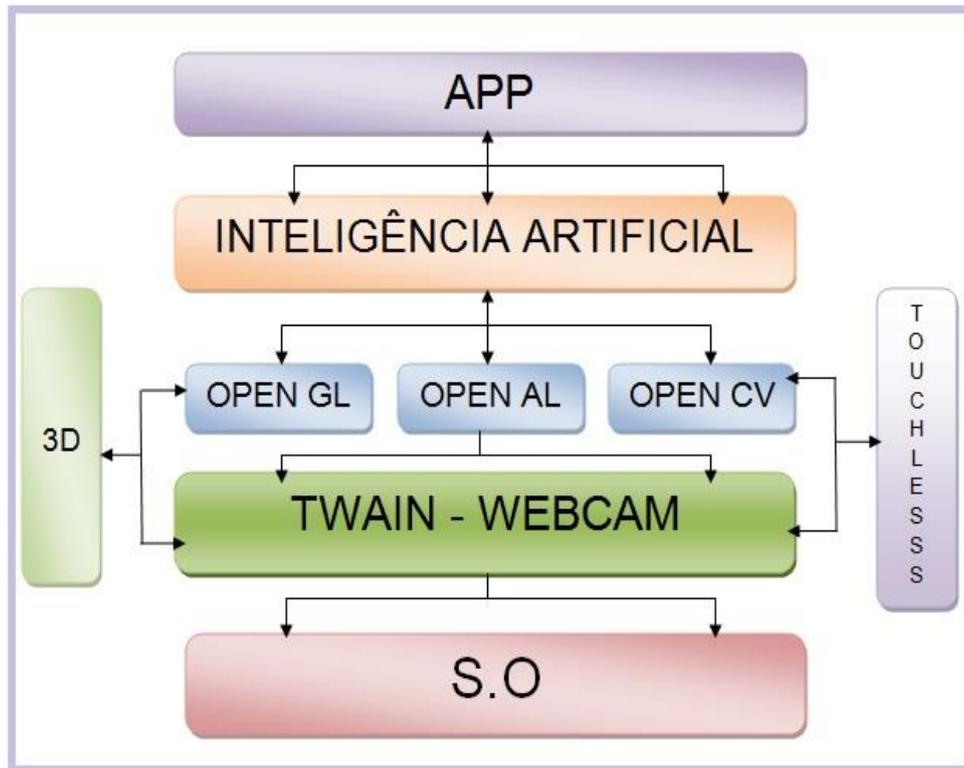


Figura 27: Arquitetura do Sistema

Como pode observar a arquitetura do sistema é *Stand Alone*, com camadas de bibliotecas.

O S.O – sistema operacional – utilizado é o Microsoft Windows, nele é possível acessar várias bibliotecas de configuração de câmera, sintetizador de voz.

O TWAIN (do inglês *Technology Without na Interesting Name*) – Webcam é um padrão de captura a partir de scanners: uma imagem capturada através de uma API para os sistemas operacionais Microsoft Windows ou Apple Macintosh (WIKIPEDIA,

O OpenGL (do inglês *Open graphics Libray*), é uma API livre utilizada na computação gráfica, para desenvolvimento de aplicativos gráficos, ambientes 3D. O

OpenGL foi produzido em linguagem de programação C e C++, podendo ser utilizada para diversas outras com um alto nível de eficiência (WIKIPEDIA, 2015).

O OpenAL (do inglês Open Audio Library), é uma API também livre e de multiplataforma desenvolvida para lidar com áudio multicanal tridimensional. O OpenAL é geralmente utilizado com o OpenGL (WIKIPEDIA, 2015).

A biblioteca OpenCV (do inglês *Open Source Computer Vision Library*), originalmente desenvolvida pela Intel em 2000. É livre para uso acadêmico e comercial para desenvolvimento de aplicativos na área de visão computacional. Esta biblioteca possui módulos de processamento de imagens e vídeo i/o, estrutura de dados, álgebra linear, Interface Gráfica do Usuário – GUI (WIKIPEDIA, 2015).

Segundo (SILVA, 2009) a inteligência artificial é um ramo de pesquisa da ciência da computação que busca, através de símbolos computacionais, construir mecanismos e/ou dispositivos que simulem a capacidade do ser humano de pensar, resolver problemas, ou seja, de ser inteligente. O progresso na principal área dessa pesquisa, que é a de fazer uma inteligência similar à do ser humano, é lento.

Todas as bibliotecas do sistema tem a responsabilidade de interoperar os dados entre si. Fazendo com que haja comunicação entre os dados da captura de movimento com o resultado final de reconhecimento do caractere realizado pela rede neural.

#### **4.9 FLUXOGRAMA DO SISTEMA**

A Figura 28 abaixo apresenta de maneira geral do processo de captura de movimento até o momento de reconhecimento da letra.

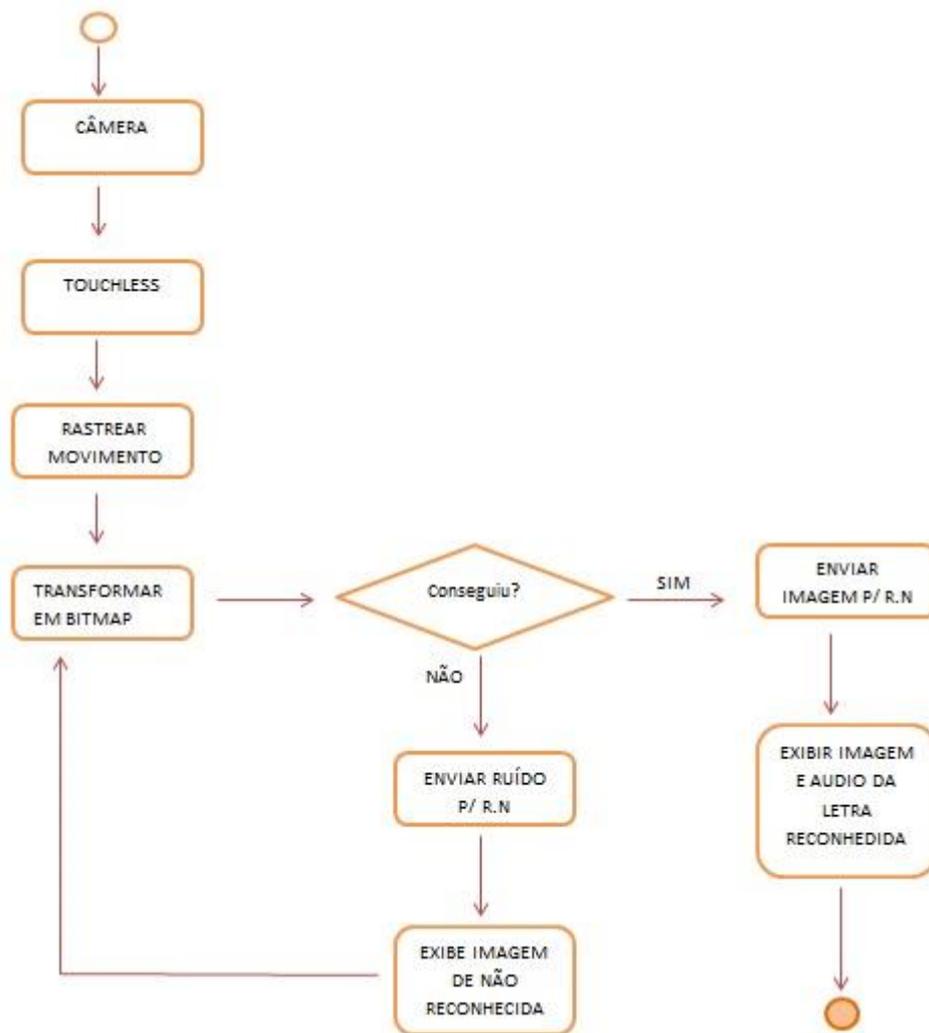


Figura 28: Fluxograma do Sistema

Inicialmente a câmera é ativada e conseqüentemente o Touchless que irá avaliar onde o marcador estará se movendo, gerando um evento que envia a coordenada do marcador para que seja possível realizar o rastreamento do movimento. Essa coordenada é convertida em um objeto, que transforma esse movimento em um bitmap. A partir do momento que esse bitmap é gerada é realizada a verificação se foi gerada uma imagem ou um ruído, caso seja gerado uma imagem, esta é enviada para rede neural que verifica suas amostras e exibe uma resposta, no caso exibe uma imagem e emite áudio (pois o mesmo deve ser associado a letra escrita pelo usuário, para que seja feita a correlação de som e imagem) da letra reconhecida, caso seja gerado não seja gerado uma imagem, é

enviado um ruído para rede neural, que não reconhece e volta a etapa de transformação do movimento em bitmap.

## 5. RESULTADOS E DISCUSSÕES

Considerando que o uso do software em ambientes de ensino escolar e clínicas médicas ainda não foram realizados, os testes foram realizados com dez usuários desconsiderando gênero, idade, classe social e considerando apenas se o mesmo é ou não alfabetizado, entretanto a faixa etária dos mesmos foi de 8 à 10 anos, foi utilizado um computador com sistema operacional Windows 7, o ambiente de teste foi no escritório da casa da autora deste projeto situada na cidade de Sertaneja, Paraná.

Inicialmente houve dificuldades com o manuseio do computador, alguns dos usuários não tinham ainda nenhum tipo de contato com o computador, sendo assim teve que utilizar a ferramenta com a ajuda de um adulto. Com base nesses usuários, foram selecionados apenas dois destes, pois estes pertencem ao grupo de portadores de distúrbios da linguagem e ambos os usuários já estão alfabetizados.

Para isso foram realizados os seguintes testes: a quantidade de acertos, o tempo de processamento para reconhecimento da letra, e quais letras foram reconhecidas de maneira mais eficiente. Levando em consideração que o treinamento da rede neural foi realizado pela desenvolvedora da ferramenta, o que talvez dificulte na eficácia dos resultados.

A ferramenta foi apresentada aos usuários de maneira simples, clara e objetiva, sendo colocada uma fita adesiva de coloração vermelha na ponta do dedo indicador de cada um, informado que este seria o “lápiz” para escrever a letra, posteriormente à desenvolvedora fez uma apresentação de como funciona a ferramenta e partir disto cada um dos usuários realizou o teste. Como dito acima alguns tiveram dificuldades com o manuseio do computador e então demoram um pouco mais e precisaram de ajuda para conseguir, outros tiveram dúvidas de como usar a ferramenta propriamente dita, e a desenvolvedora ajudou explicando detalhadamente cada passo necessário.

Os resultados dos testes podem ser analisados nos gráficos a seguir.

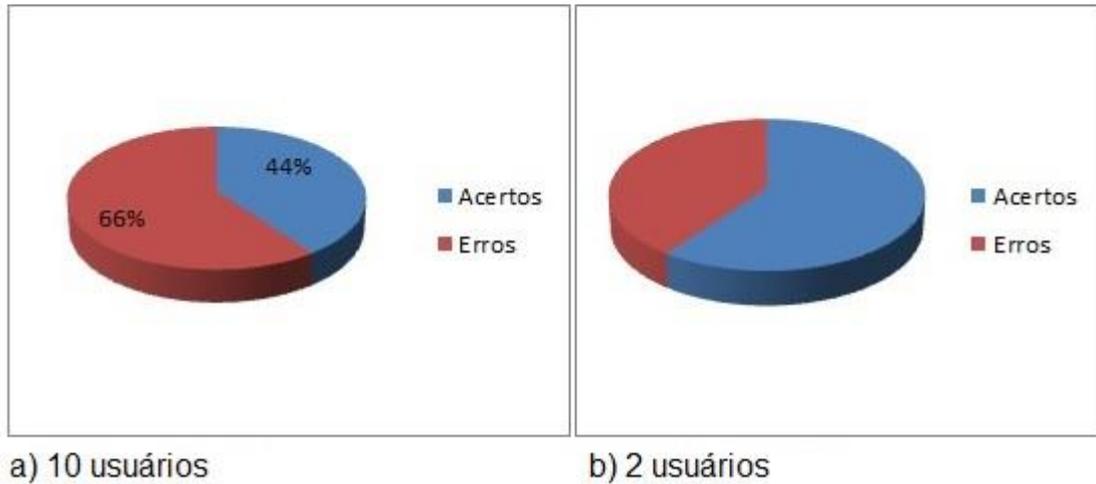


Figura 29: Taxa de Acerto do Algoritmo

O Figura 29 a exibe a porcentagem de acertos e erros do software baseado nos dez usuários, e o Figura 29 b baseado nos dois usuários com distúrbios da linguagem.

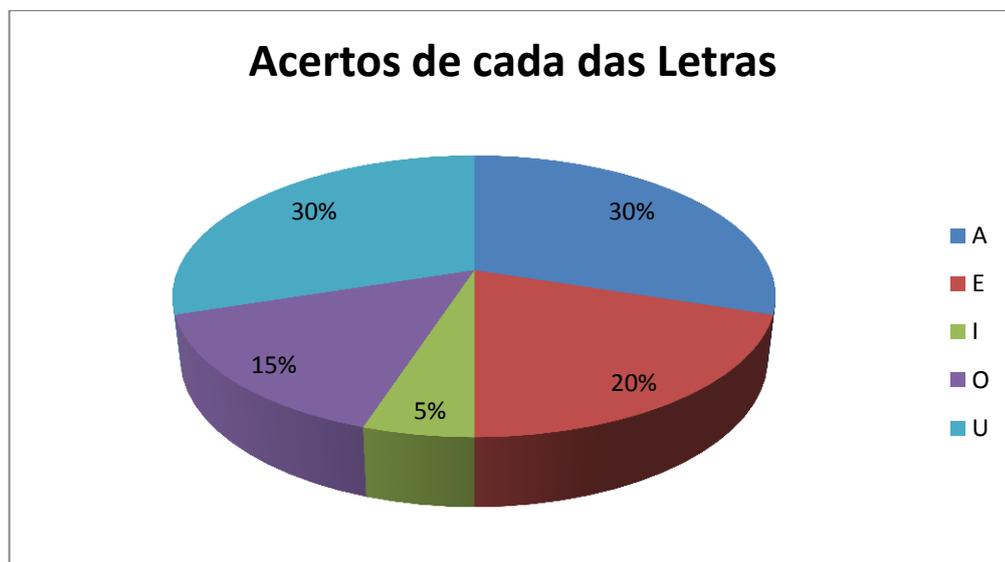


Figura 30: Taxa de acertos por letra do algoritmo

O Figura 30 mostra a porcentagem de acertos de cada letra, levando em consideração que cada um dos dez usuários testaram as cinco vogais (a, e, i, o, u). No início da apresentação do sistema, foi especificado que cada usuário deveria escrever as cinco vogais respectivamente, e como todos eram alfabetizados não foi

necessário apresentar cada letra separada, apenas um quadro com as cinco vogais juntas, como observado na Figura 31.

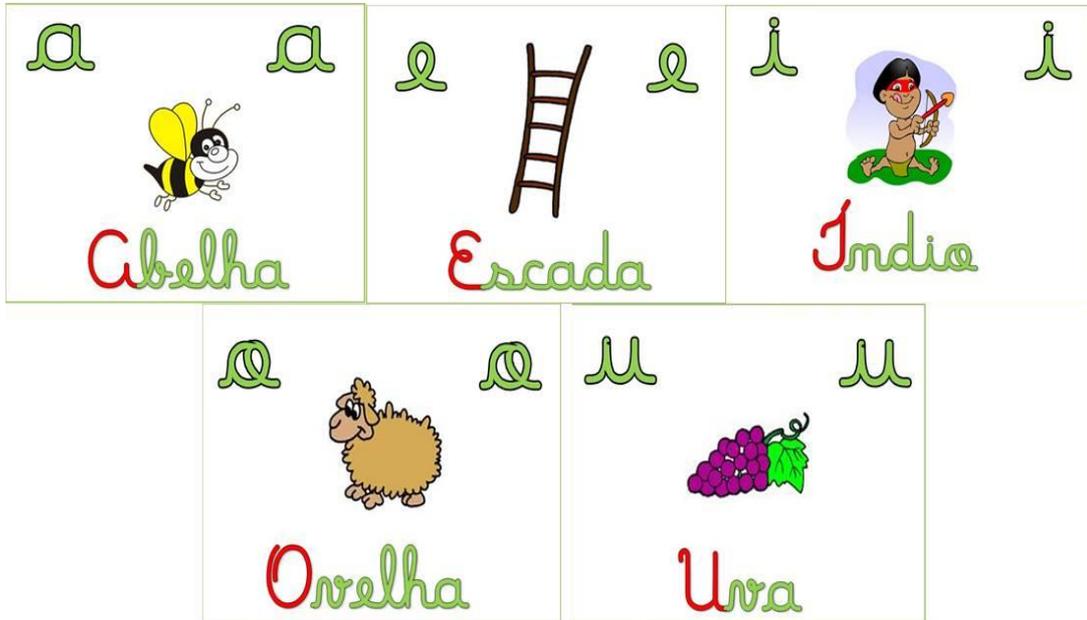


Figura 31: Apresentação das vogais aos usuários



Gráfico 1: Tempo de Execução

E o Gráfico 1, exibe o tempo de processamento para o reconhecimento da letra, pode-se observa-se que o tempo varia de 2 milissegundos a 3 milissegundos.

Com esses resultados observa-se, que ainda não possuindo uma validação do software, o mesmo já mostra resultados de processamento eficientes, devido ao reconhecimento ou não reconhecimento da letra após a escrita dos usuários ser de aproximadamente 3 milissegundos.

Os resultados de reconhecimento das letras ainda são pequenos, devido a fatores como treinamento da rede neural, que se deu pela própria desenvolvedora treinando apenas 20 amostras de cada vogal, o que é consideravelmente baixo com uma rede neural de camada única, ao pouco tempo de testes.

A rede Neural é uma ferramenta muito poderosa quando se trata de reconhecimento de padrões e exatamente por isso deve-se tomar um extremo cuidado, pois ela pode facilmente aprender a reconhecer apenas os dados alimentados na fase de treinamento. Este é um dos principais problemas na configuração da rede uma vez, que durante a fase de treinamento, podem-se observar altas porcentagens de acerto enquanto na fase de execução o índice pode cair consideravelmente se a rede não tiver generalizado corretamente, ou seja, aprendido apenas as características inerentes à letra desenhada associada aos pesos.

## 6. CRONOGRAMA

Abaixo é apresentado o cronograma de desenvolvimento da ferramenta, com a finalidade de mostrar o acompanhamento das atividades realizadas durante o todo processo de desenvolvimento.

Tabela 3: Cronograma

Atividades / Meses	2014				2015					
	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun
Etapa 1										
Etapa 2										
Etapa 3										
Etapa 4										
Etapa 5										
Etapa 6										
Etapa 7										
Etapa 8										

Etapa 1 – Revisão Bibliográfica

Etapa 2 – Instalação das Ferramentas

Etapa 3 – Desenvolvimento do Sistema

- Implementar Rotinas para Captura dos Movimentos;
  - Capturar imagem;
  - Processamento de Imagem;
  - Transformação da Imagem
- Organizar o Código Fonte;
- Implementar a Rede Neural;
- Treinar Rede Neural;

Etapa 4 – Testes do Sistema

Etapa 5 – Documentação Final

Etapa 6 – Artigos

Etapa 7 – Apresentação do Trabalho de Conclusão de Curso

Etapa 8 – Reuniões com o Professor Orientador

## 7. CONCLUSÃO

O objetivo deste trabalho foi desenvolver uma ferramenta de captura de movimento por coloração que fosse capaz de reconhecer esse movimento e exibi-lo na tela.

Apesar de considerar que a solução proposta possa ser eficiente e de baixo custo, sua utilização nas escolas requer uma avaliação de outros fatores, que ultrapassam o escopo deste trabalho; saliento que muitas particularidades ainda não foram avaliadas e serão posteriormente, sendo elas a melhora ou mudança do algoritmo da rede neural, a incorporação de novas técnicas de processamento de imagem e reconhecimento de padrões.

Acredito que apesar de se apoiar em diversas metodologias para auxiliar no ensino e facilitar a aprendizagem dos alunos, portadores de distúrbios da linguagem, ainda estou longe de encontrar uma solução final e completa. Existem muitos trabalhos que discorrem sobre sistemas de apoio, no entanto, as tecnologias de interação natural ainda se expandem para esta aplicação. Não posso restringir os estudos ao uso nessa ferramenta, contudo, permitir o avanço demarcado deste trabalho. Os estudos acerca em trabalhos que farei a combinação de tecnologias de captura de movimento mais acessíveis e mais eficientes, redução no tempo de desenvolvimento de produtos aplicados ao ambiente escolar e os distúrbios de aprendizagem. A ferramenta apresentada trabalha com a criança da mesma maneira que a professora em sala de aula pode ser utilizada como exercício adicional, uma vez que reforça o som, a partir da associação de uma personagem e ao desenho correspondente da sua letra.

Os resultados não foram satisfatórios em todos os casos das letras, o que evidencia que o treinamento da rede neural deve ser generalizado, o que significa que a quantidade de amostras deve ser muito superior a quantidade que foi utilizada para os testes (no caso de 20 amostras).

Porém como resultado tecnológico garantiu que apoiando na metodologia de ensino multissensorial que antes era apenas com uma interação convencional no computador e agora está com uma interação natural, ou seja, o usuário da ferramenta passa de um expectador para ator o processo de aprendizagem.

## 8. TRABALHOS FUTUROS

Devido aos resultados encontrados nos testes do sistema, será necessário um estudo aprofundado em técnicas distintas de reconhecimento de movimento.

O modelo de Rede Neural também será alterado, passando de uma camada simples para múltiplas camadas.

O reconhecimento das letras deixaria de ser apenas das vogais, e sim para todo o alfabeto e posteriormente a junção silábica, e os numerais.

Acredito que com a incorporação de novas técnicas e novos testes deixará a ferramenta mais robusta e completa para auxiliar no tratamento de distúrbios da linguagem.

Este trabalho e esta ferramenta apresentados não são o fim de uma jornada acadêmica, mas o início de uma longa caminhada, desejo alcançar os objetivos aqui descritos e proporcionar um grande diferencial para este tipo de tratamento.

## 9. REFERÊNCIAS

- ASSOCIAÇÃO NACIONAL DE DISLEXIA. Dislexia. Disponível em: <<http://www.andislexia.org.br/>> . Acesso em 03.out.2014.
- BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. 2ª ed. Rio de Janeiro: Elsevier, 2007.
- BUCKLAND, Mat. **Redes Neurais em Português Simples**. 2010. 22 f. Tradução. Disponível em: <<http://www.ai-junkie.com/ann/evolved/nnt1.html>>. Acesso em: 12 jan. 2015.
- BURKE, J. W.; MORROW, P.J.; MCNEILL, M. D J; MCDONOUGH, S.M.; CHARLES, D.K. **“Vision Based Games for Upper-Limb Stroke Rehabilitation”**. Proceedings of International Conference on Machine Vision and Image Processing, 2008. IMVIP '08, 2008.
- CAMEIRÃO, M.S.; BERMÚDEZ, S.; OLLER, E.D.; VERSCHURE1, P.F.M.J. **“Neurorehabilitation using the virtual reality based Rehabilitation Gaming System: methodology, design, psychometrics, usability and validation”** Journal of Neuroengineering .
- CAMPAGNA, J.P.P; BREGA, J.R. F. **Utilização da realidade aumentada no suporte à correção de movimentos em exercícios físicos que envolvem joelho**. 2009, Santos – SP. Sociedade Brasileira de Computação – SBC, 2009.
- CAPOVILLA, A. G. S. **Dislexia do desenvolvimento: definição, intervenção e prevenção**. Revista da Associação Brasileira da Psicopedagogia. Disponível em: <<http://WWW.http://www.abpp.com.br/artigos/59.htm>>. Acesso em: 30/10/2014.
- CHRISTIAN Reis . , **Caracterização de um Modelo de Processo para Projetos de Software Livre** .teste de mestrado.Instituto de Ciências Matemática e Computação. São Carlos, São Paulo Abril de 2001. Acesso em: 30/10/2014.

COELHO, Diana Teresa. **Dislexia, Disgrafia, Disortografia e Discalculia**. 2003. 48f. Tese (Mestrado) – Curso de Pedagogia, Universidade Estadual de Maringá, Maringá, 2003.

DAMASCENO, Eduardo Filgueiras. **Sistema de Reabilitação Baseado em Técnicas de Captura de Movimento para Tratamento da Lombalgia Mecânica**. 2013. 131 f. Tese (Doutorado) - Curso de Engenharia Elétrica, Departamento de Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, Uberlândia, 2013.

DUARTE, J.M.M. et al. (2013). A multimídia na dislexia: tecnologia na reeducação da dislexia.

ELLWANGER, C., Santos, C. P., and MOREIRA, G. J. (2014). **Padrões de interface, gameficação e mobilidade no desenvolvimento de aplicações educacionais**. In Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems, pages 365–368. Sociedade Brasileira de Computação.

FERNANDES, R. A; PENNA, J. S. **Contribuições da psicopedagogia na alfabetização dos disléxicos**. Revista Terceiro Setor, v. 2, n.1, 2008.

FONTES, Maria Alice. O que são os Transtornos de Linguagem? Causas e Tipos. **Plenamente**, São Paulo, v. 7, n. 8, p.05-05, 28 fev. 2014. Disponível em: <O que são os Transtornos de Linguagem? Causas e Tipos.>. Acesso em: 06 jan. 2015.

GOMES, A., Tomás, et.al (2004). **Integração curricular das tic: Ponto de vista dos professores**. Osório, A., e Puga, M. (Coords.) (eds.) As TIC na Escola. Universidade do Minho, Braga, pages 33–47.

GOTTFRIED, J.-M., Fehr, J., and Garbe, C. S. (2011). Computing range flow from multi-modal kinect data. In Advances in Visual Computing, pages 758–767. Springer.

GUEDES, Gilleanes T. A.. **UML 2 uma abordagem prática**. São Paulo: Novatec, 2009. 300 p.

LIMA, D. B de M. **“Análise da eficácia de um programa de intervenção na modificação da postura de protusão do ombro por meio de exercícios para a alteração de propriedades musculares.”** Dissertação de Mestrado. Universidade Federal de Minas Gerais, 2007.

LUCZYNSKI, Zeneida Bittencourt. **Dislexia: Você sabe o que é?** Curitiba: Autora, 2001. 265 p.

Medeiros, T. G. d. and Oliveira, E. R. C. (2008). A influencia da consciencia fonologica em crianças alfabetizadas pelos metodos fonico e silabico. Rev. CEFAC, 10(1):45–50.

MENEZES, Clariane Silva et al. Redes Neurais Classe Modular Aplicadas no Reconhecimento de Caracteres Manuscritos. **Colloquim Exactarum**, Presidente Prudente, v. 2, n. 6, p.170-183, 2 jul. 2014.

MOUSINHO, R. **Conhecendo a dislexia.** Revista Sinpro, Rio de Janeiro, p.26-33, abr. 2004.

MURPHY, Cristina Ferraz Borges. **Software brasileiro melhora percepção auditiva em crianças com dislexia.** 2008. Disponível em: <[http://www2.uol.com.br/vivermente/noticias/software\\_brasileiro\\_melhora\\_percepcao\\_auditiva\\_em\\_crianças\\_com\\_dislexia.html](http://www2.uol.com.br/vivermente/noticias/software_brasileiro_melhora_percepcao_auditiva_em_crianças_com_dislexia.html)>. Acesso em: 24 out. 2014.

NASCENTE, Sheila de Paula. **Distúrbios da Linguagem.** Uruguaiana: Video, 2010. 16 slides, color. Disponível em: <<http://pt.slideshare.net/Sheilinha1/apresentao-distrbios-da-linguagem>>. Acesso em: 06 jan. 2015.

NOGUEIRA, Emília Alves et al. **Uso da Realidade Aumentada e Reconhecimento de Voz como Ferramenta de Apoio ao Aprendizado para Pessoas com Dislexia.** Jataí: Departamento de Ciência da Computação – Universidade Federal de Goiás.

OLIVEIRA, Ércio. Dislexia. 2014. Disponível em: <<http://www.abcdasaude.com.br/pediatria/dislexia>>. Acesso em: 26.set.2014.

PRADO, Elisa; ALIOTO, Olavo Egídio. **Estratégias na alfabetização de crianças disléxicas**. 2002. 10 f. Tese (Doutorado) - Curso de Licenciatura em Pedagogia, Faculdade Método de São Paulo, São Paulo, 2002.

PRESSMAN, Roger. **Engenharia de Software: Uma Abordagem Profissional**. 4. ed. Rio de Janeiro: Mcgraw-hill, 2005. 780 p

REGTVOORT, A.G.F.M; VAN Der Leij A. **Early intervencion with children of dyslexic parentes: Effects of computer based reaing instruction at home on literacy acquisition.Learning and Individual Differences**. 2007. 17p.

RICHARDS, J. G. “**The measurement of human motion: A comparison of commercially avaailabre systems**”, Human Movement Science, v 18, issue 5, 1999, P 589 -602, 1999.

RIBEIRO, Eduardo Ferreira. **O uso do aprendizado neural para o reconhecimento de padrões**. 2004. 7 f. Tese (Graduação) - Curso de Ciência da Computação, Faculdade de Ciência da Computação, Uberlândia, 2004.

SEBRA, A. G. and DIAS, N. M. (2011). **Métodos de alfabetização: delimitação de procedi- mentos e considerações para uma prática eficaz**. Revista Psicopedagogia, 28(87):306–320.

SILVA, Marco Aurelio. **I.A. INTELIGÊNCIA ARTIFICIAL**. São Paulo: Brasil Escola, v. 1, n. 1, 19 dez. 2013. Disponível em: <<http://www.brasilecola.com/informatica/inteligencia-artificial.htm>>. Acesso em: 05 mar. 2015.

SOARES, M. (2003). **A reinvenção da alfabetização. Presença pedagógica**, 9(52):15–21. Stinchfield, S. M. (2013). *Speech disorders: A psychological study of the various defects of speech*, volume 20. Routledge.

Sue Bergin, “**Communication Disorders: Breaking through the Barriers**”, **Ensign**, fevereiro de 1991, pp. 46–50.

SHAYWITZ, S. and FIGUEIRA, V. (2006). **Entendendo a dislexia: um novo e completo programa para todos os níveis de problemas de leitura**. Artmed, 1st edition.

TELES, P. **Dislexia: como identificar? Como intervir?** Revista Portuguesa de Clínica Geral. Disponível em: <<http://www.ebah.com.br/content/ABAAABkVMAE/alfabetizar-criancas-dislexicas-nas-fases-iniciais?part=2>>

WIKIPEDIA. OPENAL. Disponível em: <<https://pt.wikipedia.org/wiki/OpenAL> >. Acesso em: 03 maio. 2010.

WIKIPEDIA. OPENCV. Disponível em: <<https://pt.wikipedia.org/wiki/OpenCV> >. Acesso em: 03 maio. 2010.

WIKIPEDIA. OPENGL. Disponível em: <<https://pt.wikipedia.org/wiki/OpenGL> >. Acesso em: 03 maio. 2010

WIKIPEDIA. TWAIN. Disponível em: <<https://pt.wikipedia.org/wiki/Twain> >. Acesso em: 03 maio. 2010

WONG, A. **Low-Cost visual / Inertial Hybrid Motion Capture System for Wireless 3D Controllers** by. 2007. University of Waterloo, 2007.

Zavaleta Jorge et al. **DysDTool: Uma ferramenta inteligente para a avaliação e intervenção no apoio ao diagnóstico da dislexia.** 2006. 10 f. Tese – Departamento de Otorrinolaringologia/Oftalmologia, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.