

UNIVERSIDADE ESTADUAL DO PIAUÍ – UESPI
CAMPUS PROFESSOR ALEXANDRE ALVES DE OLIVEIRA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

ÁDRIAN CHAVES OLIVEIRA

APLICAÇÃO INTELIGENTE DE RECONHECIMENTO DE PLANTAS MEDICINAIS
UTILIZANDO REDES NEURAS ARTIFICIAIS

Biblioteca UESPI PHS
Registro Nº _____
CDD 001.535
OUTER 0480
V _____ EX 1
Data 28 10 2018
Voto [assinatura]

PARNAÍBA

2015

ÁDRIAN CHAVES OLIVEIRA

**APLICAÇÃO INTELIGENTE DE RECONHECIMENTO DE PLANTAS MEDICINAIS
UTILIZANDO UMA TÉCNICA DE INTELIGÊNCIA COMPUTACIONAL**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual do Piauí, como parte das exigências da disciplina de Estágio Supervisionado, requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Msc. Dario Brito Calçada.

PARNAÍBA

2015

O48a

Oliveira, Ádrian Chaves

Aplicação inteligente de reconhecimento de plantas medicinais utilizando
Redes Neurais Artificiais /

Ádrian Chaves Oliveira. – Parnaíba: UESPI, 2015.

60 f. : il.

Orientador: MSc. Dario Brito Calçada

Monografia (Graduação em Ciência da Computação) - Universidade
Estadual do Piauí, 2015.

1. Reconhecimento de Padrões 2. Redes Neurais Artificiais 3. Multi Layer
Perceptron 4. Processamento Digital de Imagem 5. Aplicação Inteligente I.
Calçada, Dario Brito II. Universidade Estadual do Piauí III. Título

CDD 001.535

ÁDRIAN CHAVES OLIVEIRA

**APLICAÇÃO INTELIGENTE DE RECONHECIMENTO DE PLANTAS MEDICINAIS
UTILIZANDO UMA TÉCNICA DE INTELIGÊNCIA COMPUTACIONAL**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual do Piauí, como parte das exigências da disciplina de Estágio Supervisionado, requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Msc. Dario Brito Calçada.

Monografia Aprovada em: **24 de julho de 2015.**

Banca Examinadora:



Prof. M.Sc. Dario Brito Calçada
UESPI/Parnaíba – Orientador



Prof. Dr. Sérgio Barros de Sousa
UESPI/Parnaíba – Avaliador



Prof. M.Sc. Rodrigo Augusto Rocha Souza Baluz
UESPI/Parnaíba – Avaliador



DEDICATÓRIA

Dedico aos meus pais, a minha namorada Aline Márcia e aos que me incentivaram a superar os desafios a mim impostos. Dedico também a Deus, pela força, coragem e incentivo espiritual.

AGRADECIMENTOS

Aos meus pais, pelo incentivo, educação e apoio durante todo o curso.

Ao professor Dário Brito, pela orientação e incentivo para conclusão deste trabalho e pelas oportunidades a mim oferecidas.

A todos aqueles que contribuíram de alguma forma para que este trabalho viesse a ser concluído.

A todos aqueles que acreditaram e me deram forças para terminar este trabalho.

E, principalmente, a Deus, que foi meu companheiro em todos os momentos mais difíceis deste curso.

|

|

|

|

“Que os vossos esforços desafiem às impossibilidades, lembrai-vos de que as grandes coisas do homem foram conquistadas do que parecia impossível.”

(Charles Chaplin)

RESUMO

O uso de plantas medicinais como uma forma de prevenir e tratar doenças têm sido amplamente difundido, visto que, as mesmas apresentam propriedades farmacológicas que são de extrema importância para o bom funcionamento dos organismos dos seres vivos. Todavia, as plantas podem possuir várias espécies diferentes, bem como em cada uma delas apresentar propriedades e finalidades distintas, sendo essas, propriedades tanto benéficas ao organismo de um ser vivo, quanto prejudiciais, como é o caso das espécies *Plectranthus amboinicus* (boldo) e *Plectranthus barbatus* (malva). O processo de reconhecimento e classificação dessas plantas demandam análises e métodos que garantam que a espécie seja utilizada para tal finalidade. Entretanto, muitos destes métodos utilizam características complexas e demandam um grande conhecimento técnico para fazer esta classificação. A aplicação de técnicas computacionais para automação das atividades de reconhecimento de padrões motivou o desenvolvimento deste trabalho. Com base na literatura, a folha é a característica e a parte mais simples da planta a ser analisada, de forma a reconhecê-la. As Redes Neurais Artificiais (RNAs) em conjunto com Processamento Digital de Imagens (PDI) proporcionam uma alta capacidade em resolução de problemas, sendo possível encontrar diversas aplicações que utilizam estas técnicas em conjunto. Neste trabalho é apresentado o uso de técnicas de Processamento Digital de Imagem para extrair características das folhas dessas duas espécies da família *Plectranthus*. Neste sentido, foram coletadas amostras destas espécies e extraídas características anatômicas e morfológicas que as compõe, onde estas características foram utilizadas para compor uma base de dados para ser utilizada como parâmetros de entrada para uma RNA do tipo *Multi Layer Perceptron* aqui desenvolvida, a qual foi capaz de classificar corretamente as duas espécies com uma taxa de acerto de 96,15%. Os resultados obtidos com o treinamento da RNA possibilitou o desenvolvimento de um aplicativo inteligente para dispositivos móveis para o reconhecimento de plantas medicinais a partir da análise de características presente em suas folhas. Como resultado, obteve-se uma aplicação capaz de reconhecer plantas das espécies estudadas de modo bastante satisfatório demonstrando que é possível a utilização de RNAs para classificar duas espécies distintas e com isso realizar a produção de sistemas inteligentes.

PALAVRAS-CHAVE: Reconhecimento de Padrões. Redes Neurais Artificiais. *Multi Layer Perceptron*. Processamento Digital de Imagem. Aplicação Inteligente.

ABSTRACT

The use of medicinal plants as a way to prevent and treat diseases have been widespread, since they present pharmacological properties that are of utmost importance to the proper functioning of the bodies of living things. However, plants can have several different species and in each of them present properties and different purposes, and these, properties both beneficial to the body of a living being, as harmful, such as the species *Plectranthus amboinicus* (boldo) and *Plectranthus barbatus* (mauve). The process of recognition and classification of these plants require analysis and methods to ensure that the species is used for this purpose. However, many of these methods use complex characteristics and require a great technical knowledge to make this classification. The application of computational techniques for automation of pattern recognition activities motivated the development of this work. Based on the literature, the leaf is the characteristic and the simplest part of the plant to be analyzed in order to recognize it. The Artificial Neural Networks (ANN) in conjunction with Digital Image Processing (DIP) provide a high capacity for problem solving, and you can find several applications that use these techniques together. This work presents the use of digital processing techniques to extract image features of the leaves of these two species of *Plectranthus* family. In this way, samples were collected from these species and extracted anatomical and morphological characteristics which composes where these characteristics are used to compose a database to be used as input parameters for an RNA Multi Layer Perceptron type developed here, which was able to correctly classify the two species with a 96.15% success rate. The resultados obtained with the training of RNA enabled the development of an intelligent mobile application for recognition of medicinal plants from the present characteristics analysis on its leaves. As a result, there was obtained an application capable of recognizing plants quite satisfactorily studied species demonstrating that using RNAs it is possible to classify two distinct species and thereby perform intelligent production systems.

KEY-WORDS: Pattern Recognition. Artificial Neural Networks. Multi Layer Perceptron. Digital Image Processing. Intelligent application.

LISTA DE ILUSTRAÇÕES

Figura 1. Etapas normalmente envolvidas em um sistema de reconhecimento de padrão.....	20
Figura 2. Neurônio artificial.....	26
Figura 3. Arquitetura da Rede <i>Perceptron</i> Multicamadas.....	29
Figura 4. <i>Plectranthus Amboinicus</i>	40
Figura 5. <i>Plectranthus Barbatus</i>	40
Figura 6. Etapas do pré-processamento da imagem. (a) Imagem da folha obtida; (b) imagem convertida para escala de cinza; (c) Imagem binarizada; (d) imagem obtida após a aplicação do filtro de <i>Sobel</i>	42
Figura 7. Representação da melhor configuração da RNA encontrada pelo processo de validação cruzada.....	47
Figura 8. Gráfico resultante do conjunto de treinamento, realizado na rede neural MLP.....	47
Figura 9. Gráfico de comparação entre os novos dados inseridos com as saídas desejadas, sob o conjunto de teste.....	48
Figura 10. Relação dos resultados obtidos durante as fases de treinamento, validação e teste. Saídas desejadas (o) Saídas previstas pela RNA (●).....	49
Figura 11. Interface de entrada do aplicativo desenvolvido para realizar o reconhecimento das folhas.....	50
Figura 12. Tela de Captura da folha.....	51
Figura 13. Resultado do processo de captura da imagem da folha.....	51
Figura 14. Resultado do processamento realizado pelo aplicativo implementado....	52

LISTA DE TABELAS

Tabela 1. Dados adquiridos com o processamento de imagens.....	43
Tabela 2. Conjunto de dados de treinamento, validação e teste.....	44
Tabela 3. Relação entre a resolução e a matriz de pixels dos dispositivos utilizados para realizar a captura das imagens.....	44

LISTA DE ABREVIATURAS E SIGLAS

ACO	Ant Colony Optmization
ADT	<i>Android Development Tools</i>
AG	Algoritmos Genéticos
AIS	<i>Artificial Immune Systems</i>
API	Application Programming Interface
APK	Android Package
CE	Computação Evolutiva
CI	<i>Computational Intelligence</i>
CS	Classifier Systems
DNA	<i>DeoxyriboNucleic Acid</i>
EDGE	Enhanced Data Rates For GSM Evolution
EE	Estratégias Evolutivas
EE	<i>Enterprise Edition</i>
EP	Evolutionary Programming
ES	Evolution Strategies
GA	Genetic Algorithms
GP	Genetic Programming
GPS	<i>Global Positioning System</i>
GSM	Groupe Special Mobile
IC	Inteligência Computacional
IDE	<i>Integrated Development Environment</i>
JDK	Java Development Kit
MAPE	Erro Médio Absoluto Percentual
MLP	<i>Multi Layer Perceptron</i>

OMS	Organização Mundial de Saúde
PDI	Processamento Digital de Imagem
PE	Programação Evolutiva
PG	Programação Genética
PMC	<i>Perceptron</i> multicamadas
PSO	Particle Swarm Optimization
RBF	<i>Radial Basis Function</i>
RNAs	Redes Neurais Artificiais
RP	Reconhecimento de padrões
SC	Sistemas Classificadores
SDK	<i>Software Development Kit</i>
SI	Sistema Imunológico
SIA	Sistemas Imunológicos Artificiais

SUMÁRIO

1 INTRODUÇÃO.....	14
2 REVISÃO DE LITERATURA.....	18
2.1 RECONHECIMENTO DE PADRÕES.....	18
2.1.1 Padrão e Classe.....	19
2.1.2 Sistema de Reconhecimento de Padrões.....	20
2.1.3 Abordagem estatística.....	22
2.1.4 Abordagem Sintática.....	22
2.1.5 Abordagem Neural.....	23
2.1.6 Abordagem Difusa.....	23
2.2 INTELIGÊNCIA COMPUTACIONAL.....	24
2.2.1 Redes Neurais Artificiais.....	24
2.2.1.1 Arquitetura de uma RNA.....	26
2.2.1.1.1 Rede <i>feedforward</i> de camada simples.....	27
2.2.1.1.2 Rede <i>feedforward</i> de camadas múltiplas.....	27
2.2.1.1.3 Rede recorrente.....	27
2.2.1.2 Treinamento.....	27
2.2.1.2.1 Aprendizado supervisionado.....	28
2.2.1.3 Redes <i>Perceptron</i> Multicamadas.....	28
2.2.2 Computação Evolutiva.....	30
2.2.3 Inteligência de Enxames.....	33
2.2.4 Sistemas Imunológicos Artificiais.....	35
2.2.5 Sistemas <i>Fuzzy</i>.....	38
3 MATERIAIS E MÉTODOS.....	40

3.1 COLETA DAS AMOSTRAS E AQUISIÇÃO DAS IMAGENS.....	40
3.2 PROCESSAMENTO DAS IMAGENS.....	41
3.2.1 Pré-processamento da imagem.....	41
3.2.2 Extração das Características.....	43
3.3 REDE NEURAL ARTIFICIAL	44
3.4 IMPLEMENTAÇÃO DO APLICATIVO.....	45
4 RESULTADOS E DISCUSSÕES.....	46
4.1 PROCESSAMENTO DIGITAL DAS IMAGENS.....	46
4.2 REDE NEURAL ARTIFICIAL.....	46
4.3 APLICAÇÃO INTELIGENTE.....	49
5 CONSIDERAÇÕES FINAIS.....	53
REFERÊNCIAS.....	54

1 INTRODUÇÃO

Durante toda vida, em vários momentos, o ser humano se utiliza do reconhecimento de padrões para realizar a identificação de tudo aquilo que o rodeia. Esse reconhecimento é baseado em características específicas que são extraídas de cada objeto, como cor, largura, altura, área, forma geométrica, dentre outras várias (RAMOS, 2005).

O Reconhecimento de padrões (RP) é, por natureza, uma ciência inexata, e assim admite muitas abordagens, algumas vezes complementares, outras competitivas, para aproximar, indicar ou encontrar soluções a um dado problema. Em problemas complexos de RP, a abordagem comum utilizada é dividir o problema em dois módulos subsequentes: a extração de características e o módulo de classificação. Tanto para o primeiro como para o segundo módulo já existem várias abordagens desenvolvidas que vão desde abordagens estatísticas e matemáticas até abordagens de Inteligência Artificial (RAMOS, 2005).

Dentre as técnicas utilizadas para reconhecimento de padrões, temos as de Inteligência Computacional, uma área da Inteligência Artificial que investiga sistemas computacionais que apresentam comportamento inteligente, por meio de métodos sub-simbólicos baseados em modelos matemáticos e de cálculo numérico. A Inteligência Computacional utiliza principalmente três técnicas: Redes Neurais Artificiais (RNAs), Lógica Fuzzy e Algoritmos Evolutivos (LIBERMAN, 1997). As RNAs tiveram um crescente uso devido a sua capacidade de fazer suposições mais delicadas a respeito da distribuição dos dados de entrada do que métodos estatísticos tradicionais e a capacidade de formar fronteiras de decisão altamente não lineares no espaço de características (RAMOS, 2005). As RNAs são baseadas no funcionamento de neurônios biológicos, onde, assim como um neurônio biológico, o neurônio artificial possui uma estrutura relativamente simples e a estrutura deste neurônio é capaz de responder a estímulos de outros neurônios a ele conectados (RAMOS, 2005)(ROBERTO; FILHO, 2009).

Devido à ampla aplicação das RNAs como, por exemplo, aproximação universal de funções, controle de processos, agrupamento de dados, otimização de sistemas e sistemas de previsão, as mesmas também são utilizadas para reconhecer e classificar padrões, dentre estes, os padrões em imagens, onde são utilizadas como entradas para a RNA um conjunto de características extraídas da

imagem por intermédio do Processamento Digital de Imagem (PDI) (LIBERMAN, 1997). O uso de PDI tem por objetivo melhorar o aspecto visual de certas feições estruturais e fornecer diversos subsídios para a interpretação, gerando produtos que possam ser posteriormente submetidos a outros processamentos (RAMOS, 2005). As entradas a serem utilizadas na RNA sofrem um pré-processamento, ou seja, uma melhoria antes de serem processadas, de maneira a conseguir um melhor resultado (RAMOS, 2005).

Um dos grandes problemas, assim como o reconhecimento de faces, é o reconhecimento de plantas, devido a sua grande biodiversidade e além desta, a complexidade geométrica de suas partes como raízes, flores, caule, frutos e sementes (BORGES, 2013).

As plantas estão presentes em todos os lugares e detêm um papel importante no meio ambiente, sendo um dos principais, a presença de vida na Terra (BORGES, 2013). Além deste motivo, as plantas podem ser utilizadas na produção de alimentos, tecidos e principalmente, medicamentos. Plantas medicinais são utilizadas há vários anos para curar uma variedade de doenças, e de acordo com a Organização Mundial de Saúde (OMS, 1997), o uso das plantas medicinais como uma forma de prevenir e tratar doenças têm sido bem difundido, onde cerca de 80% da população mundial já fez uso de algum tipo de planta para aliviar alguma sintomatologia. As plantas medicinais apresentam propriedades farmacológicas que são de extrema importância para o bom funcionamento dos organismos dos seres vivos (YAMASHITA, 2005), contudo, as plantas podem possuir várias espécies diferentes, o que poderá contribuir para que sejam utilizadas de forma incorreta, visto que podem conter propriedades que também prejudicam o organismo de um ser vivo.

Muitos são os métodos utilizados para reconhecer plantas, os quais vão desde a análise morfológica da planta até métodos mais modernos, como a análise do DNA (KRESS et al., 2005). Porém, muitos destes métodos se utilizam de características complexas e podem ser realizados apenas por profissionais que tenham um amplo conhecimento. Muitas vezes algumas das técnicas de reconhecimento acabam por depender da interpretação de quem a está utilizando, com exceção de métodos que envolvam o estudo do DNA, o qual também é bastante complexo e somente feito em laboratório. A partir de estudos sobre a literatura, identifica-se a folha como sendo a característica e a parte mais simples da

planta a ser analisada para realizar o reconhecimento de uma planta (BORGES, 2013).

A principal finalidade deste trabalho é a identificação e diferenciação de duas plantas medicinais amplamente utilizadas, o Boldo (*Plenctranthus barbatus*) e a Malva (*Plenctranthus amboinicus*), por meio do uso de uma RNA implementada dentro de uma aplicação para dispositivos móveis, especificamente para plataforma *Android*. Para atender o objetivo proposto, foram estabelecidos os seguintes objetivos específicos: analisar as folhas de ambas as espécies e identificar características capazes de diferenciá-las; extrair de uma série de imagens as características escolhidas utilizando um algoritmo de PDI; treinar a RNA utilizando-se os produtos resultantes do PDI como entrada; implementar a RNA juntamente ao algoritmo de PDI na linguagem utilizada pela plataforma *Android* para gerar a aplicação.

A escolha do Boldo e da Malva deu-se por estes fazerem parte de uma mesma família com um grande número de espécies como: *Plenctranthus barbatus*, *Plenctranthus neochilus*, *Plenctranthus ernestii* e *Plenctranthus coleoides Benth* e dentre elas somente as duas analisadas (*Plenctranthus barbatus* e *Plenctranthus amboinicus*) podem ser utilizadas como medicamento benéfico ao organismo, além disso, o Boldo e a Malva são plantas medicinais amplamente utilizadas pela medicina popular e pela medicina tradicional, na produção de medicamentos e no tratamento de doenças. A plataforma *Android* foi escolhida por ser uma plataforma livre e de grande acessibilidade e ter um canal próprio disponibilizado para as aplicações construídas para ela, facilitando a divulgação e propagação da aplicação, a fim de que esta seja utilizada para prevenir o mau uso das plantas estudadas.

Além deste capítulo introdutório, que apresentou uma visão geral deste trabalho, este documento foi estruturado da seguinte forma:

No Capítulo 2, "Revisão de Literatura", serão apresentados alguns conceitos relevantes para a compreensão do tema atual desta pesquisa, onde será feita uma revisão sobre Reconhecimento de Padrões e sobre as técnicas de Inteligência Computacional.

O Capítulo 3, "Materiais e Métodos", apresenta o procedimento experimental realizado nesta pesquisa, o desenvolvimento de uma RNA para classificação das espécies *Plectranthus amboinicus* (boldo) e *Plectranthus barbatus*

(malva), bem como o desenvolvimento da aplicação para dispositivos móveis que utilizam a plataforma *Android*.

No Capítulo 4, “Resultados e Discussões”, serão apresentados os resultados desta pesquisa tanto no que diz respeito ao Processamento Digital das Imagens, quanto à melhor RNA encontrada para solucionar o problema de classificação em questão, também serão demonstrados o método de desenvolvimento e as telas da aplicação implementada neste trabalho.

O capítulo 5, “Considerações Finais”, mostra algumas considerações finais à cerca desta pesquisa, onde serão apresentadas também, algumas sugestões de melhorias para trabalhos futuros.

2 REVISÃO DE LITERATURA

2.1 RECONHECIMENTO DE PADRÕES

Reconhecimento de Padrões (RP) é provavelmente uma das áreas da ciência de maior abrangência, estudada em diversos campos de pesquisa como biologia, psicologia, medicina, marketing, finanças, meteorologia, sensoriamento remoto, processamento de imagens, entre outros (SOUZA, 1999).

Segundo Souza (1999), RP é, por natureza, uma ciência inexata, e assim admite muitas abordagens, algumas vezes complementares, outras competitivas, para aproximar, indicar ou encontrar soluções a um dado problema. A primeira explicação de um fato ou entendimento de um acontecimento novo está sempre ligada aos padrões do mesmo.

Existem muitas definições e muitas abordagens sobre Reconhecimento de Padrões. Duda e Hart, (1973), caracterizaram RP como: "campo que consiste no reconhecimento de regularidades significativas em meios ruidosos e complexos." Já Bezdek e Pal (1992) definem RP como: "a busca por estruturas em dados".

Segundo Pao (1989), o conhecimento de reconhecimento de padrões é importante devido às ocorrências na vida humana tomarem forma de padrões. A formação da linguagem, o modo de falar, o desenho das figuras, o entendimento das imagens, tudo envolve padrões.

RP é uma tarefa complexa, onde o homem busca, sempre, avaliar as situações em termos dos padrões das circunstâncias que as constituem, descobrir relações existentes no meio, para melhor entendê-lo e adaptar-se (SOUZA, 1999).

O Reconhecimento de Padrões, assim como a Inteligência Artificial, teve seu início nos anos 50. O RP inicialmente utilizou-se de técnicas probabilísticas, mais precisamente, da Estimação e Teoria da Decisão para sua fundamentação. Assim, usou a linguagem da Probabilidade em sua origem, usando a abordagem Bayesiana (SCHALKOFF, 1992). Os métodos de classificação podem ser divididos em quatro grupos: RP estatístico, RP sintático, RP neural e RP difuso (PAO, 1989; FU, 1994; BISHOP, 1995).

O estudo do reconhecimento de padrões pode ser dividido em duas categorias básicas (TOU, GONZALES, 1981): o estudo de seres humanos e

organismos vivos com o objetivo de se estabelecer os modos pelos quais os mesmos desenvolvem e aprimoram suas capacidades de reconhecimento de padrões e o desenvolvimento de teorias e técnicas objetivando a construção de máquinas ou dispositivos capazes de apresentar características semelhantes às dos seres humanos em reconhecerem padrões.

O termo reconhecimento de padrões envolve uma gama significativa de problemas de processamento de informação, com grande relevância prática, desde o reconhecimento de voz e de caracteres feitos manualmente, até à detecção de erros em equipamentos ou diagnósticos médicos. Não é possível abordar todas as áreas onde seria útil e, por vezes mesmo necessário, aplicar RP. Deve-se levar em conta que as pessoas aplicam seus próprios métodos de reconhecimento em praticamente todas as áreas da atividade humana. Na prática computacional, pode-se enumerar algumas áreas onde vem sendo usado RP (KASABOV, 1996; SMITH, 1998; TAFNER, 1998):

- Análise, segmentação e pré-processamento de imagens;
- Reconhecimento de faces;
- Identificação de impressões digitais;
- Reconhecimento de caracteres;
- Análise de manuscritos;
- Computer Vision;
- Entendimento e reconhecimento de voz;
- Diagnóstico médico.

2.1.1 Padrão e Classe

Entende-se por padrão as propriedades que possibilitam o agrupamento de objetos semelhantes dentro de uma determinada classe ou categoria, mediante a interpretação de dados de entrada, que permitam a extração das características relevantes desses objetos (TOU, GONZALES, 1981). Entende-se por classe de um padrão um conjunto de atributos comuns aos objetos de estudo. Assim, reconhecimento de padrões pode ser definido como sendo um procedimento em que se busca a identificação de certas estruturas nos dados de entrada em comparação

a estruturas conhecidas e sua posterior classificação dentro de categorias, de modo que o grau de associação seja maior entre estruturas de mesma categoria e menor entre as categorias de estruturas diferentes. Os dados de entrada são medidos por sensores e selecionados segundo o conteúdo de informações relevantes para a decisão, e passam por um processo de redução de sua dimensionalidade para que possam ser usados pelo classificador, que o designará à classe que melhor o represente (PRADO, 2002).

2.1.2 Sistema de Reconhecimento de Padrões

Os sistemas de reconhecimento de padrões seguem normalmente uma sequência de etapas conforme a Figura 1. Cada uma destas etapas tem um grau de complexidade que varia de acordo com o problema de reconhecimento de padrão em questão. Certamente, para que o sistema funcione perfeitamente, será necessário o correto desenvolvimento e otimização de cada uma das etapas (DUDA, HART, STORK, 2001).

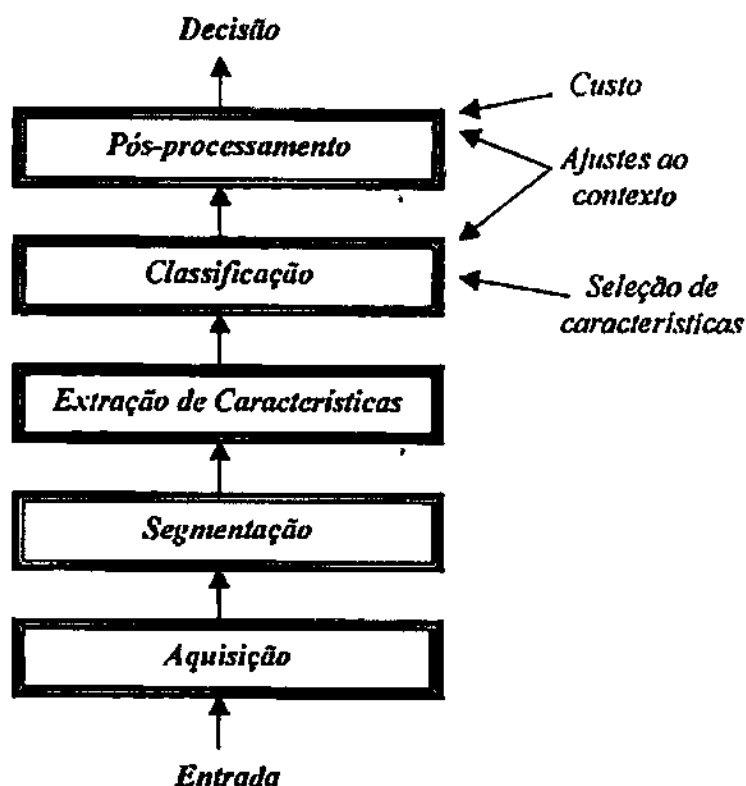


Figura 1. Etapas normalmente envolvidas em um sistema de reconhecimento de padrão
Fonte: (SILVA, 2003)

A primeira etapa é de aquisição dos sinais ou imagens, formando os dados de entrada para o sistema. Nesta etapa, recorre-se frequentemente a procedimentos para pré-processamento e normalização de dados (FUKUNAGA, 1990).

A segunda etapa é responsável pela segmentação, que de uma maneira simples para um sistema, por exemplo, envolvendo aquisição de imagens, define-se como a separação dos objetos de interesse na imagem dos outros objetos contendo informação irrelevante (*background*). Duda e Hart (1973) afirmam que o problema de segmentação é um dos problemas mais complexos em reconhecimento de padrões, citando, por exemplo, a problemática existente em sistema de reconhecimento de voz.

A terceira etapa, extração de características, é responsável pela formação de um conjunto de medidas ou valores que sejam similares para objetos pertencentes a mesma classe, e diferentes para objetos de classes distintas. Um bom sistema de extração de características deverá facilitar o trabalho do classificador. Desta forma, é recomendável que seja feita uma seleção adequada de características a serem usadas para um dado sistema de reconhecimento de padrões. É importante a escolha de características realmente relevantes para a discriminação das classes estudadas e, preferencialmente, que sejam de extração fácil. Resumindo, um conjunto ideal de características deverá ser composto pelo menor número possível de características de extração fácil, sem detrimento do desempenho do classificador (DUDA, HART, STORK, 2001).

A quarta etapa, a classificação propriamente dita, tem como objetivo usar o vetor de entrada, formado pelas características, para atribuir um objeto a uma classe. O grau de dificuldade do problema de classificação está fortemente associado à variabilidade existente nos valores das características de objetos de mesma classe, e à diferença de valores das características de objetos de classes distintas. Existem diversos tipos de problemas envolvidos nesta etapa, mas os problemas principais são encontrar qual o tipo de classificador adequado para um dado problema, e obter o melhor desempenho possível (DUDA, HART, STORK, 2001).

A quinta etapa, que pode ou não existir em determinados sistemas, envolve normalmente uma tomada de decisão a partir do resultado da classificação.

A etapa de classificação pode ser desenvolvida usando-se várias técnicas, como as que empregam classificadores estatísticos, as redes neurais artificiais, lógica difusa e sistemas híbridos. O que varia de um classificador para o outro é o algoritmo de treinamento ou aprendizagem (SILVA, 2003).

2.1.3 Abordagem estatística

Robert Schalkoff (1992) diz que o reconhecimento de padrões estatístico, como sugere o próprio nome, assume uma base estatística para os algoritmos de classificação. Um conjunto de medidas, que denotam as características, é extraído dos dados de entrada e usado para associar cada vetor de características a uma dentre as classes. Presume-se que as características são geradas por um estado da natureza, e, portanto, existe um modelo subordinado a um estado com um conjunto de probabilidades e/ou funções de densidade de probabilidade correspondente, passível de ser usado para representá-lo.

Segundo Bishop (1995), a forma mais geral e natural de formular soluções para o reconhecimento de padrões é o RP estatístico, por meio do qual é reconhecida a natureza estatística tanto da informação que se quer representar quanto dos resultados que devem ser expressos.

2.1.4 Abordagem Sintática

Muitas vezes, as informações significativas em um padrão não consistem apenas na presença ou ausência de valores numéricos de um conjunto de características. Ao invés disto, a inter-relação ou interconexão das características produzem informações estruturais importantes, que facilitam a descrição ou classificação. Esta é a base do RP sintático. Portanto é necessário, ao se usar a abordagem de RP sintático, quantificar e extrair as informações estruturais importantes para poder avaliar a similaridade estrutural entre os padrões. Para a abordagem sintática, ao enfatizar a descrição estrutural dos padrões, tenta-se fazer uma analogia entre as estruturas do padrão e a sintaxe de uma linguagem (BISHOP, 1995).

Tipicamente, o RP sintático formula uma descrição hierárquica de padrões complexos, construída a partir de sub-padrões mais simples, sendo que no

nível mais baixo se encontram os elementos mais simples, extraídos dos dados de entrada que são chamados de primitivas. Por exemplo, as frases e as sentenças são construídas juntando-se palavras, e palavras são construídas juntando-se letras (FU, 1982).

2.1.5 Abordagem Neural

Com a capacidade intrínseca de aprender, e o fato de serem aproximadores universais, as redes neurais surgiram como uma ferramenta poderosa na área de reconhecimento de padrões. Sua capacidade de fazer suposições mais delicadas a respeito da distribuição dos dados de entrada do que métodos estatísticos tradicionais e a capacidade de formar fronteiras de decisão altamente não-lineares no espaço de características, levou ao seu crescente uso.

A tarefa computacional realizada pelas redes neurais pode ser, por exemplo, a classificação de padrões, ou seja: dado um vetor de características, deve-se associá-lo a uma classe em um determinado problema (*i.e.*, reconhecimento de padrões). Na maioria das aplicações de RP usando-se redes neurais, são estabelecidas conexões entre os valores de todas as características que definem os padrões com camadas intermediárias de neurônios e a todas as classes objetivos que são consideradas como a camada de saída. O treinamento da rede é realizado corrigindo os pesos nas conexões para estabelecer-se as relações entre as características e classes que promovam a melhor discriminação possível entre os padrões de classes diferentes. Assim, ao ser apresentado à rede um novo padrão, esta indicará a classe que o representa melhor na camada de saída (SCHALKOFF, 1992).

2.1.6 Abordagem Difusa

Segundo Klir e Yuan (1995), a utilidade da teoria dos conjuntos difusos no reconhecimento de padrões já foi reconhecida em meados da década de 60, e hoje a literatura a respeito de RP difuso é bastante extensa. Existem duas formas clássicas de RP difuso, são elas: os métodos de reconhecimento de padrões por meio de listas de pertinências e os métodos difusos sintáticos. No método clássico da lista de pertinência, cada classe é caracterizada por um conjunto de padrões que

é armazenado no sistema de reconhecimento de padrões. Um padrão desconhecido a ser classificado é comparado com os padrões armazenados um a um. O padrão é classificado como um membro de uma classe se ele combinar com um dos padrões pertencentes a esta classe.

No método sintático difuso, um padrão é representado por uma cadeia de sub-padrões concatenados chamados de primitivas. Estas primitivas são vistas como o alfabeto de uma linguagem formal. Um padrão, então, é uma sentença gerada por alguma gramática. Todos os padrões, cujas sentenças são geradas pela mesma gramática, pertencem a uma mesma classe.

Um padrão desconhecido é classificado como pertencente a uma classe particular se ele puder ser gerado pela gramática correspondente a ela (KLIR, YUAN, 1995).

2.2 INTELIGÊNCIA COMPUTACIONAL

A Inteligência Computacional (IC), do inglês *Computational Intelligence* (CI) é a área da ciência que estuda a teoria e a aplicação de técnicas inspiradas na natureza, buscando o desenvolvimento de sistemas inteligentes que imitem aspectos do comportamento humano, tais como: aprendizado, percepção, raciocínio, evolução e adaptação (ENGELBRECHT, 2007). Dentre essas técnicas se destacam: Redes Neurais Artificiais; Computação Evolutiva; Inteligência de Enxames; Sistemas Imunológicos Artificiais e Sistemas *Fuzzy*.

2.2.1 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes. Estas redes adquirem conhecimento por experiência e podem ser utilizadas na resolução de uma grande classe de problemas encontrados em várias áreas aplicadas, como: classificação, identificação, diagnóstico, análise de sinais e de imagens, otimização e controle (SILVA, 2003).

As RNAs possuem característica multidisciplinar (neurociência, matemática, estatística, física, ciência da computação e engenharia) o que tornou possível o desenvolvimento de muitos tipos e muitas aplicações envolvendo o seu

uso, tanto na área acadêmica, industrial, comercial, como de serviços. Os tipos de aplicações são os mais diversos, como modelagem, análise de séries temporais, reconhecimento de padrões, processamento de sinais de controle entre outros; tudo isto devido a uma importante propriedade: a de aprender a partir dos dados de entrada (BONIFACIO, 2010).

As principais características das RNAs são (SILVA *et al.*, 2010):

- **Capacidade de aprendizado:** por meio de um processo de treinamento, a partir de um banco de dados existente, a rede consegue extrair uma relação entre as diversas variáveis que compõe a aplicação;
- **Habilidade de generalização:** uma vez que a rede tenha realizado o treinamento, ela consegue generalizar o conhecimento dando a oportunidade de estimar soluções para mesma aplicação formada por dados externos ao banco de dados de treinamento;
- **Possibilidade de prototipagem:** após o processo de treinamento e generalização da rede, é possível implementá-la em *hardware* ou *software*.

Nas RNAs, as unidades de processamento de informação fundamentais para o seu funcionamento são os neurônios artificiais (Figura 2), os quais consistem em estruturas lógicas com o objetivo de simular o funcionamento de um neurônio biológico, fazendo com que a rede atue de forma indutiva, sendo que seu funcionamento pode ser descrito conforme apresentado a seguir (COSTA, ORTOLAN, 2014):

- Um conjunto de valores é apresentado como variável de entrada;
- As variáveis de entrada são multiplicadas por pesos sinápticos respectivos conforme a influência de cada valor;
- O potencial de ativação neural é obtido pela soma ponderada dos sinais de entrada, subtraindo-se o limiar de ativação;
- A saída do neurônio é limitada utilizando uma função de ativação apropriada;

- O sinal de saída é produzido relacionado a cada um dos conjuntos de variáveis de entrada.

Com base no funcionamento dos neurônios artificiais as RNAs também podem ser definidas como um conjunto de neurônios artificiais organizados em camadas, na qual podemos identificar a entrada, os neurônios internos e os de saída. A arquitetura da rede é definida pela forma na qual esses neurônios estão organizados e interconectados, ou seja, o número de camadas, o número de neurônios por camada, tipos de conexão entre os neurônios e a topologia da rede (SILVA, 2009).

A Figura 2 ilustra o modelo de um neurônio artificial. Os sinais de entrada são representados pelos valores $x_1, x_2, x_3, \dots, x_n$, que representam impulsos elétricos externos em um neurônio biológico. Os pesos sinápticos ponderando a relevância de cada entrada são representados pelo conjunto de valores $w_1, w_2, w_3, \dots, w_n$, que serão os índices de multiplicação dos valores de entrada. Desta forma, por meio de um limiar de ativação θ , a soma ponderada das entradas u é limitada pela função de ativação $g(u)$ apresentado o resultado em y (SANTOS, 2013).

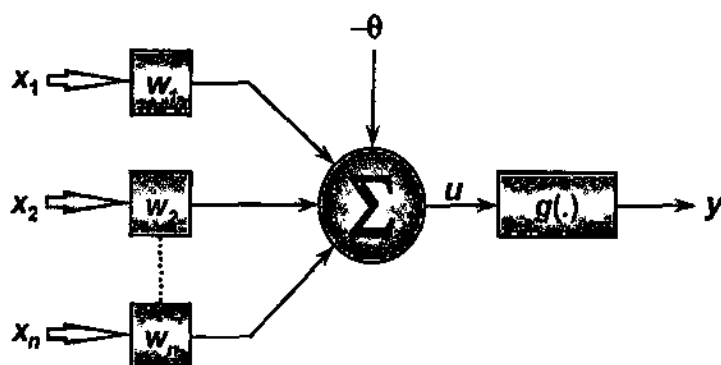


Figura 2. Neurônio artificial

Fonte: (SILVA *et al.*, 2010)

2.2.1.1 Arquitetura de uma RNA

A arquitetura de uma RNA diz respeito à forma como os diversos neurônios que compõem a rede estão dispostos uns relação aos outros, ou seja, como as conexões sinápticas estão estruturadas (Silva, *et al.*, 2010). As principais

arquiteturas das RNAs, considerando a disposição dos neurônios, formas de interligação e constituição de camadas, são as redes *feedforward* de camada simples ou camadas múltiplas e as redes recorrentes (SANTOS, 2013).

2.2.1.1.1 Rede *feedforward* de camada simples

São redes que contém apenas uma camada de entrada e a camada de saída é a própria camada de neurônios, onde a quantidade de saídas será igual ao número de neurônios da rede. Estas redes são empregadas em aplicações de filtragem linear, memória associativa e classificação de padrões. São exemplos desta arquitetura as redes *Perceptron* e *Adaline* (SANTOS, 2013).

2.2.1.1.2 Rede *feedforward* de camadas múltiplas

São redes que possuem uma ou mais camadas de neurônios escondidas e são aplicadas em diversos problemas tais como aproximação de funções, classificação de padrões, otimização e controle de processos. São exemplos desta rede o *Perceptron* multicamadas MLP (*Multi Layer Perceptron*) e as redes de base radial RBF (*radial basis function*) (SANTOS, 2013).

2.2.1.1.3 Rede recorrente

Este tipo de rede possui realimentação entre diferentes camadas de neurônios onde as saídas dos neurônios são realimentadas como sinais de entradas em outros neurônios. As redes recorrentes são utilizadas em sistemas variantes no tempo, tais como: controle de processos e previsão de séries temporais. Os principais tipos destas redes são o modelo de *Hopfield* e o *Perceptron* multicamadas (SANTOS, 2013).

2.2.1.2 Treinamento

O treinamento da rede consiste na adequação iterativa dos pesos sinápticos e limiares de ativação de todos os neurônios da rede e encerra-se quando a RNA consegue generalizar soluções para uma classe de problemas cujas

respostas são representativas na aplicação que está sendo processada (SANTOS, 2013).

O conjunto de valores reais conhecidos (amostras) normalmente é dividido em subconjuntos de treinamento e teste da RNA. Cerca de 60 a 90% das amostras é separada para efetuar o aprendizado da RNA, a fim de que esta aprenda as regras do processo e ajuste os pesos sinápticos e o limiar de ativação. Os demais dados ou amostras são apresentados à RNA após a fase de treinamento com o objetivo de verificar e testar se a generalização produzida pela rede resulta em saídas adequadas para os dados não apresentados durante o treinamento. Caso as saídas dos dados de testes sejam adequadas, a rede obteve um grau de generalização aceitável e, assim, obtém-se a validação da topologia empregada (SILVA; SPATTI; FLAUZINO, 2010).

2.2.1.2.1 Aprendizado supervisionado

O aprendizado supervisionado é aquele no qual é apresentada para a rede uma tabela de dados com entradas e saídas conhecidas caracterizando qual é o valor adequado ou inadequado para cada conjunto de entradas apresentado. Assim, a RNA é treinada para fornecer a saída desejada em relação a uma entrada específica (HAYKIN, 2001).

Quando um vetor de entrada é aplicado, a saída da rede é calculada e comparada com o padrão de saída da tabela de dados e os pesos sinápticos e limiares são ajustados continuamente, aplicando ações comparativas, executadas pelo próprio algoritmo de aprendizagem. Estas supervisionam a defasagem entre as respostas produzidas pela rede em relação às desejadas, onde, uma vez encontrada diferenças, são aplicados os procedimentos de ajustes (SANTOS, 2013; SILVA; SPATTI; FLAUZINO, 2010).

2.2.1.3 Redes *Perceptron* Multicamadas

As redes *Perceptron* de múltiplas camadas (MLP, *Multi Layer Perceptron* ou PMC) são caracterizadas pela existência de pelo menos uma camada de neurônios intermediária ou escondida situada entre a camada de entrada e a camada neural de saída. As redes MLP são aplicáveis a diversas áreas, em função

da sua versatilidade de aplicações (SILVA; SPATTI; FLAUZINO, 2010). Dentre estas áreas destacam-se: aproximador universal de funções; reconhecimento de padrões; previsão de séries temporais; otimização de sistemas; identificação e controle de processos. A Figura 3 mostra a representação de uma rede *Perceptron* multicamadas.

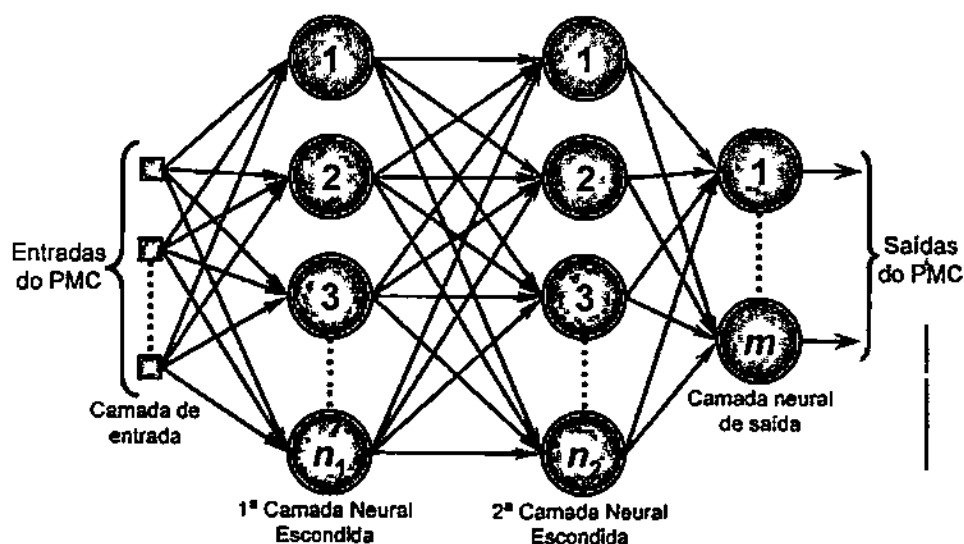


Figura 3. Arquitetura da Rede *Perceptron* Multicamadas

Fonte: (SILVA *et al.*, 2010)

O treinamento de uma rede MLP é feito utilizando-se o algoritmo *backpropagation*, conhecido também como regra Delta generalizada, a qual baseia-se na heurística de aprendizado por correção de erro (SCHIAVONI, 2010).

A estratégia de treinamento da rede por intermédio deste algoritmo, pode ser descrita da seguinte forma:

- Um padrão de entrada é apresentado à camada de entrada da rede;
- A atividade resultante é propagada pela rede, camada por camada, até que a camada de saída produza uma resposta;
- A saída obtida é comparada com a saída real desejada;
- O erro da saída é calculado e propagado a partir da camada de saída até a camada de entrada;
- Os pesos sinápticos e os limiares de ativação são alterados gradativamente a cada iteração, conforme o erro é retropropagado.

- Este processo é repetido para todos os vetores de entrada da rede até que o erro quadrático médio das saídas da rede esteja dentro de um valor admissível.

Uma das formas de potencializar o desempenho do treinamento da MLP é utilizar o método de validação cruzada (*cross-validation*), cujo objetivo é avaliar a aptidão de cada uma das topologias encontradas para as RNAs, quando aplicadas a um conjunto de dados diferentes daqueles usados no ajuste de seus parâmetros internos. Em outras palavras, a validação cruzada é utilizada para assegurar a generalização da rede, garantindo que a mesma possa responder com precisão a outras entradas que não foram utilizadas durante o treinamento (TARALLO, 2007).

2.2.2. Computação Evolutiva

Computação Evolutiva (CE) é um ramo de pesquisa emergente da Inteligência Computacional que propõe um novo paradigma para solução de problemas e compreende um conjunto de técnicas de busca e otimização inspiradas na evolução natural das espécies (LUKE, 2000). Seus conceitos têm sido empregados em uma variedade de disciplinas, desde ciências naturais e engenharia até biologia e ciência da computação (ZUBEN, 2001).

A vantagem mais significativa da computação evolutiva está na possibilidade de resolver problemas pela simples descrição matemática do que se quer ver presente na solução, não havendo necessidade de se indicar explicitamente os passos até o resultado, que certamente seriam específicos para cada caso. É lógico que os algoritmos evolutivos correspondem a uma sequência de passos até a solução, mas estes passos são os mesmos para uma ampla gama de problemas, fornecendo robustez e flexibilidade. Sendo assim, a computação evolutiva deve ser entendida como um conjunto de técnicas e procedimentos genéricos e adaptáveis, a serem aplicados na solução de problemas complexos, para os quais outras técnicas conhecidas são ineficazes ou nem sequer são aplicáveis (ZUBEN, 2001).

A CE trata da linha de pesquisa dos algoritmos evolutivos que tem por base a teoria neo-Darwiniana da evolução e baseado nesta teoria é possível propor como características para os algoritmos evolutivos a reprodução, variação genética

e seleção. Essas características descrevem uma geração, que ocorre quando a reprodução, variação genética e seleção são executadas dentro do algoritmo evolutivo (CASTRO, ZUBEN, 2001).

Dentre os algoritmos evolutivos, os principais são:

- Algoritmos Genéticos (AG) - Genetic Algorithms (GA);
- Estratégias Evolutivas (EE) - Evolution Strategies (ES);
- Programação Evolutiva (PE) - Evolutionary Programming (EP);
- Programação Genética (PG) - Genetic Programming (GP);
- Sistemas Classificadores (SC). - Classifier Systems (CS).

Os algoritmos genéticos são técnicas de busca que utilizam procedimentos iterativos que simulam o processo de evolução de uma população de possíveis soluções de um determinado problema. O processo de evolução é aleatório, porém, guiado por um mecanismo de seleção baseado na adaptação de estruturas individuais. A cada iteração do algoritmo (uma geração), um novo conjunto de estruturas é criado por meio da troca de informações (bits ou blocos) entre estruturas bem adaptadas selecionadas da geração anterior. Novas estruturas são geradas aleatoriamente com uma dada probabilidade e incluídas na população. O resultado tende a ser um aumento da adaptação de indivíduos ao meio, podendo acarretar também em um aumento global da aptidão da população a cada nova geração. Neste caso, a população evolui a cada geração se aproximando de uma solução ótima (SOBRINHO, 2003).

As Estratégias Evolutivas foram desenvolvidas focalizando a resolução dos problemas contínuos de otimizações paramétricas, sendo estendidas para tratamento de problemas discretos. Nas Estratégias Evolutivas, um indivíduo é representado por um par de vetores reais na forma $v = (x, \sigma)$, onde x representa o ponto de busca no espaço e o σ vetor de desvio padrão associado. Nas versões atuais, a descendência é obtida submetendo os indivíduos da geração a dois operadores: cruzamento e mutação. Observa-se que o parâmetro - que determina a mutação de x - também está sujeito ao processo de evolução. Esta é uma característica fundamental das estratégias evolutivas, que permite o auto ajuste de seus parâmetros. Assumindo algumas hipóteses, é possível provar que as

estratégias evolutivas convergem ao ótimo global com probabilidade 1 (um), considerando um tempo de busca suficientemente longo (WIJNS, BOSCHETTI, MORESI, 2003).

A Programação Evolutiva consiste na evolução de população com máquinas de estados finitos submetendo-as a transformações unitárias. Em PE, cada indivíduo da população é representado por uma máquina de estados finitos (MEF), que processa uma sequência de símbolos. Durante a avaliação, os indivíduos são analisados por uma função de *payoff* de acordo com a saída da máquina e a saída esperada para solução do problema. A reprodução é feita apenas por operadores de mutação, sendo que todos os indivíduos da população atual geram novos descendentes. Esse processo caracteriza a chamada reprodução assexuada. Na seleção de indivíduos para a próxima geração, os descendentes competem com os μ pais e somente os indivíduos com maior *fitness* (no caso, os de maior *payoff* entre os $\mu + \lambda$ indivíduos) sobrevivem (CARNEIRO, 2009).

A Programação Genética é uma técnica com o objetivo de gerar automaticamente programas, por meio da criação e manipulação de *software* geneticamente, com o emprego de conceitos obtidos da biologia. A ideia da Programação Genética é, a partir de dados iniciais, gerar programas na linguagem empregada, com o objetivo de encontrar o programa que melhor resolva um problema específico, conforme estabelecido nos dados iniciais (CARNEIRO, 2009). A geração dos novos programas, partindo de uma geração inicial de programas completamente aleatória, é realizada pela aplicação de operadores genéticos que efetuam uma recombinação entre as partes dos programas. Os operadores genéticos podem ser, entre outros, (CARNEIRO, 2009): reprodução; cruzamento; e, mutação. Além dos operadores genéticos é empregada uma função, denominada função de *fitness* (ou de adaptação) que irá guiar a seleção dos programas que mais se aproximam da solução (SHAOBO, GUANCI, QINGSHENG, 2008).

Os Sistemas Classificadores são sistemas baseados em regras proposicionais, e possuem as seguintes características: são indicados para operar em ambientes que tipicamente exibem eventos novos e sucessivos acompanhados de ruído e dados irrelevantes; são apropriados em situações em que haja a necessidade de agir de maneira contínua e frequentemente em tempo real; possuem a habilidade de se guiar por metas implícitas ou inexatas, bem como promovem o aprendizado baseado em recompensas esparsas (GABRIEL, DELBEM, 2008).

Essas características são possíveis porque os sistemas classificadores descobrem novas categorias e conceitos por meio de regularidades encontradas no ambiente, uma vez que estas sejam relevantes para o atendimento das metas desejadas. Do mesmo modo, os sistemas classificadores utilizam o fluxo de informação encontrado no caminho até a meta para refinar seu modelo do ambiente e desta forma associar ações de controle apropriadas às situações encontradas na busca (MOUSSI *et al.*, 2000).

Basicamente, o Sistema Classificador consiste de uma metodologia para criação e atualização evolutiva de regras (denominadas classificadores) em um sistema de tomada de decisão (HOLLAND, 1992). Dadas as características de um ambiente em um determinado instante e levando-se em conta a “energia” de cada classificador, alguns classificadores podem ser ativados. Eles codificam alternativas de ações específicas, as quais são submetidas a um processo de competição para selecionar aquela que será executada (VARGAS, 2000).

2.2.3 Inteligência de Enxames

A Inteligência de Enxames, também referenciada como Inteligência de Colônias ou Inteligência Coletiva (ESMIN, 2005; SERAPIÃO, 2009), é um conjunto de técnicas baseadas no comportamento coletivo de sistemas auto organizados, distribuídos, autônomos, flexíveis e dinâmicos. Estes sistemas são formados por uma população de agentes computacionais simples que possuem a capacidade de perceber e modificar o seu ambiente de maneira local. Esta capacidade torna possível a comunicação entre os agentes, que captam as mudanças no ambiente geradas pelo comportamento de seus congêneres. Embora não exista uma estrutura centralizada de controle que estabeleça como os agentes devam se comportar, e mesmo não havendo um modelo explícito do ambiente, as interações locais entre os agentes geralmente levam ao surgimento de um comportamento global que se aproxima da solução ótima (JUNIOR, 2012).

Nos modelos computacionais baseados em inteligência coletiva, o termo enxame pode ser aplicado de uma forma mais generalizada, sendo entendido como qualquer tipo de comportamento coletivo. Desta forma, os enxames podem ser aplicados em espaços de alta dimensionalidade em que as colisões entre os agentes (ocupação do mesmo espaço no mesmo instante de tempo) não têm maior

importância, como poderia acontecer no mundo natural. No entanto, dita colisão pode ser entendida como uma agregação matemática e, portanto, mantém um significado probabilístico (BERGH, 2001). Diversos algoritmos que visam imitar o comportamento social de agentes têm sido propostos. Estes algoritmos podem ser utilizados para simular interações sociais entre indivíduos, podendo ser aplicados para o intercâmbio de informação aplicado na resolução de problemas de otimização (MENDES, 2004; ARBOLEDA, 2012).

As propriedades principais de um sistema de inteligência de enxame são (MILLONAS, 1994):

- *Proximidade* – os agentes devem ser capazes de interagir;
- *Qualidade* – os agentes devem ser capazes de avaliar seus comportamentos;
- *Diversidade* – permite ao sistema reagir a situações inesperadas;
- *Estabilidade* – nem todas as variações ambientais devem afetar o comportamento de um agente;
- *Adaptabilidade* – capacidade de adequação a variações ambientais.

As características dos sistemas de inteligência coletiva tomam este tipo de algoritmos atrativo do ponto de vista computacional, pois, baseando-se em comportamentos simples e agentes igualmente simples, a implementação não apresenta uma complexidade elevada. Deste modo, os recursos e desempenho geral tendem, em diversos casos, a ser tão ou mais apropriados que os de outros sistemas (DORIGO, MANIEZZO, COLORNI, 1996).

Existem diversas abordagens para construção de algoritmos de Inteligência de Enxame tais como:

- Otimização por Enxame de Partículas (Particle Swarm Optimization - PSO);
- Otimização por Colônia de Formigas (Ant Colony Optimization - ACO).

A Otimização por Enxame de Partículas é um procedimento de otimização baseado no comportamento de grupos de organizações, como por exemplo, uma revoada de pássaros ou um cardume de peixes (EBERHART, KENNEDY, 2001). Soluções individuais (cada membro em uma população) são consideradas partículas que evoluem e mudam suas posições no espaço de busca conforme sua própria experiência e também na de suas vizinhas, armazenando sempre de sua melhor posição visitada e da melhor posição visitada por suas vizinhas, logo, combinando métodos de busca local e global (VELLOSO, 2008).

A Otimização por Colônia de Formigas é inspirado pelo comportamento de colônia de formigas reais, em particular, por seu comportamento de aquisição de alimentos. Uma das principais ideias é a comunicação indireta entre os indivíduos de uma colônia de agentes, chamados formigas artificiais. Esta comunicação é realizada por intermédio de trilhas de feromônio deixadas pelas formigas em componentes do problema abordado, analogamente as trilhas de feromônio depositadas por formigas reais para se comunicar. Estas trilhas artificiais são um tipo de informação numérica distribuída, que é modificada pelas formigas a fim de refletirem suas experiências acumuladas enquanto resolvendo determinado problema (ALOISE *et al.*, 2003).

2.2.4 Sistemas Imunológicos Artificiais

Os Sistemas Imunológicos Artificiais (SIA) (em inglês, *Artificial Immune Systems* (AIS)) podem ser definidos como sistemas computacionais inspirados pela teoria imunológica, na qual as funções, os princípios e os mecanismos de imunidade são utilizados para resolver problemas, apresentando características adaptativas e evolutivas assim como o Sistema Imunológico Biológico (DASGUPTA, 1998).

Os Sistemas Imunológicos Artificiais vêm sendo utilizados em diversas áreas, como reconhecimento de padrões, detecção de falhas e anomalias, segurança computacional, otimização, controle, robótica, tabulação de horários, análise de dados, aprendizagem de máquina, entre outras (BACK, FOGEL, MICHALEWICZ, 2000; TIMMIS, 2000; ALMEIDA, 2007; VIEIRA, 2009; PINA, 2010).

Nas áreas de engenharia e computação, tem surgido um forte interesse pelo estudo dos sistemas imunológicos devido, principalmente, à sua capacidade de

processamento de informação. Sob uma perspectiva de engenharia, existem diversas características do sistema imunológico (SI) que podem ser destacadas (ALMEIDA, YAMAKAMI, TAKASHI, 2006):

- **Unicidade:** cada animal possui seu próprio sistema imunológico, com suas capacidades e vulnerabilidades particulares;
- **Reconhecimento de padrões internos e externos ao sistema:** as células e moléculas que não pertencem ao organismo são reconhecidas e eliminadas pelo SI;
- **Detecção de anomalia:** o SI pode detectar e reagir a agentes patogênicos (causadores de anomalias) a que o organismo nunca havia sido exposto anteriormente;
- **Detecção imperfeita (tolerância a ruídos):** um reconhecimento perfeito não é necessário para que o SI reaja contra um elemento causador de patologia (patógeno);
- **Diversidade:** existe uma quantidade limitada de células e moléculas no SI que são utilizadas para se obter o reconhecimento de um número praticamente infinito de elementos, incluindo aqueles sintetizados em laboratório;
- **Aprendizagem por reforço:** a cada encontro com o mesmo patógeno, o sistema imunológico melhora a qualidade de sua resposta;
- **Memória:** os componentes do SI bem sucedidos no reconhecimento e combate às patologias são armazenados para uma resposta futura mais intensa e efetiva.

Assim como em outras técnicas de IC, SIA se respaldam no processamento dirigido sobre unidades de processamento (anticorpos), tal qual em RNA e neurônios artificiais; ou AG e cromossomos. Biologicamente, SIA se valem de algumas propriedades (FEYEREISL, AICKELIN, 2006):

- O sistema imunológico pode reconhecer e classificar diferentes padrões e produzir respostas seletivas. Em adição, ele usa um processo combinatório que gera um conjunto de receptores linfócitos, com a propriedade de

incrementar as chances que ao menos alguns linfócitos reconheçam um dado antígeno;

- O sistema aprende, por experiências, a estrutura de um dado antígeno. Quando células B (na Imunologia, tipo de célula que faz o reconhecimento de antígenos) são ativadas, algumas delas se tornam células de memória, com um tempo de vida estendido. Estas células ajudam o organismo a produzir uma resposta imunológica rápida quando um mesmo antígeno é encontrado no futuro. O sistema, automaticamente, determina um balanço entre economia e desempenho, mantendo um número aproximadamente suficiente destas células B;
- O mecanismo de resposta imunológica é autorregulado por natureza. Não há um órgão de controle no sistema imunológico. A regulação da resposta imunológica pode ser local ou sistêmica, dependendo do tipo de antígeno e sua localização.
- A resposta imunológica e a proliferação de células imunológicas ocorrem sob um determinado limiar de afinidade (a força que regula o acoplamento entre anticorpos e antígenos).
- O processo de expansão clonal e hipermutação somática produzem células imunológicas com alta afinidade aos antígenos invasores. Vale ressaltar que a expansão clonal e a hipermutação somática são aspectos fortemente explorados no algoritmo aiNet.

O processamento dirigido de um SIA, então, se dá por meio de metáforas implementadas algoritmicamente. Dentre alguns algoritmos, destacam-se o aiNet (*Artificial Immune NETWORK*) e o CLONALG (*CLONal selection ALGORITHM*), os quais são aplicados combinadamente em tarefas de aprendizagem de máquina, reconhecimento de padrões, compressão e *clustering*, (SILVA, 2001). O aiNet tem sua dinâmica iniciada com o fornecimento dos antígenos (padrões de treinamento) e demais parâmetros. A cada iteração (geração), na aplicação do algoritmo de seleção clonal, são produzidos os anticorpos (centróides) que se acoplam (representam) a um grupo de antígenos (dados a serem aglomerados). Disto, a cada geração, os anticorpos (que inicialmente são representações aleatórias) são refinados e/ou criados (ALMEIDA, YAMAKAMI, TAKASHI, 2006).

2.2.5 Sistemas *Fuzzy*

Sistemas *fuzzy*, baseados na lógica *fuzzy*, são sistemas capazes de trabalhar com informações imprecisas e transformá-las em uma linguagem matemática de fácil implementação computacional (FERREIRA, 2009).

A lógica *fuzzy*, também conhecida como lógica nebulosa, lógica difusa, ou ainda como teoria das possibilidades, tem como base a Teoria dos Conjuntos *Fuzzy* introduzida pelo Professor Zadeh em seu artigo "*Fuzzy Sets*" publicado no ano de 1965. Trata-se de um tipo de lógica multivalente (KANDEL, 1986). Este conceito contrapõe-se ao da lógica clássica, bivalente, fundada por Aristóteles, segundo a qual uma determinada proposição somente pode assumir os valores verdadeiro ou falso, representados respectivamente pelos números 0 e 1, excluindo-se definitivamente qualquer possibilidade de ocorrerem valores intermediários. Trata-se do Princípio do Meio Excluído (BOYER, 1996).

A lógica *fuzzy* admite, portanto que uma determinada proposição pode assumir qualquer valor no intervalo entre 0 e 1. Assim, os valores assumidos representam os graus de verdade desta proposição. Uma proposição é totalmente verdadeira se o seu grau de verdade é 1, e totalmente falsa se o seu grau de verdade é 0, admitindo-se como possível a ocorrência de quaisquer valores intermediários de graus de verdade (BOYER, 1996).

Devido à dualidade observada nos fenômenos da natureza, a lógica bivalente proporciona a criação de modelos matemáticos que em certos casos são de indiscutível utilidade. Porém há limitações quando se deseja obter modelos de fenômenos ou fatos que dependeriam da análise simultânea de muitas variáveis. A lógica *fuzzy* incorpora a forma como o ser humano realiza inferências, por meio de regras de raciocínio e atribuição de valores *fuzzy* às variáveis que observa. Valores dotados de incertezas e representados por expressões muitas vezes vagas e imprecisas.

Estas variáveis, presentes no pensamento e comunicação humanos, são consideradas válidas para a lógica *fuzzy*, haja vista que um termo linguístico pode ser representado por um conjunto *fuzzy* (SOUSA, 2005). Segundo Shaw e Simões (1999), a habilidade de se manipular conjuntos e números *fuzzy* é uma das atividades mais importantes do cérebro humano, e este processo de decisões requer

um tempo de processamento computacional mínimo por não existir modelo matemático a ser seguido.

Na lógica *fuzzy*, os valores verdade são expressos linguisticamente, (p. ex.: *verdade, muito verdade, não verdade, falso, muito falso*), onde cada termo linguístico é interpretado como um subconjunto do intervalo unitário. Estes modificadores são essenciais na geração de termos linguísticos (p. ex.: *muito alto, mais ou menos perto*). Nos sistemas lógicos clássicos existem somente os quantificadores existenciais e universais. A lógica *fuzzy* admite, em adição, uma ampla variedade de quantificadores (p. ex.: *pouco, vários, usualmente, frequentemente, em torno de cinco*). (GOMIDE, GUDWIN, 1994)

Assim, pode-se concluir que os conjuntos *fuzzy* que classificam os elementos de um dado universo são menos rígidos do que aqueles utilizados na teoria clássica visto que eles admitem graus parciais de pertinência (MARRO *et al.*, 2010).

3 MATERIAIS E MÉTODOS

3.1 COLETA DAS AMOSTRAS E AQUISIÇÃO DAS IMAGENS

A coleta de amostras (folhas) das espécies: *Plectranthus amboinicus* (Figura 4) e *Plectranthus barbatus* (Figura 5) foi realizada no horto de plantas medicinais Farmácia Viva do Grupo Centroflora, um grupo que atua no desenvolvimento e comércio de extratos vegetais para os seguimentos de cuidados pessoais, nutrição e saúde.



Figura 4. *Plectranthus amboinicus*

Fonte: Próprio Autor (2015)



Figura 5. *Plectranthus barbatus*

Fonte: Próprio Autor (2015)

Para cada uma das amostras: cinco da espécie *Plectranthus amboinicus* e cinco da espécie *Plectranthus barbatus*, foram capturadas três imagens, sendo estas obtidas por meio de câmeras digitais de três smartphones: um Sony Xperia L, com *display* de 4.3 polegadas e resolução de 854x480 *pixels*, câmera de oito *megapixels* com resolução de 3264x2448 *pixels*; um Motorola Moto G Segunda Geração, com *display* de cinco polegadas e resolução de 1280x720 *pixels*, câmera de oito *megapixels* com resolução de 3264x2448 *pixels* e um Motorola Moto Defy 525, com *display* de 3.7 polegadas e resolução de 854x480, câmera de 5 *megapixels* com resolução de 2592x1944 *pixels*. Obtendo-se um total de 30 imagens em diferentes resoluções, com maior e menor grau de detalhes, a fim de fazer com que as técnicas de PDI gerassem resultados diferentes para a mesma fotografia.

Todas as amostras foram removidas das plantas, mantendo-se a estrutura das partes a serem analisadas intactas (limbo e pecíolo). Para facilitar o processamento das imagens, a captura foi realizada sobrepondo-se as folhas em um

fundo de cor uniforme e claro, a fim de se obter um contraste entre o fundo e o objeto a ser analisado.

3.2 PROCESSAMENTO DAS IMAGENS

Após aquisição das imagens, estas passaram por um estágio de processamento, estando este dividido em: pré-processamento e extração das características.

Para a realização destas etapas foi utilizada uma biblioteca de código aberto para processamento de imagens da linguagem *Java*, nomeada de *ImageJ*, em sua versão 1.46r.

3.2.1 Pré-processamento da imagem

Primeiramente a imagem foi convertida para escala de cinza em 8 bits utilizando a função *Type*. Esta conversão é realizada *pixel a pixel* por meio da Equação 1:

$$gray = 0.299 * R + 0.587 * G + 0.114 * B$$

(1)

Os valores para R, G e B são, respectivamente, os valores das cores vermelho, verde e azul de cada *pixel*.

Após o processo de conversão para escala de cinza, a imagem é submetida ao processo de binarização, o qual consiste em atribuir valores aos *pixels* que compõem a imagem para que se possa obter uma imagem de duas cores apenas. O nível de escala de cinza de cada *pixel* está diretamente relacionado ao resultado desta binarização, onde, para cada *pixel* com o valor na escala de cinza acima do valor de um limiar, este é convertido para branco, caso contrário, convertido para preto. O valor do limiar é obtido pela análise do histograma de toda a imagem ou de uma parte dela selecionada. Como neste trabalho foram utilizadas apenas imagens no padrão RGB e os valores utilizados para escala de cinza variam de 0 a 255, o valor estabelecido para este limiar foi intermediário (50% de 256), obtendo-se o valor 128. O trecho de algoritmo abaixo descreve este processo (Equação 2).

$$color = gray > 128 ? 255 : 0$$

(2)

Como resultado deste processo de binarização, tem-se uma imagem apresentando apenas a área correspondente à folha na cor preta e o fundo na cor branca. Este resultado alcançado deve-se pela captura da imagem da folha em um fundo de tom mais claro.

Como última etapa da fase do pré-processamento, a imagem gerada pela operação de binarização é dada como entrada para a função *Find Edges*, que é responsável por gerar uma nova imagem contendo somente as bordas da folha da planta. Esta nova imagem é obtida pela aplicação do filtro de *Sobel*, que tem como função destacar regiões nas quais ocorre uma rápida variação de intensidade, usando duas matrizes 3x3 para gerar as derivadas na vertical (Matriz 1) e horizontal (Matriz 2), as quais são apresentadas abaixo.

$$\begin{vmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{vmatrix}$$

Matriz 1. Matriz 3x3 para o Sobel Vertical.

$$\begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Matriz 2. Matriz 3x3 para o Sobel Horizontal.

A Figura 6 mostra a imagem resultante de cada etapa realizada no pré-processamento.

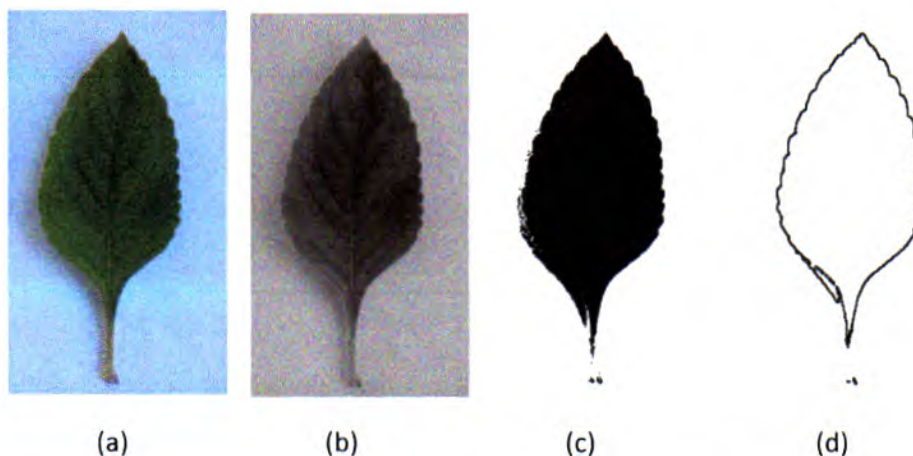


Figura 6. Etapas do pré-processamento da imagem. (a) Imagem da folha obtida; (b) imagem convertida para escala de cinza; (c) Imagem binarizada; (d) imagem obtida após a aplicação do filtro de *Sobel*

3.2.2 Extração das Características

Neste trabalho foram utilizadas como características para classificação das folhas a área, o perímetro, a largura, a altura, as cores RGB e o comprimento do pecíolo.

A área da folha foi obtida por meio da soma de todos os *pixels* na cor preta que formam a folha na imagem obtida pelo processo de binarização. O perímetro foi calculado somando-se todos os *pixels* pretos constituintes da borda da folha, que é obtida na imagem gerada pela função *Find Edges* usando o filtro de *Sobel*. As cores RGB do pecíolo podem ser obtidas por meio do histograma da área da folha correspondente ao pecíolo. A largura e altura da folha, bem como o comprimento do pecíolo podem ser obtidas por meio da função *Measure*.

Após o pré-processamento das imagens, criou-se uma tabela (Tabela 1) com os dados referentes às características morfológicas das folhas, a qual será utilizada como os parâmetros de entrada para o treinamento da Rede Neural Artificial.

Tabela 1. Dados adquiridos com o processamento de imagens

Nome da Amostra	Área	Perímetro	LarguraT	AlturaT	R	G	B	ComprimentoP
Boldo_A.jpg	1413423	6.344.779	2290	1104	127	145	92	564.125
Boldo_A.jpg	1669691	8.745.818	2549	1148	124	133	98	529.030
Boldo_A.jpg	1649294	10.291.583	2328	1224	124	150	122	620.001
Boldo_A.jpg	1595945	9.986.277	2507	1151	103	134	84	552.008
Boldo_A.jpg	1647045	8.550.823	2513	1226	125	135	88	632.000
Malva_A.jpg	1819913	7.030.623	1954	1566	102	100	103	186.000
Malva_A.jpg	2147097	7.915.005	2206	1639	107	96	111	417.011
Malva_A.jpg	2088635	9.527.714	2294	1632	134	112	119	454.004
Malva_A.jpg	2096899	8.470.541	2392	1579	114	105	87	324.000
Malva_A.jpg	1771073	7.577.109	1908	1554	115	115	104	492.000
Boldo_D.jpg	1511506	8.246.544	2793	1112	136	149	63	490.502
Boldo_D.jpg	1593823	7.677.199	2592	1138	171	191	141	495.000
Boldo_D.jpg	1537925	7.264.859	2407	1122	201	218	163	555.008
Boldo_D.jpg	1699622	8.764.673	2873	1114	98	119	38	570.000
Boldo_D.jpg	1421250	10.381.424	2760	1128	135	133	53	687.000
Malva_D.jpg	2052537	7.556.998	2071	1663	77	55	37	210.000
Malva_D.jpg	2489168	9.708.349	2380	1750	107	96	111	408.000
Malva_D.jpg	2274492	12.580.900	2380	1704	80	63	44	501.000
Malva_D.jpg	2387863	9.713.312	2584	1703	69	65	30	471.000
Malva_D.jpg	2200810	8.666.742	2179	1735	93	74	34	372.000
Boldo_R.jpg	1038778	7.324.375	2198	932	126	133	59	430.000
Boldo_R.jpg	1044610	9.297.833	2238	897	137	137	69	413.335
Boldo_R.jpg	1150591	9.397.255	2068	1062	146	144	81	502.668
Boldo_R.jpg	956916	7.082.838	2198	848	129	131	57	426.669
Boldo_R.jpg	907477	6.963.380	1958	904	109	116	73	516.000

Malva_R.jpg	1262599	6.507.653	1631	1292	69	66	66	222.002
Malva_R.jpg	1411355	7.849.710	1840	1334	64	62	65	369.000
Malva_R.jpg	1462812	8.525.790	1962	1361	106	94	61	418.669
Malva_R.jpg	1493950	10.444.916	2084	1376	111	102	60	477.333
Malva_R.jpg	1327456	8.503.344	1711	1363	68	69	38	337.000

Fonte: Próprio Autor (2015)

3.3 REDE NEURAL ARTIFICIAL

Para criar, treinar, validar e simular as RNAs, foi utilizado o software MATLAB®, na versão R2013a (8.1.0.604).

Foi implementada uma rede neural do tipo *Multi Layer Perceptron* (MLP), na qual utilizou-se como parâmetros de entrada o conjunto de dados numéricos (Área, Perímetro, LarguraT, AlturaT, R, G, B e ComprimentoP) das amostras (Tabela 1).

Para avaliar a generalização do treinamento da rede, dividiu-se o conjunto de dados em um conjunto de treinamento, validação e um de teste, selecionados aleatoriamente. A Tabela 2 mostra o número de dados utilizados em cada uma das análises.

Tabela 2. Conjunto de dados de treinamento, validação e teste.

	Nº de Experimentos	%
Treinamento	18	60
Validação	6	20
Teste	6	20
Total	30	100

Fonte: Próprio Autor (2015)

Todos os parâmetros de entrada foram normalizados no intervalo de 0,1 a 0,9. Definiu-se a arquitetura da rede como MLP do tipo *feedforward* de camadas múltiplas, 8 (oito) variáveis de entrada, 2 (duas) camadas intermediárias e 1 (uma) camada de saída. O número máximo de neurônios presentes nas camadas intermediárias foi definido conforme o teorema de Kolmogorov (KOLMOGOROV, 1957), o qual diz que uma rede de P entradas pode ser representada por $P*2+1$ neurônios na camada escondida, dessa forma, definiu-se a arquitetura da rede da seguinte forma: na primeira camada intermédia, a quantidade de neurônios variou de

2 (dois) a 16 (dezesesseis) $(8*2+1)$ e na segunda, de 0 (zero) a 8 (oito) neurônios $((8*2+1)/2)$.

3.4 IMPLEMENTAÇÃO DO APLICATIVO

A aplicação foi desenvolvida por meio da IDE (*Integrated Development Environment*) *Eclipse Java EE (Enterprise Edition)*, na sua versão *Juno*, a qual possui um pacote de ferramentas para desenvolvedores *Java*, possibilitando aos mesmos a criação de diversas aplicações móveis, *desktop* e *Web*.

Para a extração de características das folhas, utilizou-se uma biblioteca desenvolvida na plataforma *Java* para processamento e análise de imagens, denominada *ImageJ*, na versão 1.46r.

Para acessar os recursos do celular como a câmera, utilizou-se o *Android SDK (Software Development Kit)* que consiste em uma ferramenta para desenvolver diversas aplicações. O *Android SDK* permite que os desenvolvedores elaborem as aplicações (jogos e utilitários) a partir de um dispositivo virtual para os aparelhos de celular e *tablet*, a fim de que façam uso das funções oferecidas pelos aparelhos, como *touchscreen*, telefonia GSM, Câmera, GPS, bússola, acelerômetro, *Bluetooth*, EDGE, 3G e WiFi.

Foi utilizado o *Android Development Tools (ADT)*, um plugin para a IDE *Eclipse*, que possibilitam vários recursos, tais como: configuração mais veloz de novos projetos *Android*, criação de uma interface de usuário para o aplicativo, adição de pacotes com base na *API Framework Android*, depuração dos aplicativos usando o SDK do *Android* e até a capacidade de exportar APK assinadas (ou não) para que a aplicação seja distribuída.

Vale ressaltar que para o correto funcionamento do aplicativo, é preciso ter presentes no computador o *Eclipse* (versão 3.6.2 ou superior), o *Java* 1.6 (ou superior) e o *Android SDK*. Dessa forma, primeiramente, instalou-se o JDK, a fim de rodar os aplicativos *Java*, nos quais o *Android* trabalha. Após a instalação do JDK, instalou-se o *Eclipse*, que foi o software utilizado para programação da aplicação. Em seguida, fora instalado o SDK do *Android*. E por final, o ADT *plugin*.

4 RESULTADOS E DISCUSSÕES

4.1 PROCESSAMENTO DIGITAL DAS IMAGENS

Os resultados do Processamento Digital de Imagem realizado neste trabalho para o reconhecimento e classificação das folhas das espécies *Plectranthus amboinicus* (boldo) e *Plectranthus barbatus* (malva) mostraram-se eficientes para a obtenção das métricas da área, perímetro, largura, altura, cores RGB e comprimento do pecíolo.

O pressuposto de que as imagens fotografadas por dispositivos diferentes resultariam em parâmetros distintos, foi confirmado, quando constatou-se que, embora os dispositivos possuam câmeras de mesma resolução, as imagens produzidas pelos mesmos eram formadas por matrizes de *pixels* distintos, como pode ser observado na Tabela 3. Contribuindo-se dessa forma, com o enriquecimento do conjunto de dados formulados para cada uma das amostras utilizadas.

Tabela 3. Relação entre a resolução e a matriz de pixels dos dispositivos utilizados para realizar a captura das imagens.

Dispositivo	Resolução	Matriz de Pixels
Motorola® Defy MB525	5 Megapixel	2592 x 1456
SONY® Xperia L	8 Megapixel	1746 x 3104
Motorola® MotoG g2	8 Megapixel	1836 x 3264

Fonte: Próprio Autor (2015)

4.2 REDE NEURAL ARTIFICIAL

A capacidade de generalização da RNA possibilitou o estudo da importância de cada um dos parâmetros utilizados para a classificação correta das folhas em questão neste trabalho. Com a aplicação da técnica de validação cruzada obteve-se a melhor configuração para a rede neural, ou seja, a detectada com o maior índice de acerto, a qual possui topologia 3:6:1 (3 neurônios na primeira camada escondida, 6 na segunda e 1 na camada de saída) conforme vista na Figura 7. Esta rede apresentou um grau de acerto de 90,92% em sua fase de treinamento,

medido pela correlação de diferença entre resultados calculados pela RNA e resultados reais.

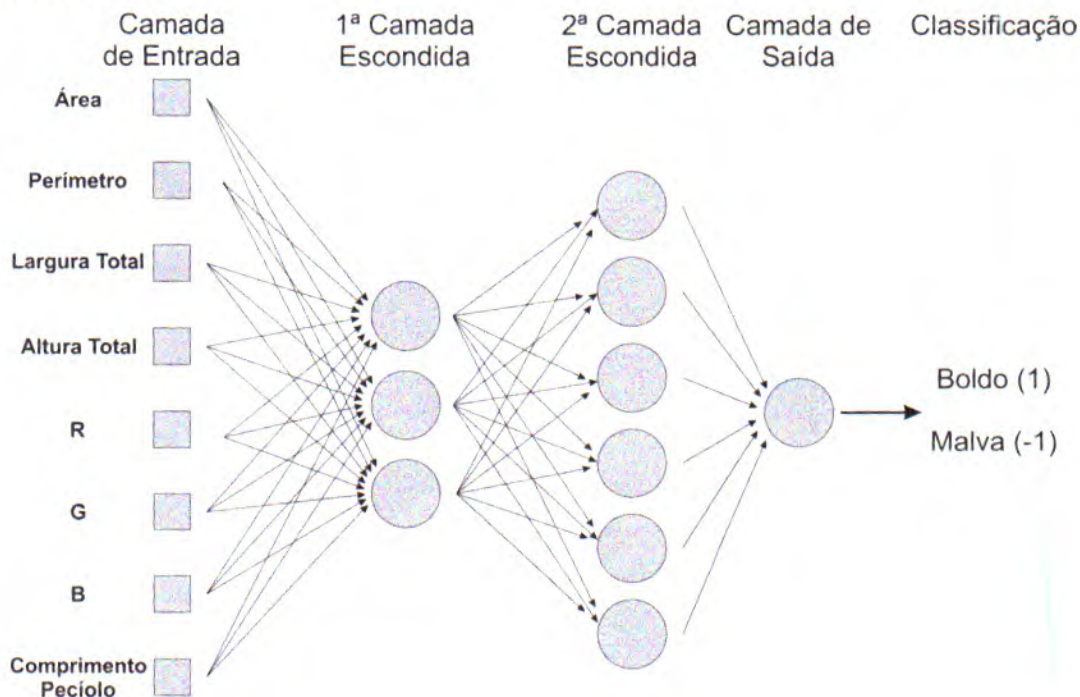


Figura 7. Representação da melhor configuração da RNA encontrada pelo processo de validação cruzada

Fonte: Próprio Autor (2015)

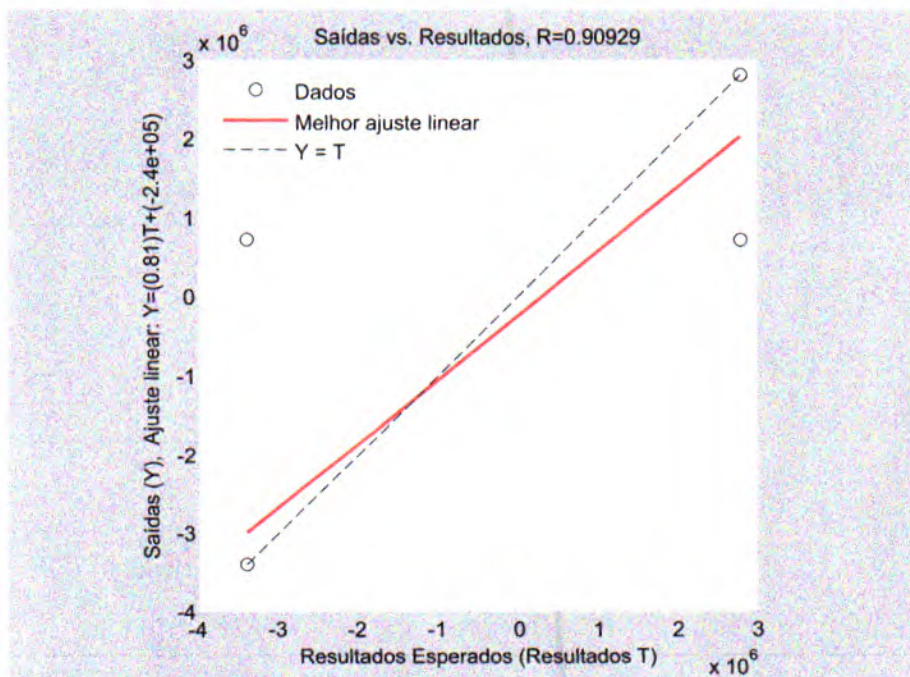


Figura 8. Gráfico resultante do conjunto de treinamento, realizado na rede neural MLP

Fonte: Próprio Autor (2015)

Utilizou-se no total 30 (trinta) amostras como parâmetros de entrada para a rede, das quais 60% foram selecionadas aleatoriamente para compor o conjunto de treinamento.

A Figura 8 mostra a correlação entre os resultados obtidos e as saídas desejadas para a RNA de menor erro quadrático médio encontrado.

Para o conjunto de treinamento, acreditou-se que o alcance dos resultados ocorreu devido à quantidade de parâmetros de entrada inseridos na rede. A capacidade de generalização da rede na classificação de padrões possibilitou um acerto de 96,15 % durante a fase de testes da rede, conforme a Figura 9.

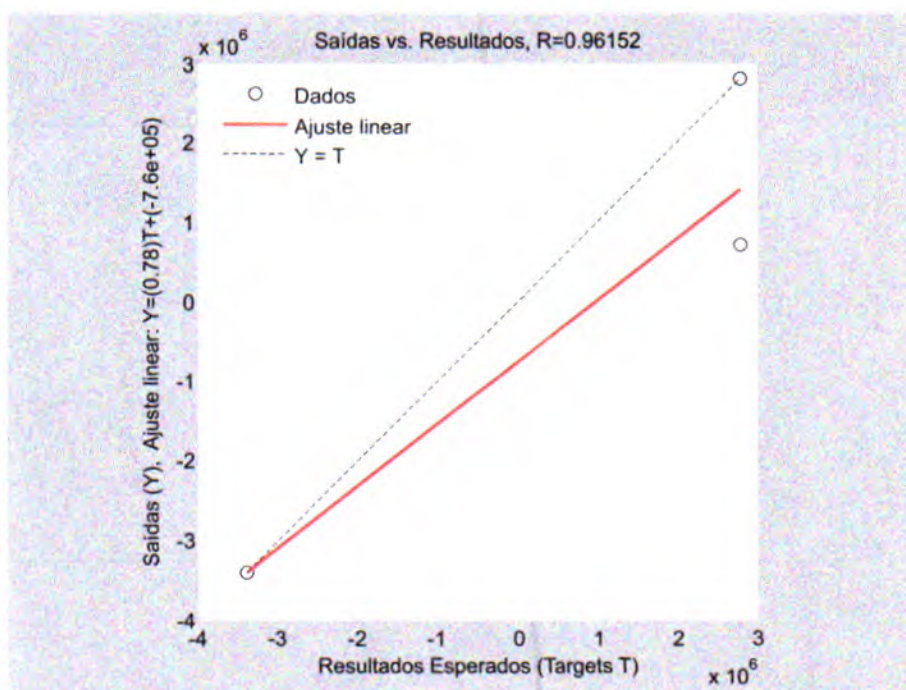


Figura 9. Gráfico de comparação entre os novos dados inseridos com as saídas desejadas, sob o conjunto de teste

Fonte: Próprio Autor (2015)

Uma das métricas utilizadas para se avaliar o desempenho da rede neste trabalho foi o erro médio absoluto percentual (MAPE), que indica quão próximo os dados preditos pela rede são dos dados reais. Em termos de MAPE, os resultados obtidos foram 31,62; 90,92 e 96,15%, para os conjuntos de validação, de treinamento e de teste, respectivamente (Figura 10).

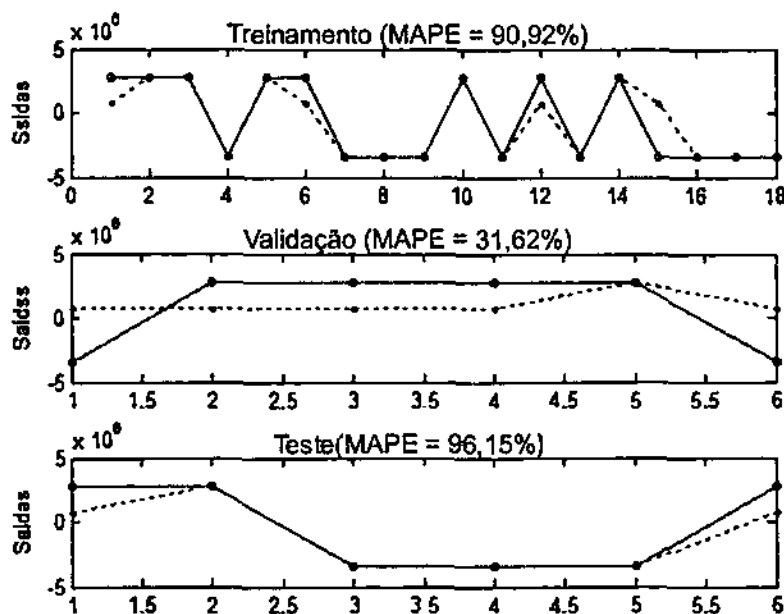


Figura 10. Relação dos resultados obtidos durante as fases de treinamento, validação e teste. Saídas desejadas (○) Saídas previstas pela RNA (●)

Fonte: Próprio Autor (2015)

Analisando-se os gráficos da Figura 10, conclui-se a eficiência nos resultados fornecidos pela RNA encontrada, tanto durante a fase de treinamento quanto na fase de teste, isso significa dizer que a rede conseguiu obter uma boa generalização do problema em classificar as duas espécies *Plectranthus amboinicus* (boldo) e *Plectranthus barbatus* (malva)

4.3 APLICAÇÃO INTELIGENTE

A RNA treinada neste trabalho foi utilizada para desenvolver uma aplicação inteligente para dispositivos móveis utilizando-se a IDE *Eclipse Java EE*, na versão *Juno* e a biblioteca *ImageJ*, em conjunto com o *Android SDK*, bem como o *ADT plugin*.

A Figura 11 mostra a interface do usuário do aplicativo desenvolvido neste trabalho para realizar o reconhecimento das plantas por meio de suas folhas.

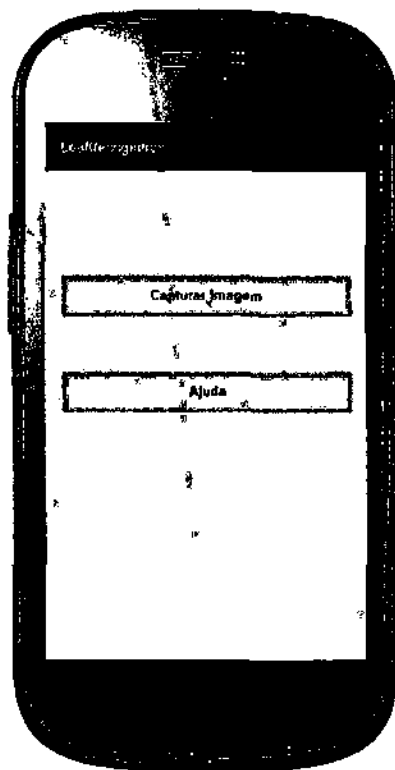


Figura 11. Interface de entrada do aplicativo desenvolvido para realizar o reconhecimento das folhas

Fonte: Próprio Autor (2015)

Ao abrir a tela inicial do aplicativo, o usuário deve clicar no botão "Capturar Imagem" para dá início ao processo de reconhecimento da folha ou, caso esteja com dúvidas em relação à forma de utilização do aplicativo, deve clicar no botão "Ajuda" para obter informações à respeito do mesmo. Quando o botão "Capturar Imagem" é pressionado, uma nova tela é mostrada ao usuário, na qual o usuário deve posicionar a folha para que uma imagem da mesma seja capturada (Figura 12).

Após a captura da folha, uma outra tela é mostrada ao usuário (Figura 13), a qual permite que o mesmo possa visualizar o resultado desta captura, além de fornecer opções como: "Analisar", que permite a usuário obter dados à cerca da folha capturada.



Figura 12. Tela de Captura da folha

Fonte: Próprio Autor (2015)



Figura 13. Resultado do processo de captura da imagem da folha

Fonte: Próprio Autor (2015)

A Figura 14 mostra a janela do resultado obtido utilizando o aplicativo proposto neste trabalho. Pode-se observar que o sistema classificou corretamente a folha a ser reconhecida, apresentando o nome popular da planta, seu nome científico, a imagem de uma outra folha da mesma classe, bem como informações de como se utilizar esta planta no tratamento de doenças.



Figura 14. Resultado do processamento realizado pelo aplicativo implementado

Fonte: Próprio Autor (2015)

5 CONSIDERAÇÕES FINAIS

Mostrou-se neste trabalho o uso de técnicas de Reconhecimento de Padrões para extração de características de dois tipos de folhas da família *Plectranthus*, coletadas no horto de plantas medicinais Farmácia Viva do Grupo Centroflora. Essas características extraídas foram utilizadas para compor uma base de dados, a qual serviu como parâmetros de entrada para o treinamento de uma Rede Neural Artificial (RNA).

Foi implementada uma RNA do tipo *Multi Layer Perceptron* para classificação das espécies *Plectranthus amboinicus* (boldo) e *Plectranthus barbatus* (malva), utilizando o algoritmo *backpropagation*. A generalização dos dados da RNA possibilitou o desenvolvimento de uma ferramenta capaz de calcular os valores dos parâmetros que compõem as características que definem morfologicamente as folhas baseando-se nos parâmetros reais calculados.

A RNA treinada foi empregada no desenvolvimento de um aplicativo inteligente para dispositivos móveis para classificação de plantas medicinais a partir de sua folha. Os resultados obtidos foram bastante satisfatórios, todavia, poderão ser melhorados utilizando-se um conjunto maior de dados, bem como utilizando-se outras características para classificação das folhas, como por exemplo, utilizar fractais para extrair sua nervura e verificar se existe um padrão na formação desta ou utilizar métodos capazes de identificar a forma de sua borda ou o ângulo formado entre o ápice e a base presentes na folha, ou até mesmo utilizar outras técnicas de Reconhecimento de Padrões.

A inovação gerada pelo presente trabalho consiste na elaboração de um produto (software) capaz de utilizar uma técnica de Inteligência Computacional (Redes Neurais Artificiais) em seu processamento e, com isso, determinar o reconhecimento de determinado conjunto de padrões.

Pode-se a partir deste trabalho, elaborar outros sistemas para aplicativos móveis utilizando técnica similar ou agregando outras técnicas através de sistema híbridos como: redes neurais *fuzzy* ou algoritmos genéticos de treinamento de RNAs.

REFERÊNCIAS

ALMEIDA, C. P. **Aplicação de Sistemas Imunológicos Artificiais para a Predição da Estrutura de Proteínas**. Dissertação de M.Sc., UTFPR, Curitiba, 2007.

ALMEIDA, T. A.; YAMAKAMI, A.; TAKASHI, M. T. **Sistema imunológico artificial para resolver o problema da árvore geradora mínima com parâmetros fuzzy**. Unicamp, 2006.

ALOISE, D.; NORONHA, T. F.; MAIA, R. S.; BITTENCOURT, V. G.; ALOISE, D. J. **Heurísticas de colônia de formigas com pathrelinking para o problema de otimização da alocação de sondas de produção terrestre**. Simpósio Brasileiro de Pesquisa Operacional. Rio de Janeiro, 2003.

ARBOLEDA, D. M. M. **Otimização por inteligência de enxames usando arquiteturas paralelas para aplicações embarcadas**. Tese (Doutor em Mecatrônica) – Universidade Federal de Brasília, Brasília, 2012.

BACK, T.; FOGEL, D. B.; MICHALEWICZ, Z. **Evolutionary Computation Basic Algorithms and Operators**. *Institute of Physics Publishing*, Bristol, UK, 2000.

BADKE, M. R. **Conhecimento popular sobre o uso de plantas medicinais e o cuidado de enfermagem**. Dissertação (Mestrado em Enfermagem) – Universidade Federal de Santa Maria, Santa Maria, 2008.

BERGH, F. **An Analysis of Particle Swarm Optimizers**. Tese (Doutorado) – Department of Computer Science, University of Pretoria, South Africa, 2001.

BEZDEK, J. C., PAL, S. K. **Fuzzy Models Pattern Recognition: Methods That Search for Structures in Data**, *IEEE*, 1992.

BISHOP, C. M. **Neural Networks for Pattern Recognition**. Oxford University Press, 1995.

BONIFACIO, F. N. **Comparação entre as Redes Neurais Artificiais MLP, RBF e LVQ na Classificação de Dados**. Dissertação (Mestre em Ciência da Computação) – Universidade Estadual do Oeste do Paraná, Cascavel, 2010.

BORGES, T. H. **Sistema de classificação de plantas por meio de suas folhas usando uma arquitetura híbrida composta por algoritmos genéticos e rede neural artificial**. Dissertação (Mestrado em Ciências) – Universidade Federal de Uberlândia, Uberlândia, 2013.

BOYER, C. B. *História da Matemática*, São Paulo, Editora Edgar Blücher, 1996.

CARNEIRO, M. L. **Programação genética aplicada à programação de controladores lógico programáveis**. Dissertação (Mestre em Engenharia Elétrica e de Computação) – Universidade Federal de Goiás, Goiânia, 2009.

CASTRO, L.; ZUBEN, F. J. V. **Introdução à Computação Evolutiva**. Unicamp, 2001.

COSTA, J. P. O.; ORTOLAN, L. D. **Classificação de imagens de sensoramento remoto utilizando redes neurais artificiais**. Trabalho de Conclusão de Curso (Bacharel em Sistemas de Informação) – Universidade Federal do Paraná, Curitiba, 2014.

DASGUPTA, D. **An Overview of Artificial Immune Systems and Their Applications**. *Artificial Immune Systems and Their Applications*. Springer-Verlag, Berlin, Germany, 1998.

DORIGO, M.; MANIEZZO, V.; COLORNI, A. **Ant system: Optimization by a colony of cooperating agents**. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, [S.l.], v.26, n.1, 1996.

DUDA, R. O., HART, P. E. **Pattern classification and scene analysis**. *John Wiley & Sons, Inc.*, 1973.

DUDA, R. O., HART, P. E., STORK, D.G., **Pattern Classification**, 2nd edition, U.S.A., *John Wiley & Sons*, 2001.

EBERHART, R., Y. S.; KENNEDY, J. **Swarm Intelligence**. *Morgan Kaufmann*, San Francisco, 2001.

ENGELBRECHT, A. P. **Computational Intelligence: An Introduction**. 2nd Edition. Chichester: *John Wiley*, 2007. 628 p.

ESMIN, A. A. A. **Estudo de Aplicação do Algoritmo de Otimização por Enxame de Partícula na Resolução de Problemas de Otimização Ligados ao SEP**. Itajubá, 2005.

FERREIRA, L. **Aplicação de sistemas fuzzy e neuro-fuzzy para predição da temperatura retal de frangos de corte**. Dissertação (Mestre em Engenharia de Sistemas) – Universidade Federal de Lavras, Minas Gerais, 2009.

FEYEREISL, J.; AICKELIN, U. **Artificial Immune Tissue using Self-Organizing Networks**, *Proceedings of the Workshop on Artificial Immune Systems and Immune System Modelling*, p. 5-6, 2006.

FU, L. M. **Rule Generation from Neural Networks**. *IEEE Transactions on Systems: Man and Cybernetics*, v. 28, n. 8. p. 1114 -1124, 1994.

FUKUNAGA, K., **Introduction to Statistical Pattern Recognition**, 2nd edition, U.S.A., *Academic Press*, 1990.

GABRIEL, P. H. R.; DELBEM, A. C. B. **Fundamentos de Algoritmos Evolutivos**. 2008.

GOMIDE, F. A.; GUDWIN, R. R. **Modelagem, controle, sistemas e lógica fuzzy**. *SBA – Controle & Automação*, v. 4, n. 3, 1994.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. 2nd edition, The MIT Press, 1992.

JESUS, N. Z. T.; LIMA, J. C. S.; SILVA, R. M.; ESPINOSA, M. M.; MARTINS, D. T. O. Levantamento etnobotânico de plantas popularmente utilizadas como antiúlcera e antiinflamatórias pela comunidade de Pirizal, Nossa Senhora do Livramento – MT, Brasil. *Revista Brasileira de Farmacognosia*. v.19, n.1A, p.130-9, 2009.

JUNIOR, J. N. R. S. **Sintonia ótima de regulador automático de tensão e estabilizador de sistema de potência utilizando algoritmo de otimização por enxame de partículas**. Dissertação (Mestre em Engenharia Elétrica) – Universidade Federal do Ceará, Fortaleza, 2012.

KANDEL, A. *Fuzzy Mathematical Techniques with Applications*, U.S.A., Addison-Wesley Publishing Company, 1986.

KASABOV, K. N. **Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering**, *Massachusetts Institute of Technology Press*, USA, 1996.

KLIR, G. J., YUAN, B. **Fuzzy sets and fuzzy logic - theory and applications**. *Prentice Hall*, 1995.

KOLMOGOROV, A. N. On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition. *Doklady Akademii Nauk*, v. 114, p. 679-681, 1957.

KRESS, W. J., WURDACK, K. J., ZIMMER, E. A., WEIGT, L. A.; JANZEN, D. H. **Use of DNA barcodes to identify flowering plants**. *PNAS*. v.102, p. 8369-8374, 2005.

LIBERMAN, F. **Classificação de Imagens Digitais por Texturas usando Redes Neurais**. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 1997.

LUKE, S. **Two fast tree-creation algorithms for genetic programming**. *IEEE Transactions in Evolutionary Computation*, v. 4, p. 274-283, 2000.

MARRO, A. A.; SOUZA, A. M. C.; CAVALCANTE, E. R. S.; BEZERRA, G. S.; NUNES, R. O. **Lógica Fuzzy: Conceitos e aplicações**. Universidade Federal do Rio Grande do Norte, Natal, 2010.

MENDES, R. **Population Topologies and Their Influence in PSO**. Tese (Doutorado) – Universidade do Minho, Portugal, 2004.

MILLONAS, M. M. (1994). **Swarms, phase transitions, and collective intelligence**. In C.G. Langton (Ed.), *Artificial Life III*, pp. 417–445. Reading, MA: Addison Wesley.

MOTA, K. Instalando e atualizando o plugin ADT no Eclipse. Disponível em: <<http://www.klebermota.eti.br/2010/02/08/installing-and-updating-adt/>>. Acesso em: 10/06/2015.

MOUSSI, L. N.; GUDWIN, R. R.; ZUBEN, F. J. V.; MADRID, M. K. **Classificadores com redes neurais (nnns) : aplicação ao controle de um veículo autônomo simulado computacionalmente**. São Paulo, 2000.

PAO, Y. **Adaptive Pattern recognition and neural networks**. Addison-Wesley Publishing Company, 1989.

PEREIRA, A. P. **Android Development Tools**. Disponível em: <<http://www.baixaki.com.br/download/androiddevelopmenttools.htm#ixzz3faaLN0J7>>. Acesso em: 10/06/2015.

PINA, A. A. **Metodologias de análise, síntese e otimização de sistemas para produção de petróleo offshore através de metamodelos e enxame de partículas**. Tese (Doutor em Engenharia Civil) – Instituto Alberto Luiz Coimbra, Rio de Janeiro, 2010.

PRADO, A. A. M. C. P. P. L. **Algoritmos para reconhecimento de padrões**. *Ciênc. Exatas*, Taubaté, v. 5-8, p. 129-145, 2002.

SANCHES, R. O. Criando ambiente Android no Eclipse. Disponível em: <<http://www.devmedia.com.br/criando-ambiente-android-no-eclipse/24621>>. Acesso em: 10/06/2015.

SANTOS, E. J. D. **Aplicação de Redes Neurais Artificiais na análise do encapsulamento de bobinas de transformadores**. Dissertação (Mestrado em Engenharia Elétrica) – Universidade Estadual de Londrina, Londrina, 2013.

SCHALKOFF, R. J. **Pattern Recognition: Statistical, Structural and Neural Approaches**. John Wiley & Sons, Inc., 1992.

SCHIAVONI, A. S. **Um estudo comparativo de métodos para balanceamento do conjunto de treinamento em aprendizado de redes neurais artificiais**. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Lavras, Minas Gerais, 2010.

SERAPIÃO, A. B. S. **Fundamentos de Otimização por Inteligência de Enxames: Uma Visão Geral**. *Revista Controle & Automação*, Porto Alegre, v. 20, n. 3, p. 271-304, 2009.

SHAOBO, L.; GUANCI, Y.; QINGSHENG, X. **Automatic design method of dynamic systems based on Hungarian algorithm and genetic programming**. International conference on wireless communications, networking and mobile computing, *IEEE*, Guiyang, China, 2008.

SHAW, I. S.; SIMÕES, M. G. **Controle e Modelagem Fuzzy**, São Paulo, Editora Edgar Blücher, 1999.

SILVA, A. M. Utilização de redes neurais artificiais para classificação de spam. Dissertação (Mestre em Modelagem Matemática e Computacional) – Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte, 2009.

SILVA, I. N., SPATTI, D. H. E FLAUZINO, R. A. Redes Neurais Artificiais para engenharia e ciências aplicadas. São Paulo: Artliber, 2010.

SILVA, J. M. N. Redes Neurais Artificiais: Rede Hopfield e Redes Estocásticas. Dissertação (Mestre em Ciência da Computação) – Universidade Federal Fluminense, Rio de Janeiro, 2003.

SILVA, P. V. S.; NASCIMENTO, G. N. L. Investigação etnobotânica das plantas medicinais cultivadas nas hortas comunitárias de Palmas – TO. PIVIC – UFT, Tocantins, 2013.

SILVA, R. R. Reconhecimento de padrões de defeitos de soldagem em radiografias industriais. Tese (Doutor em Ciências em Engenharia Metalúrgica e de Materiais) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2003.

SILVA, R. R. Reconhecimento de imagens digitais utilizando redes neurais artificiais. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Lavras, Minas Gerais, 2005.

SMITH, J. Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression, Doctor thesis, 1998.

SOBRINHO, C. Uma Análise das Aplicações dos Algoritmos Genéticos em Sistemas de Acesso à Informação Personalizada, 2003.

SOUZA, J. A. Reconhecimento de padrões usando indexação recursiva. Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Santa Catarina, Florianópolis, 1999.

SOUSA, J. N. P. Aplicação de lógica fuzzy em sistemas de controle de tráfego metropolitano em rodovias dotadas de faixas exclusivas para ônibus. Dissertação (Mestre em Ciências em Engenharia de Transportes) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2005.

TAFNER, M. A. Estagiamento Automático do Sono usando uma Rede Neural Artificial com Alta Representação Cortical. Exame de Qualificação - Departamento de Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 1998.

TARALLO, A. DE S. Segmentação e Classificação de Imagens Digitais de Úlceras Cutâneas Através de Redes Neurais Artificiais. [s.l.] Universidade de São Paulo, 2007.

TIMMIS, J. Artificial Immune Systems: A Novel Data Analysis Technique Inspired by the Immune Network Theory. Tese de D.Sc., University of Whales,

Department of Computer Science, Aberystwyth, Ceredigion, Wales, 2000.

TOU, J. T., GONZALEZ, R. C. **Pattern Recognition Principles**. Addison-Wesley Publishing Company, Massachusetts, 1981.

VARGAS, P. A. **Sistemas Classificadores para Redução de Perdas em Redes de Distribuição de Energia Elétrica**, *Dissertação de Mestrado*, Faculdade de Engenharia Elétrica e de Computação, 2000.

VELLOSO, B. P. **Proposta de uma metodologia baseada na teoria da percolação para configuração dos parâmetros de um algoritmo de inteligência de enxames**. *Dissertação (Mestre em Ciência da Computação)* – Universidade Federal de Santa Catarina, Florianópolis, 2008.

VIEIRA, I. N. **Algoritmos Bio-Inspirados Aplicados a Otimização de Risers Rígidos Em Catenaria**. *Dissertação de M.Sc.*, COPPE/UFRJ, Rio de Janeiro, 2009.

WIJNS, C.; BOSCHETTI, F.; MORESI, L. **Inverse modelling in geology by interactive evolutionary computation**. *J. Struct. Geol.* v. 25, n. 10, p 1615 1621, EUA, 2003.

YAMASHITA, C. I. **Estudo sobre os constituintes inorgânicos presentes em diferentes espécies da planta medicinal do gênero *Casearia* coletadas em regiões distintas da Mata Atlântica, SP**. *Dissertação (Mestrado em Tecnologia Nuclear)* – Instituto de Pesquisas Energéticas e Nucleares, São Paulo, 2006.

ZUBEN, F. J. V. **Computação Evolutiva: Uma Abordagem Pragmática**. Unicamp, 2001.