

UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

DANIELE VIEIRA MINATTO

**ANÁLISE DE CARACTERÍSTICAS EM REDES NEURAS ARTIFICIAIS
APLICADAS EM VISÃO COMPUTACIONAL COM ÊNFASE EM
RECONHECIMENTO FACIAL**

CRICIÚMA

2016

DANIELE VIERIA MINATTO

**ANÁLISE DE CARACTERÍSTICAS EM REDES NEURAIS ARTIFICIAIS
APLICADAS EM VISÃO COMPUTACIONAL COM ÊNFASE EM
RECONHECIMENTO FACIAL**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. Esp. Sergio Coral

Coorientadora: Prof.^a. Dr.^a. Merisandra Côrtes de Mattos Garcia

CRICIÚMA

2016

DANIELE VIEIRA MINATTO


**ANÁLISE DE CARACTERÍSTICAS EM REDES NEURAIS ARTIFICIAIS
APLICADAS EM VISÃO COMPUTACIONAL COM ÊNFASE EM
RECONHECIMENTO FACIAL**

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Automação.

Criciúma, 20 de junho de 2016.

BANCA EXAMINADORA


Prof. Sergio Coral – Especialista – Esp. – Orientador


Profª. Merisandra Côrtes de Mattos Garcia – Dra. - (UNESC) - Coorientadora


Prof. Evânio Ramos Nicoleit – MEng. - (UNESC)


Prof. Gustavo Bisognin – MSc. - (UNESC)

**Para meus pais que estiveram junto
comigo durante esta caminhada.**

AGRADECIMENTOS

Primeiramente, gostaria de agradecer as duas pessoas mais importantes da minha vida que sem as quais eu não seria nada: Ademir Minatto e Denir Vieira Minatto. Mãe, Pai vocês são os pilares da minha vida, me deram sonhos e me ensinaram a lutar por eles, a cada obstáculo que a vida me lançava vocês estavam lá por mim, não tenho palavras para agradecer por tudo que vocês fizeram, e fazem até hoje por mim.

Ao meu irmão Diego pelo apoio e suporte nos momentos difíceis.

Vitor, obrigada por ser esse namorado amoroso e compreensivo, que não mediu esforços para me incentivar a lutar quando eu pensava em desistir.

A todos os amigos que fiz durante o curso, vocês foram essenciais para a minha formação.

Ao meu orientador Sergio Coral por dar o suporte necessário para fazer este projeto acontecer, a minha co-orientadora Merisandra C. de Matos Garcia por partilhar seu conhecimento e experiência. Ao professor Kristian Madeira pela contribuição e orientação acerca dos métodos estatísticos aplicados nesta pesquisa.

E por fim, a todos que contribuíram direta ou indiretamente para minha formação, meu muito obrigada.

“Não considere nenhuma prática como imutável. Mude e esteja pronto a mudar novamente. Não aceite verdade eterna. Experimente.”

B. F. Skinner

RESUMO

Tecnologias de reconhecimento estão cada vez mais ganhando espaço no mercado tanto nas áreas de saúde com aplicações de reconhecimento dos mais variados tipos de exames, como em segurança com vigilância por vídeo e autenticação bancária. Aliada a essas tecnologias, as características biométricas surgem como um identificador viável, único e intransferível. Tais características são utilizadas atualmente no Brasil para autenticação das veias das palmas das mãos e da digital, em sua maioria aplicada em urnas eletrônicas e caixas bancários, entretanto uma grande parte dos Estados Unidos e União Europeia já utilizam características da face como identificadores de identidade em controle de fronteiras e vigilância. Com a criação de novas técnicas pesquisas sobre tecnologia de reconhecimento facial vem avançando muito e cada vez mais gerando produtos comerciais, principalmente aplicados em segurança. Entretanto alguns problemas ainda são relatados como: qualidade da imagem, luminosidade, oclusão e orientação. Esta pesquisa apresenta um estudo baseado em um dos algoritmos mais populares para reconhecimento de características, o algoritmo de Viola-Jones, e a inclusão de uma rede neural artificial a fim de melhorar as taxas de reconhecimento em ambientes onde a luminosidade é variada. Para a consolidação dos conceitos estudados foi montado um protótipo e desenvolvido um *software* de reconhecimento facial. Visando comprovar a eficiência do *software* foram recolhidos os resultados da aplicação executando o algoritmo de Viola-Jones e os resultados do mesmo algoritmo anexado a uma rede neural artificial. A solução desenvolvida é um protótipo, sendo que para aplicação em ambientes reais é necessário um estudo de *hardware* mais aprofundado, bem como submeter o *software* desenvolvido a um tempo maior de treinamento.

Palavras-chave: Biometria. Reconhecimento Facial. Viola-Jones. Visão Computacional. Redes Neurais Artificiais.

ABSTRACT

Recognition technologies are increasingly gaining market share both in health applications of various types of tests, such as in security with video surveillance and banking authentication. Coupled with these technologies, biometric characteristics emerge as a viable identifier, unique and nontransferable. These features are currently used in Brazil for authentication of the veins of the palms and digital, mostly applied to electronic voting machines and tellers, though a large part of the United States and European Union already use face characteristics as identity identifiers border control and surveillance. With the creation of new technical research on facial recognition technology has advanced a lot and increasingly generating commercial products, mainly applied safely. However some problems are still reported as image quality, brightness, occlusion and guidance. This study describes a study based on a popular algorithms for recognizing characteristics, Viola-Jones algorithm, and including an artificial neural network to improve recognition rates in environments where the lighting is varied. To consolidate the concepts studied was assembled a prototype and developed a facial recognition software. Aiming to prove the efficiency of the software was collected the results of applying running Viola-Jones algorithm and the results of this algorithm attached to an artificial neural network. The solution developed is a prototype, and for application in real environments need further study hardware and software developed submit to a longer training.

Keywords: Biometrics. Facial Recognition. Viola-Jones. Computer Vision. Artificial Neural Networks.

LISTA DE ILUSTRAÇÕES

2 BIOMETRIA	20
Figura 1 - Esquema de funcionamento de um reconhecedor de retina	22
Figura 2 – Leitor de geometria da mão	23
Figura 3 - Processamento da imagem digital	23
Figura 4 - Íris humana	24
Figura 5 - Reconhecimento da Face	27
Figura 6 - Etapas do reconhecimento facial	29
Figura 7 - Mapeamento da face	30
Figura 8 - Imagem em tons de cinza	35
Figura 9 - Realce em uma imagem em escala de cinza.....	36
Figura 10 - Aplicação do filtro Gaussian Blur em imagens em escala de cinza	37
Figura 11 - Segmentação em escala de cinza	38
Figura 12 - Exemplo de <i>features</i> que compoe a imagem.....	40
Figura 13 - Pseudocódigo do Adaboost	42
Figura 14 - Características seleccionadas pelo algoritmo AdaBoost	43
Figura 15 - Diagrama de funcionamento do <i>harr-cascades</i>	44
Figura 16 - Exemplo de rede MLP.....	46
Figura 17 - Modelo de Aprendizagem Supervisionada.....	47
Figura 18 - Modelo da propagação do erro do algoritmo <i>backpropagation</i>	48
Figura 19 - Pseudocódigo <i>Backpropagation</i>	50
Figura 20 - Raspberry PI 2 Modelo B	58
Figura 21 - Protótipo montado.....	60
Figura 22 - Compilação do <i>framework</i> OpenCV	60
Figura 23 - Base de Testes Positivas e Negativas	62
Figura 24 - Etapas do Sistema	63
Figura 25 - Função de transformação de imagens.....	65
Figura 26 - Carregamento do <i>haar-cascade</i>	66
Figura 27 - Função principal do algoritmo Viola-Jones	67
Figura 28 - Estrutura de detecção dos olhos.....	68
Figura 29 - Exemplo da imagem de <i>.log</i>	70
Figura 30 - Gráfico da Curva de Aprendizado	70
Figura 31 - Criação da RNA com o <i>framework</i> OpenCV	72

Figura 32 - Parâmetros da RNA.....	73
Figura 33 - Treinamento da RNA	74
Figura 34 - Tendência de treinamento da RNA	76
Figura 35 – Amostra de Faces reconhecidas	77

LISTA DE TABELAS

Tabela 1 - Modalidades Biométricas	21
Tabela 2 - Aplicações biométricas.....	26
Tabela 3 – Arquiteturas utilizadas na construção da topologia da RNA	72
Tabela 4 - Matriz de Confusão de Viola-Jones.....	77
Tabela 5 - Acurácia de sistemas semelhantes	79

LISTA DE ABREVIATURAS E SIGLAS

CPF	Cadastro de Pessoa Física
CYMK	Canal de cores Ciano, Amarelo, Magenta e Preto
DNA	Ácido Desoxirribonucleico
FP	Falso Positivo
FN	Falso Negativo
MLP	Multilayer Perceptron
OPENCV	Open Source Computer Vision
PC	Computador Pessoal
PCA	Análise de Componentes Principais
RG	Registro Geral
RGB	Canal de cores Vermelho, Verde e Preto
ROC	Receiver Operator Characteristic Curve
RNA	Redes Neurais Artificiais
SO	Sistema Operacional
VP	Verdadeiro Positivo
VN	Verdadeiro Negativo

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVO GERAL	15
1.2 OBJETIVOS ESPECÍFICOS	15
1.3 JUSTIFICATIVA	15
1.4 ESTRUTURA DO TRABALHO	18
2 BIOMETRIA	20
2.1 RECONHECIMENTO FACIAL	25
3 VISÃO COMPUTACIONAL	31
3.1 PROCESSAMENTO DE IMAGENS	33
3.1.1 Escala de tons de cinza	34
3.1.2 Realce de imagem	35
3.1.3 Remoção de Ruído	36
3.1.4 Segmentação	37
3.2 ALGORITMO DE VIOLA-JONES	38
3.3 REDES NEURAS ARTIFICIAIS	45
3.3.1 Aprendizagem Supervisionada	47
3.3.1.1 Algoritmo <i>Backpropagation</i>	48
4 TRABALHOS CORRELATOS	51
4.1 BIOID – APLICAÇÃO EXPERIMENTAL DE IDENTIFICAÇÃO BIOMÉTRICA BASEADA EM IMPRESSÃO DIGITAL	51
4.2 RECONHECIMENTO FACIAL SIMULTÂNEO USANDO O ALGORITMO DE VIOLA-JONES E REDES NEURAS ARTIFICIAIS PARA VERIFICAÇÃO DE IDENTIDADE	52
4.3 SISTEMA DE RECONHECIMENTO FACIAL	52
4.4 RECONHECIMENTO DE FACES UTILIZANDO REDES NEURAS ARTIFICIAIS COM FUSÃO DE CARACTERÍSTICAS FACIAIS POR MÉTODO DE VOTAÇÃO ...	53
4.5 RECONHECIMENTO DE FACES UTILIZANDO REDES NEURAS E BIOMETRIA	54
4.6 ACELERAÇÃO DE DETECÇÃO DE FACES EM TEMPO REAL COM UM ROBÔ DE TELEPRESENÇA UTILIZANDO UM RASPBERRY PI	55

5 ANÁLISE DE CARACTERÍSTICAS EM REDES NEURAS ARTIFICIAIS APLICADAS EM VISÃO COMPUTACIONAL COM ÊNFASE EM RECONHECIMENTO FACIAL.....	57
5.1 COMPONENTES DO PROTÓTIPO.....	57
5.1.1 Raspberry Pi.....	57
5.1.2 Framework OpenCV.....	58
5.1.3 Montagem do Protótipo.....	59
5.2 BASE DE TESTES.....	61
5.3 DESENVOLVIMENTO DO PROTÓTIPO DE SOFTWARE DE ANÁLISE DE CARACTERÍSTICAS EM RECONHECIMENTO FACIAL APLICANDO REDES NEURAS ARTIFICIAIS E O ALGORITMO DE VIOLA-JONES.....	62
5.3.1 Preparação da Imagem.....	64
5.3.2 Desenvolvimento do algoritmo de Viola-Jones.....	66
5.3.3 Aplicação e desenvolvimento da rede neural artificial MLP.....	71
5.3 ANÁLISE DOS RESULTADOS OBTIDOS.....	76
7 CONCLUSÃO.....	80
REFERÊNCIAS.....	82

1 INTRODUÇÃO

Em janeiro de 2016 foram registradas 150.643 tentativas de fraude conhecidas como roubo de identidade, de acordo com o Indicador Serasa Experian (2016), este número é equivalente a uma tentativa de fraude a cada 17,8 segundos no país.

No ano de 2015, segundo a Serasa Experian (2016), estimou-se em 1.944.200 o total de tentativas de roubo de dados pessoais, número equivalente a 16,3 tentativas, para agravar mais este quadro, sistemas de empresas e governos sofreram constantes ataques. Diante deste panorama, pode-se concluir que a necessidade de pesquisas e investimentos em tecnologias capazes de identificar indivíduos é imprescindível para erradicar o roubo de identidade e fraudes, bem como reconhecer indivíduos potencialmente perigosos ou suspeitos, trazendo assim mais segurança à população de forma geral (CA TECHNOLOGIES, 2015).

Dos cinco sentidos: visão, audição, olfato, paladar e tato, a visão é sem dúvida, a que o homem tem maior dependência, sendo a que fornece a maior parte de dados que ele recebe, segundo Davies (2012, tradução nossa) cada um dos olhos chega a fornecer taxas de dados para visualização contínua, em uma estimativa superior a 10 megabits por segundo. Outra característica do sistema visual humano é a facilidade com a qual a interpretação é realizada. Observa-se uma cena qualquer na rua e sem a necessidade de dedução ou esforço aparente ela é rapidamente processada e identificada fornecendo uma interpretação dentro de um décimo de segundo. Ao longo da história as maiores mentes da humanidade: cientistas, filantropos, educadores, políticos, líderes e filósofos, ficaram fascinados em descobrir como o cérebro humano funciona. De Michelangelo para Lomonosov, de DaVinci a Einstein, tem-se inúmeras tentativas de decifrar a forma com que a mente humana trabalha para replicá-la, inicialmente em dispositivos mecânicos simplórios para, mais tarde, no século 20 por meio de máquinas e *softwares* (MONWAR; GAVRILOVA, 2013, tradução nossa).

A identidade é representada por um conjunto de informações que podem ser relacionadas a um indivíduo e que descrevem suas características e

singularidades. Neste contexto, uma identidade é a informação relativa a pessoa, e não a esta fisicamente. Uma identidade é constituída de vários elementos como: nome, endereço, preferências, status social, financeiro e características físicas do indivíduo. Partindo deste princípio, a biometria começou a ser estudada como uma das alternativas mais seguras no ramo de autenticação de indivíduos (SMART CARD ALLIANCE, 2006).

O conceito de usar um sistema biométrico para fins de verificação da identidade não é novo. Os antigos egípcios, sumérios e outros povos da antiguidade praticavam tais técnicas, entretanto apenas na década de 90 ocorreu um crescimento no mercado biométrico e conseqüentemente um maior estudo, dando início a criação de padrões e novas tecnologias na área (ASHBOURN, 2014, tradução nossa).

São muitos e variados os exemplos de informação biométrica, sendo divididos em dois grandes grupos de acordo com suas características: fisiológicas e comportamentais. Dentre as modalidades de biometria, estão inclusas: reconhecimento facial, vocal, leitura da retina, geometria da mão, análise do DNA, odor do indivíduo, leitura das veias da palma ou costas da mão, geometria dos dedos, entre outros, sendo que, o mais popular atualmente é a leitura de impressão digital (ASHBOURN, 2014, tradução nossa). Por ser um dos métodos mais baratos, a identificação pela leitura da digital vem sendo empregada na área de segurança e amplamente estudada, tendo registros de sua utilização desde o século XIX. Segundo Jain, Prabhakar e Pankanti (2002) as impressões digitais são formadas geralmente no sétimo mês de gestação e permanecem intactas por toda a vida, se não houver um ferimento ou corte profundo, possuindo uma chance em cem bilhões de existirem duas pessoas com a mesma digital, além de que, apenas cerca de 2% da população mundial possuem digitais ilegíveis. Entretanto, é a modalidade biométrica que apresenta mais vulnerabilidade quanto a fraudes, pois para um sistema é muito difícil distinguir entre um dedo verdadeiro e uma réplica artificial contendo uma impressão digital, outro ponto grave, é a presença de cortes ou machucados impossibilitarem a extração de algumas características (MODI, 2011, tradução nossa)

Segundo Gavrilova e Monwar (2013, tradução nossa) a face é o identificador biométrico mais utilizado para realizar a autenticação do ser humano. Ele

é usado todos os dias como o principal meio para o reconhecimento de outros seres humanos. Tornou-se objeto de estudo da área de visão computacional, área esta de interesse de uma ampla parcela da sociedade científica, pois além do ser humano já possuir a capacidade inata de reconhecer faces, independente das variações que a mesma apresenta, a face é um elemento altamente deformável cuja aparência pode mudar em diferentes poses, expressões e iluminações (NAPPI; TISTARELLI; MARISCO, 2014, tradução nossa).

Este tipo de aplicação requer um alto nível de sensibilidade referente a detecção e comparação de características biométricas, principalmente em lugares onde a face é obtida dinamicamente em ambientes não controlados (VIGLIAZZI, 2006). Para Davies (2012) um meio óbvio de enfrentar este tipo de problema é padronizar as imagens de alguma forma. No entanto, na prática, considerando-se a variação natural de padrões, incluindo os efeitos de luminosidade, ruído e distorção (ou até mesmo os efeitos de quebras ou oclusões), um dos problemas mais relatados é sobre a generalização sobre os padrões.

Localização visual e rastreamento de objetos, particularmente o de faces humanas são uma tarefa crítica e um campo ativo de pesquisas e aplicações em visão computacional. Considerando que a face é um elemento disforme e depende de algumas variáveis para sua localização como a iluminação de imagens e o fundo a qual elas aparecem, pode-se obter alguma dificuldade em detectá-las (DELIGIANNIDIS; ARABNIA, 2014, tradução nossa).

Inúmeros pesquisadores tentaram de alguma forma amenizar estes problemas dando origem a inúmeros trabalhos como os de Sung e Poggio (1998) que propuseram um detector de faces a base de distribuição, Rowley et al (1998) relataram um detector de faces com base em redes neurais artificiais inspirados em Féraud et al (1997) que desenvolveu um detector baseado em um novo modelo de rede neural artificial que foi nomeado de Constrained Generative Model (CGM). Entre todos estes trabalhos, o que atualmente está se destacando é o algoritmo desenvolvido por Viola e Jones (2001) que propôs o primeiro método de detecção de faces em tempo real, utilizando um conjunto de características simples em vez de extrair informações dos pixels da imagem.

Estes avanços permitem incorporar técnicas de Inteligência Computacional a estes algoritmos, aprimorando o processo de reconhecimento, embora nenhuma máquina, até o momento tenha sido capaz de imitar o comportamento de reconhecimento no mesmo nível dos seres humanos (ASHBOURN, 2014, tradução nossa). A adição de algoritmos de aprendizagem as redes neurais artificiais, permite que a máquina aprenda mais rapidamente e com um baixo custo computacional. Segundo Vigliuzzi (2006) para utilizar sistemas automatizados de identificação biométrica, torna-se necessário o uso de sistemas de inteligência computacional, mais precisamente de redes neurais artificiais.

A arquitetura de uma rede neural artificial restringe o tipo de problema no qual a rede poderá ser utilizada, sendo definida pelo número de camadas (camada única ou múltiplas camadas), pelo número de nós em cada camada, pelo tipo de conexão entre os nós e por sua topologia (HAYKIN, 2001). Técnicas de Inteligência computacional como redes neurais artificiais podem agregar robustez ao processo de reconhecimento (BARRETO, 2002). Idealmente, a rede se torna mais instruída sobre o seu ambiente após cada iteração do processo de aprendizagem (HAYKIN, 2001).

Sistemas de rede neural artificial aplicados em visão computacional empregam algoritmos para determinar a semelhança das características globais únicas contra faces de referência, usando o máximo de área da imagem possível. Sistemas neurais são projetados para aprender quais recursos são mais eficazes dentro do sistema, para combinar e ajustar-se de forma pragmática, com base nos indicadores que se mostram mais eficazes. Este método, teoricamente, leva a um aumento da capacidade de identificar as faces em condições difíceis (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

Partindo desta premissa esta pesquisa propõe o desenvolvimento de um protótipo de *software* de reconhecimento facial implementando o algoritmo de Viola-Jones, que pode detectar e processar imagens de forma rápida e com um baixo custo computacional, fazendo o uso de uma rede neural artificial do tipo Multilayer Perceptron (MLP) utilizando o algoritmo *backpropagation* para aprimorar o desempenho da detecção em condições difíceis.

1.1 OBJETIVO GERAL

Esta pesquisa tem como objetivo desenvolver um protótipo de reconhecimento facial, fazendo uso do algoritmo de Viola-Jones juntamente com uma rede neural artificial fazendo uso de um Raspberry Pi.

1.2 OBJETIVOS ESPECÍFICOS

A presente pesquisa contém os seguintes objetivos específicos:

- a) compreender os conceitos que englobam o reconhecimento facial e as formas de processamento de imagens digitais;
- b) identificar os principais pontos de extração de características faciais, por meio do algoritmo de Viola-Jones;
- c) aplicar o modelo de rede neural artificial MLP e o algoritmo de aprendizagem de *backpropagation* por meio do *framework* OpenCV;
- d) analisar os resultados do reconhecimento facial com o algoritmo de Viola-Jones e com a integração da rede neural artificial por meio da técnica de visualização de resultados.

1.3 JUSTIFICATIVA

O procedimento de identificação facial é algo natural para o ser humano, pois todos os dias, reconhecem pessoas, aprende-se características novas e armazenam-se novas faces no cérebro. Procedimentos estes que são executados rapidamente e parecem ser uma tarefa trivial. No entanto, compreender o processamento do cérebro humano vem sendo um dos principais desafios para os pesquisadores a fim de criar métodos de reconhecimento, utilizando uma junção de inúmeras características biométricas, técnica esta que foi nomeada como biometria multimodal, porém para que este método seja desenvolvido com eficácia é necessário

que cada método que o compõe seja otimizado de maneira particular (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

Sistemas de reconhecimento facial que empregam algoritmos como os de Viola-Jones para determinar a semelhança de amostras fornecidas em relação a características pré armazenadas, usam o máximo da área facial informada, desta forma, os sistemas de redes neurais artificiais aprendem quais recursos são mais eficazes para combinar e se ajustar de forma pragmática, levando a um aumento da capacidade de identificar as faces em condições difíceis (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa). Utilizando um sistema de votação as redes neurais artificiais do tipo MLP fazem a triagem final, comparando as características informadas com a base de dados, modificando os pesos/valores de cada característica a fim de determinar se existe uma correspondência (HAYKIN, 2001).

Um sistema de identificação biométrica é capaz de reconhecer padrões que identificam uma pessoa por meio da determinação da autenticidade de uma característica específica (GAVRILOVA; MONWAR, 2013, tradução nossa). Atualmente a biometria é usada para dezenas de aplicações, no Brasil a modalidade biométrica mais difundida é o reconhecimento de digitais, ela está presente na autenticação na urna eletrônica, em bancos e no ponto dos trabalhadores. Já o reconhecimento facial, vem ganhando espaço no cotidiano da população, em aplicações como desbloqueio de celulares, identificação de faces em câmeras fotográficas digitais e até mesmo em câmeras de vídeo. Em países como os Estados Unidos e União Europeia, os sistemas biométricos, mais precisamente os de reconhecimento facial, têm sido implantados com sucesso em uma série de aplicações no mundo real oferecendo um desempenho razoavelmente bom nos setores de Perícia Forense, Civil e Comercial, em agências do Governo, estudos sobre Genética, e nas demais áreas da Saúde.

As vantagens de reconhecimento facial incluem a aceitação pública da modalidade, sensores comumente disponíveis, a natureza não fisicamente intrusiva e a facilidade com que os seres humanos podem verificar os resultados do sistema de segurança baseado em biometria facial (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa). Esta é a única modalidade biométrica capaz de operar sem a

cooperação do utilizador, pois, pode usar como base imagens estáticas, como fotos de carteira de motorista e utilizar o equipamento de aquisição de imagem existente, sem a necessidade de equipamentos mais sofisticados.

Para Gonzales e Woods (2000) uma das principais metas de análise de imagens por computador é dotar uma máquina com a capacidade de aproximar-se, em um determinado sentido, da capacidade dos seres humanos. Porém, este processo é complexo, apresentando-se como uma tarefa difícil, pois não se sabe ainda como o cérebro humano desempenha esta função.

Féraud et al (1997) desenvolveu o Constrained Generative Model (CGM) um detector baseado no rosto. CGMS são conectados a MLP de uma rede neural auto associativa que são treinados para executar uma Análise de Características Principais do inglês Principal Component Analysis (PCA) não-linear. Sung e Poggio (1998) propuseram um detector de faces à base de distribuição. Este método particiona a distribuição do rosto de seis grupos em seis *clusters* no subespaço para analisar as PCA, e classificar faces e não-faces com base em distâncias que são computadas entre os pontos demarcados do grupo candidato. Rowley et al (1998) relataram um detector de face com base em redes neurais artificiais. Um conjunto de redes neurais artificiais funcionam nos pixels de regiões candidatas, cada rede tem diferentes sub redes com conexões para calcular as relações espaciais de características faciais e, em seguida, mesclar, a partir de redes individuais para um classificador final. Yang et al (2000) propôs um detector de faces baseado no método Sparse Networks of Windows (SNoW). Tal método é definido como uma única camada de rede neural que usa a regras de atualização para partes definidas da imagem. A aprendizagem da rede neural SNoW utiliza recursos booleanos que codificam a intensidade e posição dos pixels.

No início de 2001 Paul Viola e Michael Jones apresentaram sua estrutura para detecção de objetos e desde aquela época, o quadro não se alterou significativamente. Sendo este modelo amplamente utilizado em uma variedade de aplicações de *software* e *hardware* que incorporam elementos da visão computacional, como o módulo de detecção de rosto em videoconferência, interação humano-computador e câmeras fotográficas digitais (HWU, 2011).

Segundo Araújo (2010), o método de Viola-Jones é capaz de detectar faces com precisão e com alta taxa de acerto. Embora os métodos de Féraud et al (1997), Rowley et al (1998), Sung e Poggio (1998) e Yang et al (2000) tenham apresentado bons resultados quanto ao reconhecimento, mesmo gerando falso-positivos, todos os métodos acima, sem exceção, precisam de um tempo maior para processar uma imagem, enquanto o de Viola e Jones apresenta resultados muito bons, quando comparados aos demais, tanto em termos de precisão quanto de velocidade.

1.4 ESTRUTURA DO TRABALHO

Esta pesquisa é dividida em seis capítulos onde são abordados os principais conceitos e metodologias a serem utilizadas. No primeiro capítulo é realizada uma introdução identificando as principais razões para a criação de soluções baseadas em reconhecimento facial, fazendo uma breve apresentação de técnicas de reconhecimento facial, como o algoritmo de Viola-Jones e redes neurais artificiais e os benefícios de aplicar um hibridismo entre as duas técnicas.

No segundo capítulo é feita uma introdução sobre a biometria, seus principais tipos, vantagens e desvantagens, dando ênfase na modalidade de reconhecimento facial e exemplificando aplicações reais aplicando o conceito de biometria utilizando técnicas de visão computacional.

No terceiro capítulo são apresentados os conceitos de Visão Computacional, neste capítulo é dado abordado os recursos necessários e principais métodos para o desenvolvimento de um sistema dotado de visão. Ainda neste capítulo são abordadas técnicas para extração de características e classificação de imagens.

O quarto capítulo se destina a apresentação dos trabalhos correlatos voltadas a abordagens relativas a modalidade de reconhecimento facial, técnicas biométricas e protótipos feitos aplicando estes mesmos conceitos.

No quinto capítulo é apresentado o protótipo desenvolvido, todas as etapas de montagem e o desenvolvimento do *software* utilizado para o reconhecimento de faces. Em seguida é feita a coleta de dados referente a efetividade do detector e feita

uma comparação entre o reconhecimento através do algoritmo de Viola-Jones e o algoritmo de Viola-Jones atrelado a uma rede neural artificial.

E por fim, no sexto capítulo são apresentadas as conclusões relativas ao protótipo desenvolvido, feito um fechamento dos assuntos aqui apresentados e sugeridos temas para a continuação desta pesquisa.

2 BIOMETRIA

Hegg e Luongo (1975) descrevem a biometria como a ciência que estuda as características mensuráveis dos seres vivos, ou em um sentido generalizado que interpreta fenômenos vitais. A etimologia da palavra biometria pode ser encontrada nas palavras gregas *bios* que significa vida e *metron* que significa medida. Tecnologias biométricas são baseadas em características fisiológicas e comportamentais as quais são únicas para o ser humano, tais como: impressões digitais, estrutura da face, padrões de íris e tom de voz (MODI, 2011, tradução nossa).

Na atualidade, em termos de infraestrutura digital interage-se com um número crescente de sistemas, tanto no mundo físico como no virtual. O processo de identificação de indivíduos e controle de acesso baseado em privilégios vem se tornando progressivamente complexo com o passar do tempo (MODI, 2011, tradução nossa). Técnicas de reconhecimento tradicionais são baseadas no que se sabe, tal como senhas, ou algo que se possui, como por exemplo cartões de identificação, entretanto estes métodos podem ser esquecidos, roubados, usados clandestinamente ou até mesmo extraviados (MODI, 2011, tradução nossa).

Dado ao aumento da sensibilização para questões de segurança em empresas e aplicações voltadas ao cliente, a necessidade de estratégias de segurança eficazes tem crescido amplamente (NANAVATI; THIEME; NANAVATHI, 2002, tradução nossa). Para Ashbourn (2014, tradução nossa) devido a essa necessidade, criou-se um mercado significativo para a aplicação de tecnologia biométrica nas áreas de segurança.

Aplicações biométricas podem ser divididas em dois tipos principais: sistemas de verificação e sistemas de identificação. Sistemas de verificação respondem à pergunta "Sou quem reivindico ser?" exigindo que o usuário informe uma representação de identidade e forneça uma amostra biométrica para iniciar a busca por esta combinação na base de dados, a resposta fornecida por este tipo de sistemas é um booleano, aceitando ou não um resultado. Já sistemas de identificação respondem à pergunta "Quem sou eu?" solicitando ao usuário apenas uma amostra biométrica para dar início a comparação de dados. A resposta fornecida por este tipo

de sistema é uma representação de identidade (NANAVATI; THIEME; NANAVATHI, 2002, tradução nossa). Seja qual for a técnica empregada, o princípio fundamental da verificação de identidade biométrica é combinar uma amostra com outra, geralmente, uma amostra viva contra uma referência armazenada (ASHBOURN, 2014, tradução nossa).

Ao longo de um século de pesquisa biométrica, tem-se desenvolvido tecnologias com vários parâmetros físicos e comportamentais. Reconhecimento de impressões digitais, reconhecimento facial, da íris e de voz tem um maior nível de conscientização do público devido aos aplicativos voltados ao consumidor, além da cobertura da mídia, filmes e livros, enquanto há outros como reconhecimento de veias e dinâmica de digitação que são relativamente desconhecidos. De modo geral, as características físicas proporcionam uma leitura mais consistente já que são minimamente afetadas pelo comportamento do indivíduo e são consideradas mais precisas que a biometria comportamental (MODI, 2011, tradução nossa). Estas características além de existirem em todos os indivíduos, obedecendo o princípio da universalidade, devem satisfazer alguns dos critérios descritos na tabela 1 para serem aceitas como características biométricas.

Tabela 1 - Modalidades Biométricas

Modalidade	Exclusividade	Precisão	Permanência	Desempenho	Aceitação
Impressão Digital	●	●		●	●
Retina	●	●			
Íris	●	●	●		
Veias da Mão	●		●		●
Fala				●	●
Face	●	●		●	●
Assinatura		●			●

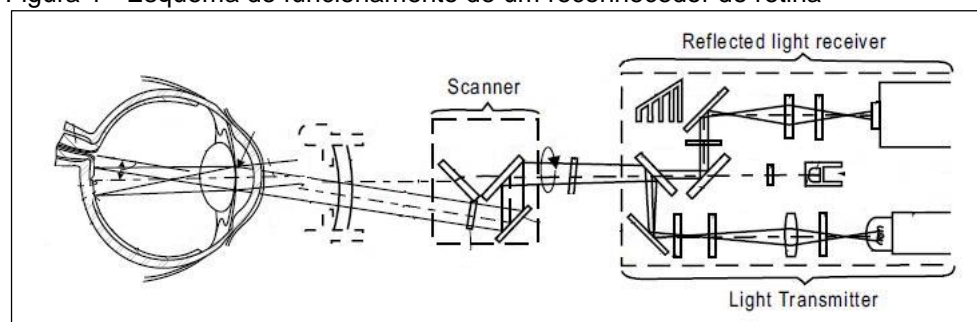
Fonte: do Autor.

Todas as tecnologias apresentadas na tabela são unimodais, o que significa que elas usam uma única característica de reconhecimento. A capacidade de fundir várias tecnologias biométricas para melhorar o desempenho e eliminar os pontos fracos levou à concepção de sistemas multimodais. Estes sistemas combinam

várias características, ou uma variedade de outras fontes de informação. Para Modi (2011, tradução nossa) as expectativas do que tecnologias uni modais biométricas podem conseguir, vem crescendo com o aumento da sua utilização. Não existe melhor tecnologia biométrica, pois o domínio da biometria é extremamente dinâmico, com várias novas tecnologias surgindo a cada momento.

Uma das primeiras implementações de verificação de identidade biométrica foi a digitalização de retina. Essa verificação consistia em envolver a retina com um feixe de luz infravermelha a fim de expor o padrão das veias na parte de trás que foi considerada única para o indivíduo (figura 1). Na opinião de Ashbourn (2014, tradução nossa) esta não era uma forma intuitiva para verificar a identidade, a fim de obter acesso a uma instalação ou outro benefício. Além disso, os usuários de óculos estavam em desvantagem já que seria necessário remover seus óculos, a fim de interagir com o dispositivo de digitalização de retina. Esta intrusão não tolerada pelos usuários, unido ao preço relativamente elevado dos dispositivos originais, assegurou que a digitalização da retina não tivesse um mercado imediato fora de aplicações específicas.

Figura 1 - Esquema de funcionamento de um reconhecedor de retina



Fonte: Jain, Bolle e Pankanti (1999).

Outra técnica biométrica que foi desenvolvida inicialmente foi a geometria da mão. Os primeiros dispositivos obtinham uma taxa de acerto razoavelmente boa, mas não eram muito práticos e na perspectiva do usuário, eram muito grandes e pesados (figura 2). No entanto, com o passar do tempo o modelo do dispositivo foi rapidamente refinado (VIGLIAZZI, 2006). Ainda hoje os leitores de geometria da mão são usados em locais como aeroportos para aplicações de controle de acesso físico,

sendo para Ashbourn (2014, tradução nossa), um bom exemplo de técnica que provou ser adaptável por meio de uma variedade de aplicações e permanece em uso até hoje.

Figura 2 – Leitor de geometria da mão



Fonte: ATSolutions (2014)

Já o reconhecimento de impressões digitais utiliza o padrão encontrado na pele dos dedos para identificar indivíduos, é a tecnologia biométrica mais antiga e mais amplamente adotada e, como resultado, é a mais madura de todas as tecnologias biométricas. A impressão digital é rica em detalhes e possui vários métodos diferentes para a captura conforme demonstradas na figura 3 (MODI, 2011, tradução nossa). Para Ashbourn (2014, tradução nossa) esta modalidade biométrica representa um exemplo interessante de uma tecnologia que continua com evolução do produto dentro de um mercado complexo e variado.

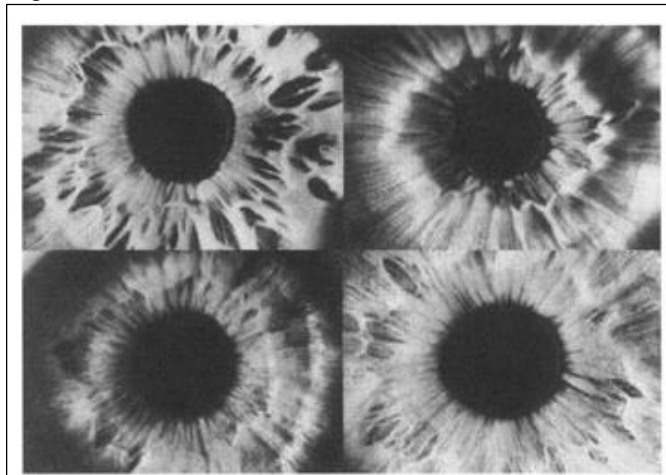
Figura 3 - Processamento da imagem digital



Fonte: Nanavati, Thieme e Nanavati (2002).

Na tecnologia de reconhecimento de íris, são utilizadas as características distintivas da íris humana (figura 4), a fim de identificar ou verificar a identidade dos indivíduos, possuem o potencial de desempenhar um grande papel no mercado biométrico se os sistemas do mundo real e suas soluções atendessem a promessa teórica da tecnologia (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa). Embora outras tecnologias sejam capazes de identificação de alta precisão, a íris é a única característica que não muda ao longo da vida de uma pessoa (MODI, 2011, tradução nossa)

Figura 4 - Íris humana



Fonte: Nanavati, Thieme e Nanavati (2002).

Os sistemas de reconhecimento de assinatura, também conhecidos como sistemas de verificação da dinâmica de assinatura, ao contrário do que se imagina, não analisa apenas o formato das letras. Esse sistema captura características como a pressão da caneta, velocidade, a identificação dos movimentos e os pontos em que a caneta é levantada do papel. São essas as características que autenticam a assinatura de uma pessoa (VIGLIAZZI, 2006). É uma técnica bastante utilizada por instituições financeiras que aplicam sistemas de reconhecimento de assinatura para verificar a autenticidade dos cheques emitidos pelos seus clientes (VIGLIAZZI, 2006).

O Reconhecimento da fala é uma tecnologia que sempre pareceu promissora e ainda tem lutado para atingir uma massa crítica na sua utilização (ASHBOURN, 2014, tradução nossa). A tecnologia de reconhecimento da fala utiliza os aspectos distintos da voz para verificar a identidade dos indivíduos, combinando

elementos da biometria comportamental e fisiológica, ela é atrelada a uma chave, ou seja, é dependente de uma frase ou palavra particular cujo usuário deve falar para ser reconhecido, pois o sistema não consegue verificar um usuário com trechos aleatórios de frases.

De todos os métodos apresentados o reconhecimento facial, é o único que se aproxima da forma como seres humanos se reconhecem, utilizando características distintas do rosto humano, a fim de verificar ou identificar os indivíduos. No mercado biométrico, o reconhecimento facial vem sendo aplicado em inúmeras áreas, sendo utilizado em conjunto com sistemas de cartões, aplicações de identificação e vários tipos de operações de vigilância (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa). Nesta pesquisa optou-se por se aprofundar na área de reconhecimento facial.

2.1 RECONHECIMENTO FACIAL

O reconhecimento facial automatizado foi desenvolvido pela primeira vez na década de 1960. Durante 1964 e 1965 Woodrow Wilson Bledsoe, Helen Chan Wolf e Charles Bisson trabalharam no reconhecimento facial humano e desenvolveram o primeiro sistema semiautomático de reconhecimento (TRASLAVIÑA, 2014).

Na década de 70 Goldstein, Harmon e Lesk, usaram 21 características físicas como a cor do cabelo e a espessura dos lábios para automatizar o reconhecimento facial, entretanto identificar estas características ainda requeria um processo manual. Ao final dos anos 80 eles produziram um ponto de referência quando Kirby e Sirovich aplicaram uma técnica padronizada de álgebra linear, a PCA (TURK; PENTALAND, 1991). Esta tecnologia chamou muito a atenção do público quando em 2001, durante o SuperBowl da Liga Nacional de Futebol Americano, foram capturadas imagens de vigilância e comparadas com uma base de dados de fotos arquivadas digitalmente. Desta demonstração originou-se um importante debate sobre como usar a tecnologia para satisfazer necessidades, principalmente governamentais, levando em consideração as preocupações sociais e de privacidade das pessoas. Atualmente a tecnologia de reconhecimento facial está sendo muito utilizada para combater, por exemplo, as fraudes de passaportes, a identificação de crianças

desaparecidas, em minimizar as fraudes nas identificações e controle de acesso, entre outras (NUNES, 2015).

Em alguns aspectos o reconhecimento facial pode ser considerado uma tecnologia fácil de implementar. A maioria das câmeras, incluindo a de notebooks, celulares e outros dispositivos portáteis são capazes de capturar uma imagem facial nítida. Com a popularização da tecnologia, aparelhos de captura de imagem têm se tornando relativamente baratos. Entretanto para o reconhecimento facial trabalhar de forma precisa e confiável, podendo localizar e diferenciar indivíduos, um certo rigor é necessário na concepção e implantação destes sistemas (ASHBOURN, 2014, tradução nossa).

Sistemas de reconhecimento facial podem se apresentar nas mais variadas soluções, baseadas apenas em *softwares* que processam imagens adquiridas através de circuitos fechados de televisão, câmeras para sistemas de aquisição e processamento, incluindo câmeras e estações de trabalho. Em alguns sistemas de reconhecimento facial, a tecnologia de base é otimizada para trabalhar com câmeras específicas e dispositivos de aquisição (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

A tabela 2 apresenta algumas áreas e aplicações específicas com reconhecimento facial:

Tabela 2 - Aplicações biométricas

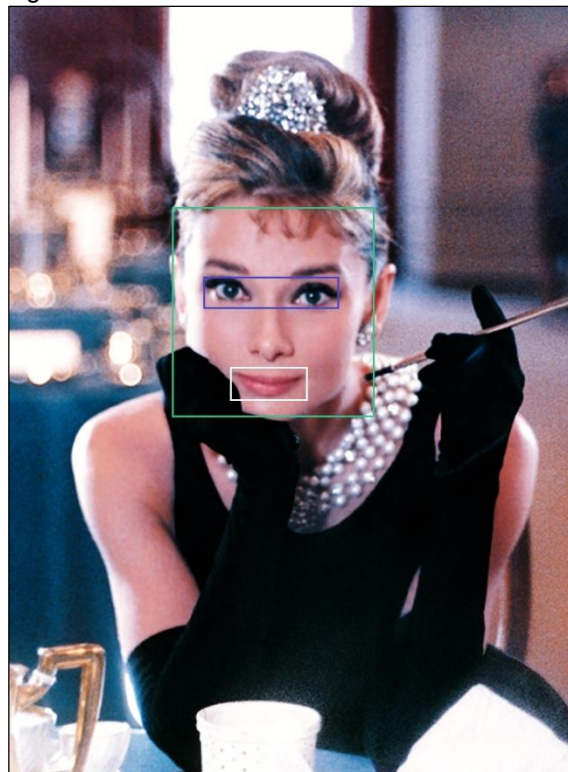
Área	Aplicações
Segurança da Informação	<ul style="list-style-type: none"> ✓ Carteira de habilitação ✓ Passaportes ✓ Controle de Fronteiras ✓ Registro Eleitoral ✓ Segurança de Aplicações.
Vigilância e Segurança	<ul style="list-style-type: none"> ✓ Vídeo Vigilância ✓ Caixas Eletrônicas ✓ Acompanhamento e Investigação de Suspeitos
Controle de Acesso	<ul style="list-style-type: none"> ✓ Acesso a instalações ✓ Controle de Portas ✓ Acesso a registros médicos ✓ PCs ✓ Smartphones

Fonte: do Autor.

Muitas vezes sistemas de reconhecimento facial são integrados em aplicações de acesso físico ou lógico e são mais propensos a serem usados na identificação ou na vigilância em larga escala. Existem dois componentes básicos para aplicar a tecnologia de reconhecimento facial, uma câmera, capaz de capturar imagens nítidas, e um *software* capaz de reconhecer faces e as rastrear a partir de uma base de dados. Normalmente estes dois componentes ficam separados, mas atualmente com o rápido crescimento tecnológico, estes componentes já podem ser encontrados em smartphones e tablets (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

Imagens faciais são facilmente obtidas por meio de passaportes, carteiras de motorista e RG, fazendo do reconhecimento facial uma escolha óbvia para algumas aplicações (ASHBOURN, 2014, tradução nossa). Para reconhecer o rosto de uma pessoa, os programas mapeiam a geometria e as proporções da face, como exemplificado na figura 5.

Figura 5 - Reconhecimento da Face



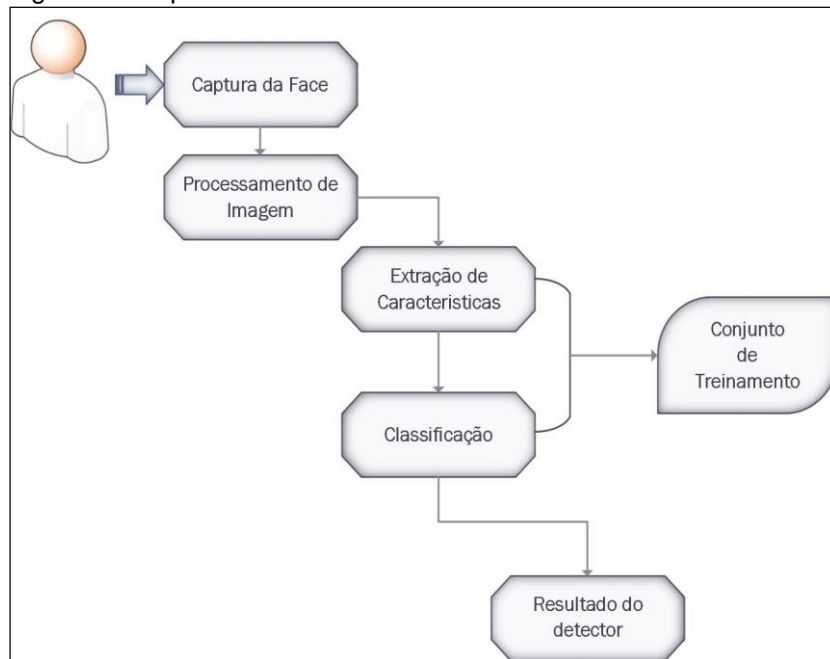
Fonte: Adaptado de Clapham Studio Hire (2015).

É possível adquirir rostos de praticamente qualquer sistema de câmera ou vídeo que gerem imagens de boa qualidade e resolução suficiente, imagens consideradas ideais para o reconhecimento facial são de alta resolução, com o usuário de frente para a câmera, sendo exposto a iluminação moderada, o não enquadramento destes requisitos torna mais difícil a precisão da tecnologia (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

Para chegar ao resultado final, um programa de reconhecimento facial registra vários pontos na imagem, tentando encontrar os pontos delimitadores de uma provável face, os quais permitem definir proporções, distancias e formas de cada elemento contido entre estes pontos e com base nesses dados, iniciar as comparações a fim de identificar esta possível face. Os principais pontos delimitadores são: olhos, nariz, queixo, maçãs do rosto, orelhas, lábios, entre outros (VIGLIAZZI, 2016).

A sensibilidade de sistemas de reconhecimento facial para iluminação pode resultar em redução da capacidade para adquirir rostos de indivíduos de certas raças e etnias, enquanto o cérebro humano é capaz de adquirir imagens faciais sob diferentes condições de iluminação, a tecnologia facial de varredura geralmente é incapaz de adquirir com precisão imagens com problemas de exposição. Existem vários métodos de tratamento de imagem para posterior extração de características, dependendo dos requisitos do algoritmo, parte dos métodos analisados utiliza a imagem normalizada, em escala de cinza a fim de facilitar as comparações, independente do algoritmo de reconhecimento de faces utilizado, sendo correto afirmar que toda a imagem passa por um tratamento antes da análise de características (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa). As etapas que um sistema de reconhecimento facial passa são baseadas na aquisição da imagem, captura da sequência biométrica, localização das características distintas, criação de um modelo e na avaliação da correspondência entre as amostras, como demonstra a figura 6.

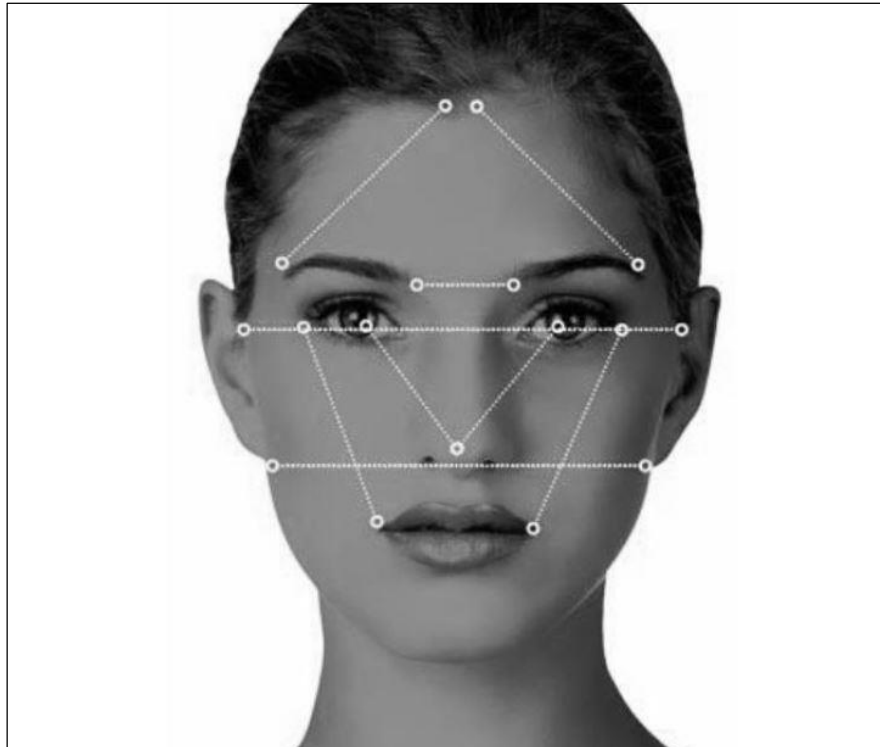
Figura 6 - Etapas do reconhecimento facial



Fonte: do Autor.

Embora existam vários métodos de equivalência, todos os sistemas de reconhecimento facial tentam combinar características faciais visíveis de uma forma semelhante a forma que as pessoas reconhecem umas as outras. As características mais frequentemente utilizadas em sistemas de reconhecimento facial são as que possuem mudanças menos significativas ao longo do tempo (figura 7): a área superior dos olhos, em torno das maçãs do rosto, os lados da boca, forma do nariz, e a posição das principais características em relação aos olhos.

Figura 7 - Mapeamento da face



Fonte: Adaptada de Guerrero (2012).

Um dos desafios envolvidos em torno desta tecnologia é que o rosto é uma característica fisiológica razoavelmente mutável. Alterações comportamentais, tais como alteração de penteado, barba e óculos são comportamentos que têm impacto sobre a capacidade dos sistemas faciais de varredura localizar características distintas. Enquanto o cérebro humano pode ser capaz de explicar algumas dessas modificações, sistemas de reconhecimento facial precisam de um recadastramento do usuário cada vez que uma mudança marcante ocorra (NANAVATI; THIEME; NANAVATI, 2002, tradução nossa).

Atualmente os métodos para reconhecimento de características faciais vêm sendo aprimorados e suas técnicas refinadas, garantindo assim um resultado mais preciso (ASHBOURN, 2014, tradução nossa). Nesta pesquisa são comparados dois destes métodos: um através da aplicação do algoritmo de Viola Jones e outro por meio de um hibridismo entre o algoritmo de Viola-Jones e uma RNA, estes métodos são aplicados através de técnicas de visão computacional.

3 VISÃO COMPUTACIONAL

Visão Computacional pode ser considerada a ciência que pesquisa tecnologias que permitam dotar as máquinas com a capacidade de extrair e processar informações através de imagens (BRUNELLI, 2009, tradução nossa). Originado a partir do campo de inteligência computacional, tem sido uma área de pesquisa bastante ativa e crescente nas últimas duas décadas (KOUROUPETROGLOU, 2013, tradução nossa).

Um sistema de visão computacional é treinado para reconhecer características desejadas, objetos e atividades (KRIG, 2014, tradução nossa). Como entrada, estes sistemas recebem uma imagem, ou sequência de vídeo, captada através de câmeras simples, e realizam uma série de processamentos sob estas imagens a fim de prepará-las para detecção e reconhecimento de imagens ou objetos (LIU, 2014, tradução nossa).

Características e padrões de detecção são um problema clássico na área de visão computacional com potencial para todo tipo de aplicação, variando de reconhecimento automático de objetos ou pessoas até inspeção industrial (SUNG; POGGIO, 1998, tradução nossa). O campo de visão computacional está intimamente relacionado com outras áreas, pois pesquisas na área de visão computacional abrangem uma gama de tópicos e aplicações, tais como aprendizagem de máquina, processamento de sinais, análise de imagens médicas, movimento humano, ciência cognitiva, vigilância, entre outras (KOUROUPETROGLOU, 2013, tradução nossa).

Existem várias vantagens em utilizar sistemas baseados em visão computacional. As câmeras são dispositivos de entrada relativamente não-invasivas, em comparação com outros dispositivos, são mais versáteis, sistemas para diferentes aplicações podem ser implementados usando o hardware padrão de câmeras e computadores, em vez de sensores feitos sob medida. Uma imagem também pode capturar uma quantidade muito maior de informação em relação a outros sensores (KOUROUPETROGLOU, 2013, tradução nossa).

Extrair, representar e comparar o conteúdo da imagem é uma das tarefas mais importantes nas áreas de visão computacional e reconhecimento de padrões

(FARINELLA; BATTIATO; CIPOLLA, 2013, tradução nossa). E a compreensão de como os seres humanos percebem o mundo pode inspirar novas representações de dados de imagem e soluções alternativas para alguns dos problemas de visão computacional (LIU, 2014, tradução nossa).

Depois de várias décadas de pesquisa e estudo, produtos comerciais baseados em tecnologia de visão por computador já estão disponíveis no mercado. Por exemplo, o chip de detecção de rostos dentro de câmaras digitais permite que uma câmera ajuste o foco e a exposição para uma melhor qualidade automaticamente. Ao detectar as regiões da face e usando as informações de referência no rosto, a câmera também é capaz de detectar o movimento dos olhos, efeito de olhos vermelhos, ou um sorriso. Algumas aplicações dão um passo além, ao reconhecer os rostos sendo usados como substitutos de nomes de usuários e senhas para que os usuários possam acessar seus PCs sem passar pelo processo de digitar uma senha longa. Outra área que se baseia fortemente em tecnologias da visão computacional é o domínio de vigilância de segurança ou vídeo. Por exemplo, com a montagem de uma ou mais câmeras que vigiam o estacionamento ou a área de entrada de um edifício, o computador executa algoritmos de visão computacional para analisar o vídeo, detectando e acompanhando quaisquer objetos em movimento (LIU, 2014, tradução nossa).

As partes mais importantes do sistema visual humano são os olhos e o cérebro – em particular, a parte do cérebro associada a visão e o córtex visual. O primeiro estágio da análise uma vez que uma imagem tenha sido obtida, é determinar onde as bordas estão na imagem, há evidências de que a identificação de bordas seja uma parte importante no sistema de visão dos mamíferos. Como quase todos os objetos no mundo real tem bordas sólidas de um tipo ou de outro, detectar estas imagens é a primeira etapa no processo de determinar quais objetos estão presentes em uma cena (COPPIN, 2012).

Características distintivas de uma imagem são muitas vezes descritas por recursos de imagens visuais que servem como entrada para aplicações tais como registro de imagem, recuperação de imagem, reconstrução 3D, navegação, reconhecimento de objetos e rastreamento de objetos. A sensibilização para a

necessidade de descrever adequadamente recursos visuais surgiram na década de 1920 no domínio da percepção visual e conceitos fundamentais foram estabelecidos para que quase todas as abordagens na extração de recursos possam ser rastreadas. Após a transferência das ideias básicas para o campo de visão computacional, muita investigação tem sido realizada, incluindo o desenvolvimento de novos conceitos e métodos para a extração de tais características, a melhoria das ideias prevaletentes e comparações de vários métodos diferentes (FARINELLA; BATTIATO; CIPOLLA, 2013, tradução nossa).

No final dos anos 1990 duas linhas de pesquisa contribuíram de forma significativa, apresentando técnicas que vieram a se tornar populares no campo de inteligência computacional, uma delas foi a pesquisa de reconhecimento de padrões com ênfase em faces humanas baseado em redes neurais artificiais de Rowley, Bajula e Canadá (1998) apresentando uma taxa de 92.7% de acerto e o de Viola e Jones (2000), que em seus trabalhos demonstraram o poder das técnicas de reconhecimento de padrões em tarefas importantes e úteis (RUSSELL; NORVIG, 2013). Entretanto, antes da aplicação dos algoritmos a imagem necessita passar por algumas transformações para ser normalizada.

3.1 PROCESSAMENTO DE IMAGENS

Uma única imagem contém tantas informações quanto um grande texto descritivo, no entanto essas informações têm de ser extraídas e analisadas de um modo eficiente. Tais informações são de grande importância em visão computacional e reconhecimento de padrões (FARINELLA; BATTIATO; CIPOLLA, 2013, tradução nossa).

Para sistemas de visão computacional, o processamento de imagens digitais é um aliado para melhorar a imagem de entrada, impactando diretamente no resultado final do processamento de padrões (SRIVASTAVA; SINGH; SHUKLA, 2013, tradução nossa).

A formulação clássica do problema de reconhecimento de objetos data de 1960, o processo de extração de recurso começou por extrair características

categóricas. Como resultado, a maioria dos sistemas de reconhecimento até os anos 90 tentou extrair recursos desta forma, geralmente começando com a extração de bordas (DICKINSON; PIZLO, 2013, tradução nossa). Atualmente o papel do processamento de imagem é a de modificar a imagem original para a melhoria da qualidade a fim de extrair informações por meio de técnicas de restauração como: segmentação, detecção de formas, textura e análise (SRIVASTAVA; SINGH; SHUKLA, 2013, tradução nossa).

O processamento de imagem é usado frequentemente para melhorar a qualidade da imagem para posterior processamento feito por algoritmos de visão (PARKER, 2010, tradução nossa). Normalmente as manipulações de imagens descritas a seguir são necessárias,

3.1.1 Escala de tons de cinza

Uma das tarefas mais comuns no processamento de imagem e edição é alterar a cor de uma imagem. Imagens em escala de cinza podem encontrar suas inúmeras aplicações em edição de fotos preto e branco, nos clássicos filmes com colorização, e ilustrações científicas (CHEN et al, 2004).

Imagens digitais contém um modo de cores, normalmente, estas imagens estão nos canais Vermelho, Verde e Azul (RGB), outro canal comumente trabalhado é o Ciano, Magenta, Amarelo e Preto (CMYK) comumente referido como cor de processo, este modo normalmente é usado para imagens que necessitam de uma boa definição de cores, como em impressões de imagens (OBERMEIER; PADOVA, 2009).

Geralmente, em câmeras comuns é utilizado o canal RGB. Para transformar uma imagem em tons de cinza é necessário que se aplique filtros individualmente em cada canal RGB ou CMYK (DAVIES, 2012).

Visualmente características únicas aparecem quando uma imagem colorida é convertida em escala de cinza. Esta transformação acontece quando é executado o mapeamento em escala de cinza de imagens coloridas conforme pode-se observar na figura 8 (GOOCH et al, 2005).

Figura 8 - Imagem em tons de cinza



Fonte: Adaptado de Clapham Studio Hire (2015).

Entretanto em sistemas de visão computacional se torna necessário o realce de imagens para a captura de detalhes com uma maior precisão (GOOCH et al, 2005).

3.1.2 Realce de imagem

O objetivo principal das técnicas de realce é processar a imagem de modo que o resultado seja mais adequado para uma aplicação específica do que a imagem original (GONZALEZ; WOODS, 2000).

O componente de cor é representado com uma precisão de 8 bits, os valores entre 0 e 255 são usados para representar um componente de cor particular. Com técnicas de realce são encontrados os valores máximo e mínimo de um componente, e então aplicado um cálculo matemático para obter o componente de cor máxima 255. Este processo aumenta as cores da imagem, por exemplo o verde se torna mais verde, a cor amarela torna-se mais amarela, e assim por diante (MALEPATI, 2010, tradução nossa).

Por meio do realce, os detalhes mais finos de uma imagem são destacados a fim de se ter uma ideia dos objetos exibidos em uma imagem conforme a figura 9 (MARQUES FILHO; VIEIRA NETO, 1999)

Figura 9 - Realce em uma imagem em escala de cinza



Fonte: Adaptado de Clapham Studio Hire (2015).

Além do realce torna-se necessário aplicar um filtro para remoção de ruído para diminuir os níveis de granulosidade da imagem (MONTABONE, 2009, tradução nossa).

3.1.3 Remoção de Ruído

O ruído, no mundo de processamento de imagem, é uma flutuação aleatória no brilho da imagem (LANIER, 2009, tradução nossa). Em imagens digitais, pixels aleatórios coloridos que não representam a cena são chamados de ruído. Cada câmara digital produz uma espécie de ruído nas suas imagens, devido à utilização de sensores digitais, que não são perfeitos (MONTABONE, 2009, tradução nossa).

Existem uma infinidade de filtros utilizados para diminuir o nível de ruído nas imagens. Um dos filtros mais populares para a redução de ruído é chamado Gaussian Blur que aplica um desfoque na imagem para de reduzir o ruído, preservando as bordas dos objetos (MONTABONE, 2009, tradução nossa). A transformação que tal filtro opera na imagem pode ser observado na figura 10.

Figura 10 - Aplicação do filtro Gaussian Blur em imagens em escala de cinza



Fonte: Adaptado de Clapham Studio Hire (2015).

3.1.4 Segmentação

Para Gonzales e Woods (2000) a segmentação de imagens pode ser enquadrada na área de análise de imagem. A segmentação visa identificar segmentos que podem ser utilizados para descrever o conteúdo de uma imagem, como categorias específicas de objetos como por exemplo olho, boca ou cabelo (KLETTE, 2014, tradução nossa).

A técnica de segmentação de imagens cria segmentos de pixels conectados através da análise de alguns critérios de similaridade, possivelmente

suportado pela detecção de pixels que mostram alguma dissimilaridade com pixels adjacentes (KLETTE, 2014, tradução nossa).

Alguns autores acreditam que esta segmentação deva ser feita com imagens monocromáticas, pois na classificação de pixels em imagens em escala de cinza, pixels com níveis semelhantes em uma região próxima geralmente pertencem a um mesmo objeto, desta maneira a complexidade do objeto é reduzida e os procedimentos de reconhecimento e classificação se tornam mais simples, dada a complexidade dos dados terem sido simplificadas (PARKER, 2010, tradução nossa).

Figura 11 - Segmentação em escala de cinza



Fonte: Adaptado de Clapham Studio Hire (2015).

Ao fim destas transformações a imagem está pronta para ser submetida ao algoritmo de extração de características.

3.2 ALGORITMO DE VIOLA-JONES

O reconhecimento facial está diretamente ligado com o reconhecimento e detecção de objetos, pois é por meio dele que se obtém modelos relacionados a face.

A detecção de objetos em imagens estáticas e vídeo está entre as técnicas mais aplicadas que se originam da visão computacional (HWU, 2011, tradução nossa).

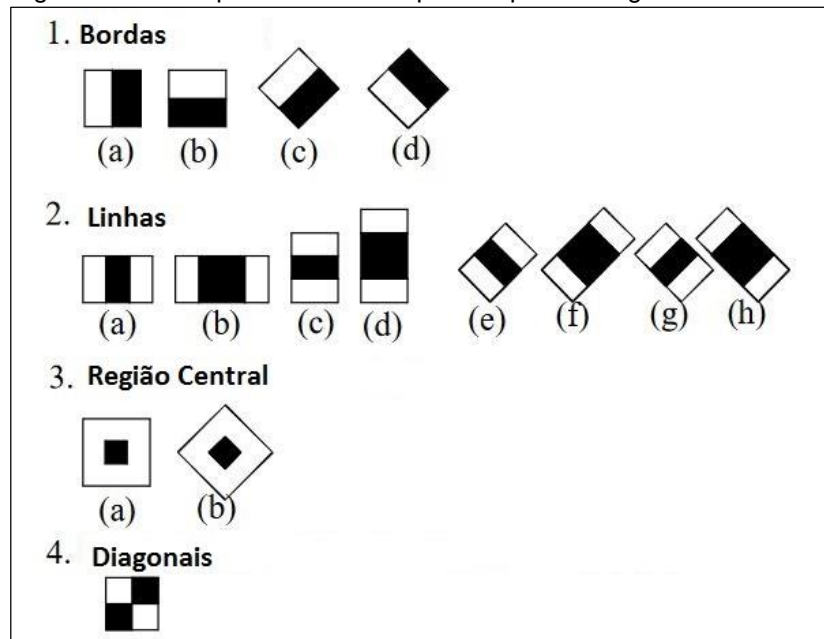
O processo de formação de um detector de faces depende de três componentes principais. O primeiro componente, que é a chave para o sucesso do detector, é a qualidade e a variação representadas no conjunto de dados de treinamento usados. O segundo, é a seleção do tipo de extração de características que irão ser alimentados para o módulo de classificação. Por último, o modelo de classificação de padrões deve ser selecionado (TRAORE; AHMED, 2011, tradução nossa).

O algoritmo de detecção de rosto mais famoso nesta categoria é o algoritmo de Viola-Jones (COLIC; MARQUES; FURHT, 2014, tradução nossa). Paul Viola e Michael Jones apresentaram sua estrutura para detecção de objetos no início de 2001. É amplamente utilizado em uma variedade de aplicações de *software* e *hardware* que incorporam elementos da visão computacional, como o módulo de detecção de rosto em videoconferência, interação humano-computador, e câmeras fotográficas digitais (HWU, 2011).

Viola e Jones (2001, tradução nossa) descrevem um *framework* de detecção de rosto que é capaz de processar imagens de forma rápida conseguindo altas taxas de acerto. Este detector se baseia em três conceitos chave: o uso de uma representação integral da imagem, um classificador utilizando o algoritmo de aprendizagem AdaBoost¹ e o uso de *Harr-cascades* para analisar potenciais regiões de face.

Primeiramente, para a construção do detector torna-se necessária a obtenção da imagem integral a partir da imagem de entrada. Para compor a imagem integral são usados como base *features* para a classificação da área de interesse da imagem (VIOLA; JONES, 2004, tradução nossa). Na figura 12 tem-se as *features* que o sistema utiliza, as áreas escuras indicam uma área negativa, ou seja, não existe uma provável face no local, as áreas claras indicam a presença de uma possível face.

¹ Um algoritmo iterativo que gera um conjunto de classificadores que participam na classificação de exemplos de teste por votação ponderada (FACELI et al, 2015).

Figura 12 - Exemplo de *features* que compoe a imagem

Fonte: Adaptado de Lienhart, Kuranov e Pisarevsky (2002).

A utilização de *features* aumenta significativamente o aprendizado do sistema de detecção e pode ser computada rapidamente (LIENHART; KURANOV; PISAREVSKY, 2002, tradução nossa). A soma dos *pixels* encontrados no triângulo claro são subtraídas da soma dos *pixels* encontrados no triângulo escuro. Estes valores são submetidos utilizados no cálculo da imagem integral que é descrita pela equação (1):

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

Onde $ii(x, y)$ representa a imagem integral e $i(x, y)$ representa a imagem original. Utilizando a quantidade de *features* encontrada é gerada a soma cumulativa da imagem original e da imagem integral. Esta soma é encontrada utilizando o seguinte par de coordenadas:

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (2)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y) \quad (3)$$

Onde $s(x, y)$ é a soma da imagem original, e $ii(x, y)$ é a imagem integral, como pode ser observado nas equações (2) e (3), a imagem integral é calculada usando como referência a imagem original. Usando a imagem integral qualquer quantidade de retângulos pode ser calculada (VIOLA; JONES, 2004, tradução nossa).

O conjunto de *features* provê uma representação de imagem de qualidade, que juntamente com a imagem integral fornece um conjunto de aprendizagem eficaz (VIOLA; JONES, 2001, tradução nossa). Com um conjunto de treino de imagens positivas e negativas, qualquer algoritmo de aprendizado pode ser usado para a função de classificação de imagens, entretanto o AdaBoost possui um poderoso conceito de aprendizagem (LIENHART; KURANOV; PISAREVSKY, 2002, tradução nossa).

O AdaBoost é um método de aprendizagem que constrói um classificador combinando recursos de classificadores fracos. Inicialmente todas as amostras do conjunto de treinamento são tratadas de forma igual pelo algoritmo, posteriormente, em cada nova iteração o AdaBoost utiliza o classificador mais fraco para determinar a função de limiar ótimo, de modo que um número mínimo de exemplos seja classificado. Ao final, o classificador forte apresentado pelo algoritmo é uma combinação linear dos classificadores fracos (TRAORE; AHMED, 2011, tradução nossa). A figura 13 tem-se o pseudocódigo do Adaboost.

Figura 13 - Pseudocódigo do Adaboost

```

Entrada: Um algoritmo de aprendizado  $\phi$ 
Um conjunto de treinamento  $\mathbf{D} = \{(\mathbf{x}_i, y_i), i = 1, \dots, n\}$ 
Número de Iterações  $N_r$ 
Um conjunto de teste com  $nt$  exemplos  $\mathbf{T} = \{(\mathbf{x}_j, ?), j = 1, \dots, nt\}$ 
Saída: Previsões para o conjunto de teste
/* Fase de Treinamento */ ;
para cada exemplo  $i \in \mathbf{D}$  faça
     $w(i) \leftarrow 1/n$ ;
fim
para cada  $l = 1$  to  $N_r$  faça
    para cada exemplo  $i \in \mathbf{D}$  faça
         $p_l(i) \leftarrow w_l(i) / \sum_i w_l(i)$  ;
    fim
    /* Chamada ao Algoritmo de Aprendizado */ ;
     $\hat{f}_l \leftarrow \phi(p_l)$  ;
    /* Calcular o Erro */ ;
     $e_l = \sum_i p_l(i) [ \hat{f}_l(\mathbf{x}_i) \neq y_i ]$  ;
     $\beta_l \leftarrow e_l / (1 - e_l)$  ;
    para cada exemplo  $i \in \mathbf{D}$  faça
         $w_{l+1}(i) := w_l(i) \beta_l^{1 - [ \hat{f}_l(\mathbf{x}_i) \neq y_i ]}$  ;
    fim
fim
/* Fase de Teste */ ;
para cada  $j = 1$  to  $nt$  faça
     $\hat{y}_j = \arg \max_{y \in Y} \sum_{l=1}^{N_r} (\log \frac{1}{\beta_l}) [ \hat{f}_l(\mathbf{x}_j \in \mathbf{T}) = y ]$  ;
fim
Retorna: Vetor de previsões  $\hat{\mathbf{y}}$ ;

```

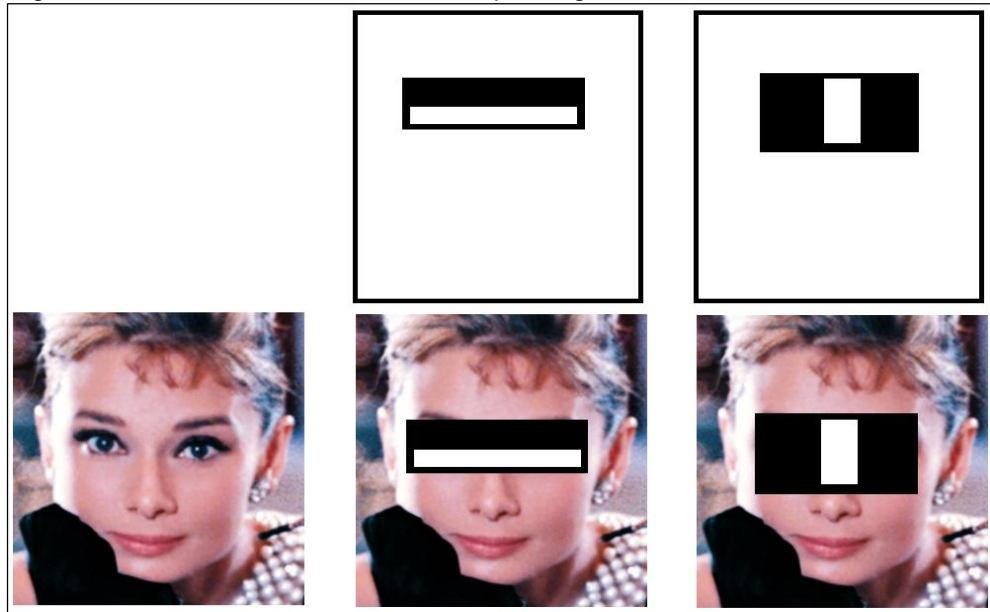
Fonte: Faceli et al (2015).

No algoritmo de Viola-Jones o Adaboost é usado tanto para selecionar um pequeno conjunto de recursos quanto para treinar o classificador. Na prática, não existe um recurso único que possa executar a tarefa de classificação com uma baixa taxa de erro. No estudo apresentado por Viola e Jones (2001, tradução nossa) características que foram selecionadas no início de cada rodada, afim de impulsionar o processo, obtiveram uma taxa de erro entre 0,1 e 0,3 enquanto características selecionadas em rodadas posteriores, quando a tarefa se tornava mais difícil, as taxas de erro ficaram entre 0,4 e 0,5.

Na figura 14 tem-se a primeira e a segunda características selecionadas por AdaBoost. As duas características são mostradas na linha superior e, em seguida, a demonstração da aplicação destas características em uma face. O Adaboost utiliza um recurso que consegue identificar a região dos olhos e do nariz, pois estas áreas são frequentemente mais escuras do que a da face (VIOLA; JONES, 2001, tradução nossa). O segundo quadro exibido no segundo quadrante apresenta a diferença em

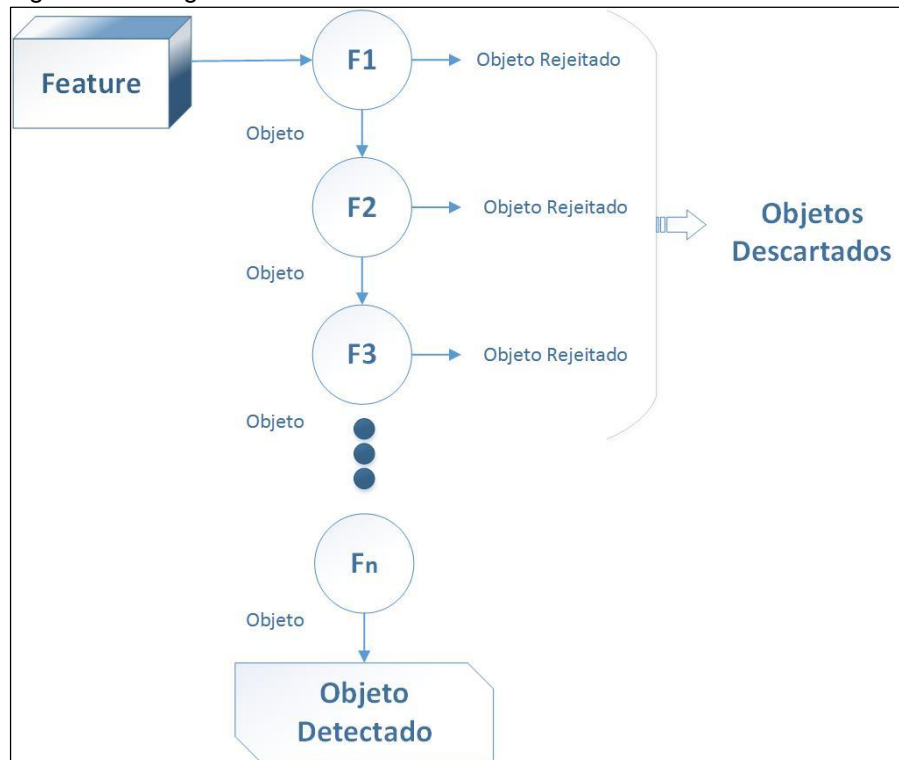
comparação com toda a face entre a região dos olhos, e o terceiro quadro mostra a diferença entre o nariz e o resto da face.

Figura 14 - Características selecionadas pelo algoritmo AdaBoost



Fonte: Adaptado de Clapham Studio Hire (2015).

Na última etapa de detector o algoritmo divide o classificador forte fornecido pelo processo do AdaBoost em uma sequência de classificadores menores por meio de *harr-cascades* (TRAORE; AHMED, 2011, tradução nossa). A técnica de *harr-cascades* se baseia em árvores de decisão onde a cada nó possui um classificador que é treinado para detectar os objetos de interesse (figura 15), no caso do algoritmo de Viola-Jones, faces (LIENHART; KURANOV; PISAREVSKY, 2002, tradução nossa).

Figura 15 - Diagrama de funcionamento do *harr-cascades*

Fonte: do Autor.

O classificador inicial elimina um grande número de imagens negativas com pouco processamento. Depois de várias fases de processamento do número de *features* foram reduzidos radicalmente (VIOLA; JONES, 2004, tradução nossa). Esta configuração permite o descarte de uma determinada *feature* antecipadamente quando esta não contém uma face. Como a maioria das *features* não irá conter faces, isso acelera o processo, permitindo que o detector preserve recursos computacionais e concentre-se em possíveis *features* com rostos (TRAORE; AHMED, 2011, tradução nossa).

Atualmente os algoritmos para reconhecimento de características vêm sendo aprimorados e suas técnicas refinadas, garantindo assim um resultado mais preciso. Algoritmos de reconhecimento de características vem sendo acoplados a RNA para aprimorar e refinar o processo de reconhecimento (ASHBOURN, 2014, tradução nossa).

3.3 REDES NEURAIIS ARTIFICIAIS

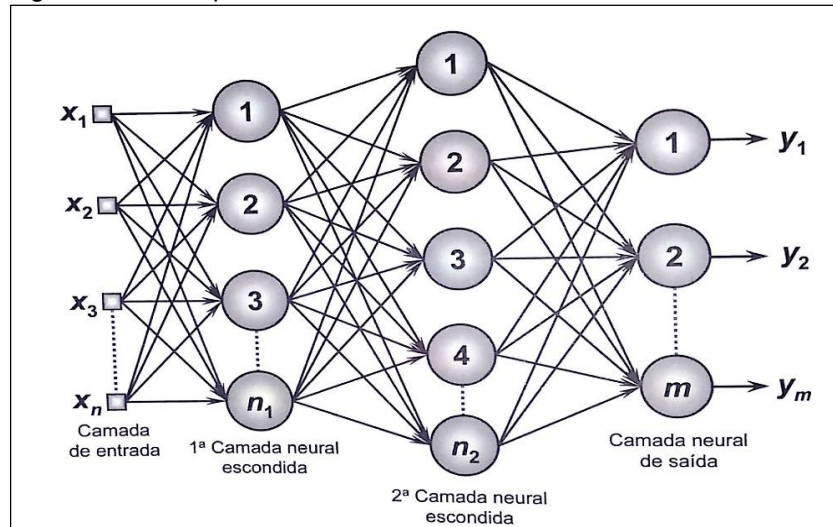
Redes Neurais Artificiais são redes computacionais que tentam simular o processo de decisão do sistema nervoso central biológico, por meio de uma ou mais redes de neurônios organizados em camadas (GRAUPE, 2013, tradução nossa). Desde 1943 têm sido desenvolvidos modelos muito mais detalhados e realistas, tanto em neurônios como em sistemas maiores do cérebro, levando ao campo moderno da neurociência computacional (RUSSELL; NORVIG, 2013).

A palavra artificial é comumente usada para descrever as RNA, a fim de diferenciá-las das redes neurais biológicas que compõe o cérebro humano (COPPIN, 2012). As redes neurais representam funções não lineares complexas com uma rede de unidades de fronteira linear (RUSSELL; NORVIG, 2013).

Haykin (2001) define que uma rede neural artificial é uma máquina projetada para modelar a maneira como o cérebro realiza uma tarefa particular em função de interesse. Para alcançarem um bom desempenho, as RNA empregam um algoritmo de aprendizagem para treinar seus neurônios modificando os pesos da rede de uma forma ordenada para alcançar o objetivo de um projeto desejado.

A arquitetura de uma RNA define a forma como seus neurônios serão estruturados, essa organização ocorre por meio do direcionamento das conexões entre os neurônios (SILVA; SPATTI; FLAUZINO, 2010). Redes *single-layer perceptron* possuem apenas uma camada de neurônios, são normalmente utilizadas em classificações de padrões e filtragem linear. Redes *multi-layer perceptron* são capazes de modelar funções mais complexas, incluindo aquelas que não são linearmente separáveis e possui no mínimo três camadas, a primeira camada é a camada de entrada, responsável por receber informações do meio externo e encaminha os sinais de entrada para a próxima camada, que geralmente é uma camada oculta. Uma rede pode ter uma ou mais camadas ocultas e são nestas camadas que quase todo o processamento interno da rede é realizado. A camada de saída opera na última etapa de processamento, sendo responsável pela produção e apresentação dos resultados finais, conforme demonstrado na figura 16 (COPPIN, 2012).

Figura 16 - Exemplo de rede MLP



Fonte: Silva, Spatti e Flauzino (2010).

A função dos neurônios ocultos, também chamado de escondidos, é intervir entre a camada de entrada e a camada de saída. Adicionando uma ou mais camadas ocultas, a rede MLP se torna capaz de extrair estatísticas de ordem elevada, onde a rede adquire uma perspectiva global apesar de sua conectividade local, devido ao conjunto extra de conexões sinápticas e da dimensão extra de iterações neurais. A habilidade da camada oculta de extrair características de ordem elevada é particularmente valiosa quando o tamanho da camada de entrada é grande (HAYKIN, 2001).

Modelos neurais afirmam que a inteligência surge em sistemas de componentes simples por meio de um processo de aprendizagem, pelo qual as conexões entre os componentes são ajustadas, o processamento destes sistemas é distribuído por meio de camadas de neurônios (LUGER, 2013).

A principal característica da estrutura das RNA é sua habilidade de aprendizagem pelo ambiente, permitindo lidar com situações não definidas (HAYKIN, 2001). Para tal existem dois paradigmas de aprendizado em redes MLP: na aprendizagem supervisionada a RNA tem um professor, que possui exemplos de entrada e saída a qual será transmitida pela rede por meio de treinamento (HAYKIN, 2001). Já no aprendizado não supervisionado, não existe um conhecimento prévio ou

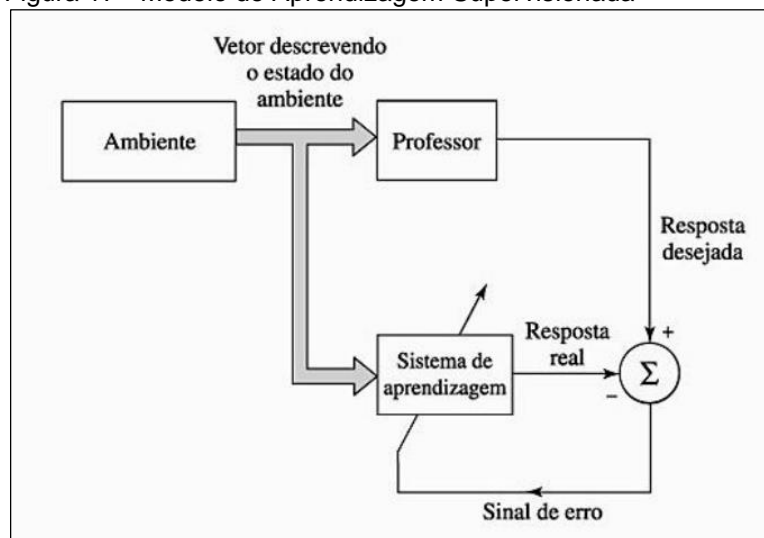
qualquer outro método externo que indique o tipo de entrada que a rede irá receber e requer que o próprio algoritmo de aprendizado avalie os conceitos (LUGER, 2013).

Considerando que as redes MLP são caracterizadas pela versatilidade de suas aplicações de acordo com Silva, Spatti e Flauzino (2010), e tendo como áreas de destaque a identificação e o reconhecimento de padrões, optou-se por abordar neste estudo redes MLP aplicada à abordagem supervisionada pois, idealmente a rede se torna mais instruída sobre o seu ambiente após cada iteração do processo de aprendizagem (HAYKIN, 2001).

3.3.1 Aprendizagem Supervisionada

A aprendizagem supervisionada (figura 17), também denominada aprendizagem com um professor, consiste em uma rede neural artificial sem conhecimento do ambiente em que irá atuar e um professor que possui exemplos de entrada-saída deste ambiente (HAYKIN, 2000). Este professor irá indicar o comportamento que a rede deve possuir, fornecendo a esta um modelo de resposta desejada (BARRETO, 2002).

Figura 17 - Modelo de Aprendizagem Supervisionada



Fonte: Luger (2013).

Os parâmetros da rede são ajustados sob a influência do professor, este ajuste é feito passo a passo, com o objetivo de se adquirir o conhecimento do

professor para a rede neural (LUGER, 2013). Desta forma, quando o conhecimento do ambiente disponível for adquirido pela rede neural, o professor pode ser dispensado, pois a rede neural já está capacitada a lidar com o ambiente por si própria (RUSSELL; NORVIG, 2013).

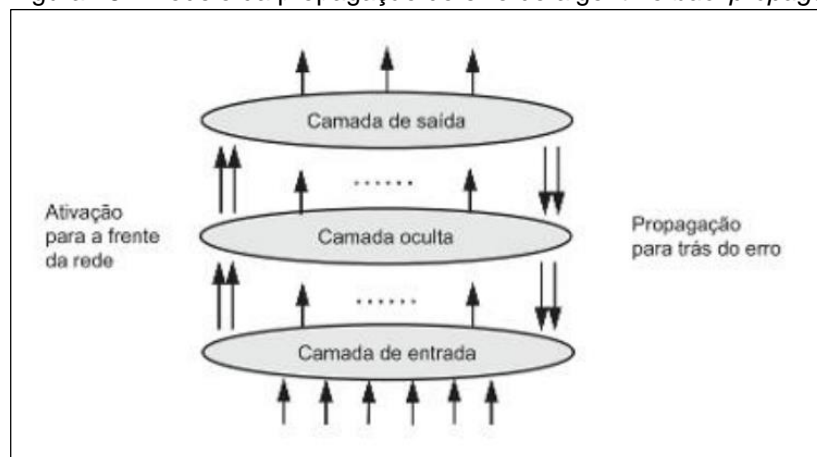
Para Luger (2013) uma classe de conceitos pode ser aprendida desde que exista um algoritmo eficiente que tenha a probabilidade de encontrar um conceito aproximadamente correto. Segundo Haykin (2001) as redes MLP tem sido aplicadas com sucesso por meio do seu treinamento de forma supervisionada pelo algoritmo *backpropagation*.

3.3.1.1 Algoritmo *Backpropagation*

O algoritmo *backpropagation* é baseado na regra de aprendizagem por correção de erro (HAYKIN, 2001). Este método fornece um algoritmo para atribuir aos neurônios uma parcela do erro da rede e ajustar os pesos de forma correspondente (LUGER, 2013).

O algoritmo de *backpropagation* implementa um gradiente de descida no espaço de parâmetros para minimizar o erro de saída (RUSSELL; NORVIG, 2013). A abordagem adotada pelo algoritmo consiste em iniciar na camada de saída e propagar o erro retroativamente através das camadas ocultas (figura 18) (LUGER, 2013).

Figura 18 - Modelo da propagação do erro do algoritmo *backpropagation*



Fonte: Luger (2013).

Inicialmente são atribuídos valores aleatórios aos pesos na rede, que geralmente fica entre -0,5 e 0,5. Cada iteração do algoritmo envolve desde o início a alimentação de dados por meio da rede, desde as entradas até as saídas. A próxima fase, que dá origem ao nome do algoritmo, envolve a alimentação de erros de volta através da rede, fazendo mudanças nos pesos dos nós, ao longo do caminho. O algoritmo segue deste modo, até que as saídas para os dados se aproximem dos valores desejados (COPPIN, 2012). A derivada parcial demonstrada pela equação (4) realiza o cálculo de erro cometido pelo neurônio de saída quando sua resposta é comparada a desejada, medindo a contribuição de cada peso para a classificação de um objeto (FACELI et al, 2015).

(4)

$$e_l = \frac{1}{2} \sum_{q=1}^k (y_q - \hat{f}_q)^2$$

Se o resultado desta derivada for positivo, significa que o peso está provocando um aumento da diferença entre a saída da rede e a saída desejada, e se o resultado da saída for negativo, significa que o peso está contribuindo para que a saída da rede fique mais próxima da desejada (FACELI et al, 2015). O funcionamento do algoritmo está representado através de seu pseudocódigo demonstrado na figura 19.

Figura 19 - Pseudocódigo *Backpropagation*

```

Entrada: Um conjunto de  $n$  objetos de treinamento
Saída: Rede MLP com valores dos pesos ajustados
Inicializar pesos da rede com valores aleatórios
Inicializar  $erro_{total} = 0$ 
repita
  para cada objeto  $x_i$  do conjunto de treinamento faça
    para cada camada da rede, a partir da primeira camada intermediária faça
      para cada cada neurônio  $n_{ji}$  da camada atual faça
        Calcular valor da saída produzida pelo neurônio,  $\hat{f}$ 
      fim
    fim
    Calcular  $erro_{parcial} = y - \hat{f}$ 
    para cada camada da rede, a partir da camada de saída faça
      para cada cada neurônio  $n_{ji}$  da camada atual faça
        Ajustar pesos do neurônio utilizando Equação 4
      fim
    fim
    Calcular  $erro_{total} = erro_{total} + erro_{parcial}$ 
  fim
até  $erro_{total} < \xi$ ;

```

Fonte: Faceli et al (2015).

4 TRABALHOS CORRELATOS

A detecção de características e a performance dos sistemas de visão computacional são temas ainda muito discutidos no meio acadêmico, neste capítulo são apresentados alguns trabalhos correlatos a esta pesquisa.

4.1 BIOID – APLICAÇÃO EXPERIMENTAL DE IDENTIFICAÇÃO BIOMÉTRICA BASEADA EM IMPRESSÃO DIGITAL

O Trabalho de Conclusão do Curso de Ciência da Computação de Zomer (2005) na Universidade do Extremo Sul Catarinense, teve como objetivo implementar um modelo de identificação biométrica pelo método de comparação de impressões digitais fazendo uso de redes neurais artificiais com treinamento Backpropagation.

Na implementação da ferramenta foram utilizadas a linguagem de programação Object Pascal no ambiente de desenvolvimento Borland Delphi, na versão 6.0. As imagens de impressões digitais utilizadas para o reconhecimento e treinamento foram obtidas utilizando-se o *software* Sfinge, o qual gera imagens de impressão digital de forma randômica.

Para o treinamento da RNA foram separados dois grupos de amostras contendo tipos diferentes de partes de impressões digitais, sendo que o grupo de teste possuía 75 imagens e o de treinamento 100.

Segundo Zomer (2005) sua rede não conseguiu identificar digitais com inclinações, entretanto este fato foi atribuído ao pouco tempo e ao conjunto pequeno de imagens usadas para o treinamento. A autora obteve com seu protótipo 80% de acerto em um grupo com 75 amostras testadas, podendo localizar corretamente pontos que se evidenciam na impressão digital para realizar o cálculo das distâncias entre os mesmos, medidas estas, utilizadas no momento do reconhecimento.

4.2 RECONHECIMENTO FACIAL SIMULTÂNEO USANDO O ALGORITMO DE VIOLA-JONES E REDES NEURAIS ARTIFICIAIS PARA VERIFICAÇÃO DE IDENTIDADE

No artigo de Fernandez et al (2014) publicado no 10º Simpósio Regional de Kuala Lumpur do IEEE, foi apresentado um protótipo de reconhecimento facial utilizando o algoritmo de Viola – Jones para a detecção de imagens e uma rede neural artificial para reconhecer as faces detectadas anteriormente.

Esta pesquisa teve um maior foco no problema de luminosidade descrito Viola-Jones e outros pesquisadores da área. A arquitetura do protótipo foi dividida em: detecção de faces, extração de características, reconhecimento de uma RNA, e o treinamento de imagens.

Para o treinamento das Harr-cascades e da RNA, os autores utilizaram 1050 imagens por pessoa e foram extraídos sete tipos de características, estas características foram submetidas a RNA para identificação da pessoa. A RNA criada contava com 4 neurônios de entrada, 20 intermediários e 4 de saída.

Ao final do experimento Fernandez et al (2014, tradução nossa) determinaram que seu protótipo alcançou 88,06% de acurácia em condições normais, a performance do sistema em pessoas a 100 centímetros de distância da câmera, conseguiu uma acurácia de 88,8%, e em ambientes com luz especializada foi obtido um percentual de 89,79% de acurácia.

4.3 SISTEMA DE RECONHECIMENTO FACIAL

O Trabalho de Conclusão de Curso de Engenharia Elétrica de Braga (2013) na Universidade de São Paulo teve como objetivo o desenvolvimento de um sistema de reconhecimento facial utilizando o algoritmo de Viola-Jones e uma análise discriminante a fim de preservar o máximo possível de informação da imagem.

Para desenvolver o *software* de reconhecimento Braga utilizou o sistema MATLAB dada as facilidades da linguagem e a rapidez do desenvolvimento. No treinamento do algoritmo foram utilizadas cerca de 3000 imagens com dimensões de

24x24 em escala de cinza, utilizando uma técnica chamada de Pirâmide de Imagens para diminuir a escala das imagens.

Segundo Braga (2013) a maioria das imagens testadas obtiveram um resultado satisfatório, porém algumas amostras apresentaram falsos positivos e falsos negativos. No entanto o autor salienta que a presença destes resultados se deve a quantidade de seu conjunto de testes, que contava com apenas 1000 imagens positivas para treinamento.

No geral o sistema obteve uma baixa taxa de erros e na maioria das condições testadas apresentou resultados satisfatórios, mesmo com a região dos olhos e da boca cobertas. O módulo de extração de características levou cerca de 1,6 segundos para analisar 200 imagens–teste, dessas imagens o módulo de reconhecimento identificou 130 imagens corretamente e uma incorretamente, sendo que o restante foi rejeitado (BRAGA, 2013).

4.4 RECONHECIMENTO DE FACES UTILIZANDO REDES NEURAIS ARTIFICIAIS COM FUSÃO DE CARACTERÍSTICAS FACIAIS POR MÉTODO DE VOTAÇÃO

No artigo de Fernandes e Gonzaga (2007) publicado no III Workshop de Visão Computacional da Sociedade Brasileira de Computação propôs-se a utilização de redes neurais artificiais nos processos de classificação das características individuais e para fusão um método de votação com o propósito de se obter um menor esforço computacional, tanto na etapa de treinamento, quanto na validação e também melhores resultados na extração dos principais componentes.

Para o desenvolvimento do protótipo foi proposto um reconhecimento de faces sobre três características, aplicando-se PCA junto com um discriminante linear de Fisher para identificar cada característica facial e a seguir aplicar um método de fusão probabilística que fornecesse como resultado a pessoa reconhecida.

Fernandes e Gonzaga (2007) realizaram duas análises de seu sistema, na primeira o sistema foi submetido a dois conjuntos de dados diferentes, onde a quantidade de imagens pertencentes a base de dados variou entre 200 e 800 imagens, mostrando que quanto menos classes tiverem, melhor será o resultado. Já

a análise do segundo teste consistia em verificar a viabilidade do sistema quando tem-se a redução do conjunto de treinamento da rede neural, neste caso imagens de cada classe foram dadas como entrada, usando-se no treinamento da rede neural o conjunto de validação com 80% a 95% do conjunto de dados.

Nesta etapa os resultados mostraram uma grande perda da taxa de acertos quando os resultados foram variados entre 80% e 90%, porém, quando variados entre 90% e 95% estes apresentaram pouca diferença mostrando que o sistema se torna mais estável. Vale ressaltar que quando utilizada uma base de dados com apenas 200 imagens, as variações dos conjuntos de validação apresentaram mais estabilidade do que quando utilizada a base de dados com 800 imagens (FERNANDES; GONZAGA, 2007).

4.5 RECONHECIMENTO DE FACES UTILIZANDO REDES NEURAIS E BIOMETRIA

A Dissertação de Oliveira (2006) para o Mestrado de Computação Aplicada no Instituto Nacional de Pesquisas Espaciais apresenta um protótipo de avaliação para a construção de um sistema híbrido inteligente, direcionado ao reconhecimento de faces. Este sistema busca mesclar as características locais e globais da face, usadas nos métodos geométricos, com a capacidade de generalização e robustez a falhas, observadas nos métodos inteligentes baseados em redes neurais artificiais.

Foram usados para determinar pontos de referência, componentes e medidas faciais encontradas em estudos anatômicos da área de ortodontia, fonoaudiologia e reconstituição maxilo-facial (OLIVEIRA, 2006).

Buscando tornar a tarefa de reconhecimento facial mais simples Oliveira (2006) utilizou a estratégia "dividir para conquistar". A primeira tarefa é verificar se uma determinada face é conhecida. Em caso positivo inicia-se a segunda tarefa, que consiste na procura daquela face entre as amostras empregadas no treinamento. Foram utilizadas MLP alimentadas adiante e treinadas pelo algoritmo de retro propagação de erro, fim de verificar se um padrão era conhecido ou não, usou-se também outra rede com as mesmas características estruturais, com um número maior

de entradas e de neurônios escondidos, para avaliar a similaridade entre os padrões vistos na etapa de treinamento e o padrão avaliado naquele momento. O uso da estratégia se mostrou eficaz, pois foram obtidos bons índices de acerto no reconhecimento facial.

Foram alcançados índices de acerto acima de 95% para as redes da etapa 1, avaliando-se padrões vistos na fase de treinamento acrescidos de ruídos. E foram alcançados índices de acerto superiores a 85%, quando apresentados padrões que não estavam presentes na fase de treinamento. As redes da etapa 2 de reconhecimento obtiveram acertos superiores a 97% (OLIVEIRA, 2006).

4.6 ACELERAÇÃO DE DETECÇÃO DE FACES EM TEMPO REAL COM UM ROBÔ DE TELE PRESENÇA UTILIZANDO UM RASPBERRY PI

No artigo de Janard e Maruringsith publicado na 5ª conferência de Inovação de Computação Tecnológica em 2015, é apresentado um robô para rastreamento facial utilizando dois Raspberry Pi modelo B. Neste artigo também é feita uma análise da placa de processamento gráfico do Raspberry e seu desempenho frente a um detector de faces fazendo o uso do *framework* de computação distribuída, o ServoBlaster.

Para o desenvolvimento do protótipo foi conectado um *smartphone* remotamente a duas placas do Raspberry Pi trabalhando em paralelo juntamente com uma webcam, foi utilizado também o *framework* OpenCV para o desenvolvimento do módulo de reconhecimento, o qual, fazia o uso do algoritmo de Viola-Jones para detecção de faces.

Nos experimentos de Janard e Maruringsith (2015) chegou-se à conclusão de que a memória interna de dois Raspberry Pi (1 GB no total) é suficiente para todas as operações de tele presença no robô, incluindo a detecção autônoma de rosto. Entretanto, adicionando o reconhecimento facial no Raspberry o uso da CPU vai ao seu máximo, assim, quando adicionado recursos como mobilidade, um robô para prover o processamento adicional se faz necessário. Ao utilizar o ServoBlaster é adicionado uma sobrecarga sobre a CPU causando uma resposta mais lenta do robô.

Assim, todos os resultados sugeriram que a técnica de detecção de rosto proposto tende a chegar ao uso máximo de recursos computacionais do Raspberry modelo 1B. Para conseguir um tele presença autónoma de baixo custo computacional a largura de banda de comunicação, a velocidade e a separação das cargas de CPU usando paralelismo de tarefas são os fatores-chave (JANARD; MARURNGSITH, 2015, tradução nossa).

5 ANÁLISE DE CARACTERÍSTICAS EM REDES NEURAIS ARTIFICIAIS APLICADAS EM VISÃO COMPUTACIONAL COM ÊNFASE EM RECONHECIMENTO FACIAL

A área de biometria, mais especificamente a de reconhecimento de características vem apresentando uma evolução de seus métodos tecnológicos para detectar faces ao longo das duas últimas décadas, conforme foi mencionado anteriormente. A evolução desta tecnologia beneficia a sociedade em geral, pois suas aplicações são muitas. No desenvolvimento deste protótipo foi necessário aplicar os conceitos aprendidos durante esta pesquisa. Este capítulo apresenta a montagem e desenvolvimento do protótipo concebido a partir dos conceitos aqui abordados, ele se divide em três sessões: a primeira aborda os componentes e recursos utilizados para a montagem do protótipo, a segunda apresenta o desenvolvimento do protótipo, bem como suas etapas de treinamento, e por fim, na terceira sessão são apresentados os resultados obtidos pelo detector.

5.1 COMPONENTES DO PROTÓTIPO

Para o desenvolvimento do protótipo foi necessário levar em consideração a plataforma, linguagem de programação e *frameworks* oferecidos para então decidir a ferramenta mais adequada para a situação. Nas próximas seções são detalhados os componentes necessários para este desenvolvimento, bem como a montagem do protótipo.

5.1.1 Raspberry Pi

O Raspberry Pi é um computador de baixo custo do tamanho de um de cartão de crédito (Raspberrypi.org, 2016, tradução nossa). O projeto Raspberry Pi é visto como uma revolução relacionada a educação em Ciência da Computação, este projeto foi idealizado para obter-se experiência prática na construção e programação de sistemas distribuídos completos (TSO et al, 2013, tradução nossa).

O Raspberry Pi 2 Modelo B, exibido na figura 20, é a segunda geração Raspberry Pi. Ele substituiu o modelo Raspberry Pi 1 Modelo B+ em fevereiro de 2015. Possui um processador ARMv7, podendo executar sistemas operacionais como o Ubuntu, Raspbian, Windows 10 IoT, entre outras distribuições personalizadas (Raspberrypi.org, 2016, tradução nossa).

Figura 20 - Raspberry Pi 2 Modelo B



Fonte: do Autor.

Com a popularização do minicomputador Raspberry Pi entre o ambiente acadêmico tem-se aplicado o mesmo em um grande número de áreas e diferentes projetos (JAIN; VAIBHAV; GOYAL, 2014, tradução nossa). Para este projeto foi adicionada, via conexão USB uma Webcam de 1,3 Megapixels, com a inserção desta Webcam na placa, foi criada uma conexão de vídeo em tempo real, permitindo que a placa obtivesse imagens para serem analisadas pelo *software* desenvolvido com o *framework* OpenCV.

5.1.2 Framework OpenCV

Durante a elaboração do projeto pesquisou-se *frameworks* que ajudassem no desenvolvimento do protótipo, entre eles, o que mais obteve destaque foi o *framework* OpenCV. O OpenCV é o mais conhecido *framework* para o

desenvolvimento de aplicações utilizando visão computacional e possui várias extensões. Fornece utilitários para a leitura a partir de uma grande variedade de tipos de arquivos de imagens, bem como de vídeo e câmeras (BRADSKI; KAEHLER, 2008, tradução nossa).

A biblioteca tem mais de 2500 algoritmos otimizados, á qual inclui um conjunto abrangente de algoritmos de aprendizado clássicos e especializados em visão computacional. Esses algoritmos podem ser usados para detectar e reconhecer rostos, identificar objetos, classificar as ações humanas em vídeos, objetos em movimento (Opencv.org, 2016, tradução nossa). O *framework* OpenCV foi utilizado tanto no desenvolvimento do algoritmo Viola-Jones, quanto na rede neural artificial. Na montagem do protótipo foi necessário instalar algumas dependências para habilitar e compilar os módulos do *framework*, para dar o início ao desenvolvimento.

5.1.3 Montagem do Protótipo

O Raspberry Pi funciona com um mini cartão de memória de no mínimo 8 GB recomendado pelo site oficial, para este projeto foi adquirido um cartão de 16 GB e inserido entrada da placa. Durante a elaboração do projeto foi pesquisado qual Sistema Operacional (SO) seria o mais eficiente em termos de desempenho, quanto praticidade. Foram feitos testes nas distribuições de SO Windows 10 IoT, Ubuntu e Raspbian, entretanto foi escolhido o Raspbian por se tratar de um sistema operacional livre baseado em Debian otimizado para o hardware Raspberry Pi.

Figura 21 - Protótipo montado



Fonte: do Autor.

Após a placa estar funcionando corretamente alguns programas pré-instalados do SO foram deletados, de modo a liberar espaço na memória. O *framework* OpenCV e suas dependências foram instalados.

Figura 22 - Compilação do *framework* OpenCV

```

danie@danie:~/opencv-3.1.0/build
-- C/C++ Examples: NO
-- Install path: /usr/local
-- cvconfig.h is in: /home/comp1/opencv-3.1.0/build
-----
-- Configuring done
-- Generating done
-- Build files have been written to: /home/comp1/opencv-3.1.0/build
comp1@comp1-22:~/opencv-3.1.0/build$ make
Scanning dependencies of target libwebp
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/alpha.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/buffer.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/vp8l.c.o
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/quant.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/webp.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/idec.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/frame.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/layer.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/vp8.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/tree.c.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/io.c.o

```

Fonte: do Autor.

Para o desenvolvimento deste projeto o *framework* foi compilado em ambiente Linux, por meio da distribuição Ubuntu 15.10 utilizando a IDE CodeBlocks com a linguagem C++. O OpenCV também foi instalado no Ubuntu para a realização testes e o desenvolvimento do protótipo. No Raspberry Pi 2 foi iniciada a compilação

do *framework* a fim de treinar o algoritmo de Viola-Jones e a RNA, e coletar os respectivos resultados. Á seguir é detalhado a base de testes utilizada para o treinamento e avaliação da acurácia do software desenvolvido neste protótipo.

5.2 BASE DE TESTES

Para o treinamento do Viola-Jones e da RNA foi utilizado o banco de faces do Dr. Libor Spacek, Mestre em inteligência computacional e pesquisador da área de visão computacional, disponibilizado de forma gratuita na internet², este banco conta com 7.900 imagens de indivíduos de sexo, etnias e idades variadas, contendo a presença de óculos, barba e objetos no fundo. Estas imagens foram feitas com uma câmera profissional em uma luminosidade artificial tendo o formato de JPEG 24bits colorido, sendo que 3.460 imagens de 180x200 *pixels* sendo consideradas de fácil reconhecimento, elas possuem o fundo verde, sem variações de luminosidade, posição, expressão e penteado. 1.400 imagens de 180x200 *pixels* são de reconhecimento intermediário com fundo vermelho havendo movimentação nas fotos e sombras na face decorrentes da luminosidade, estas imagens apresentam variação de luminosidade, posição e pouca variação de expressões.

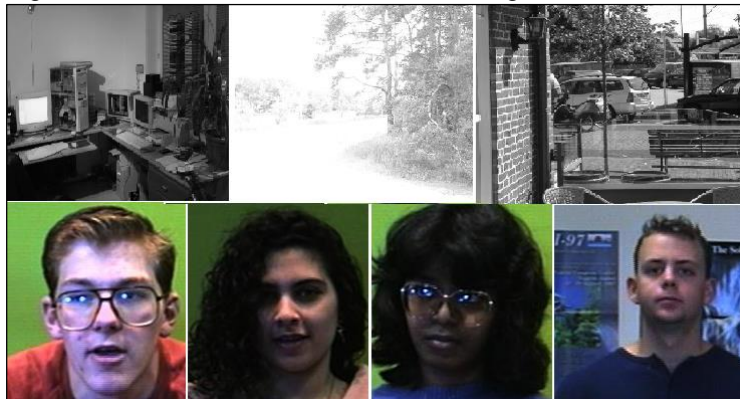
As outras 3.400 imagens são consideradas de difícil reconhecimento e se dividem em dois conjuntos, um de 3.040 imagens tiradas em uma sequência de 20 fotos com 0.5 segundos de intervalo enquanto o indivíduo se movimentava. Estas fotos possuem alguma inclinação da face, posições variadas com um fundo contendo diversos objetos, apresentando uma luminosidade variada com sombras na face decorrente da variação de luminosidade. O outro conjunto possui 360 imagens com o indivíduo fazendo caretas, em diferentes posições com pouca variação de luminosidade e penteados.

Estas imagens foram divididas em um conjunto de treinamento e um conjunto de testes, entretanto algumas imagens estavam corrompidas e as mesmas foram substituídas por imagens da internet com a mesma classificação. A base de

² Disponível em: <http://cswww.essex.ac.uk/mv/allfaces/>

dados de imagens negativas utilizada foi obtida por meio da página de Naotoshi Seo³ em sua página sobre treinamento de cascatas de classificadores. Esta base contém 4979 imagens cinza com resolução 640x480 pixels, disponibilizadas de forma gratuita e que não contém faces de pessoas e foram obtidas em diversos ambientes e condições de iluminação.

Figura 23 - Base de Testes Positivas e Negativas



Fonte: Adaptadas de Spacek(2008) e Seo().

Ao final estes dois conjuntos de imagens somaram 9.518 imagens as quais foram subdivididas em um conjunto de teste e em um conjunto de treino, para realizar esta divisão foi usado o método de Holdout (THEODORIDIS;KOUTROUMBAS, 2003, tradução nossa), utilizando $\frac{2}{3}$ da base de dados inicial como conjunto de testes, o qual contava com 3.140 imagens e $\frac{1}{3}$ para o conjunto de treino contando com 6.378 imagens, após essa definição foi dado início ao desenvolvimento do software de reconhecimento.

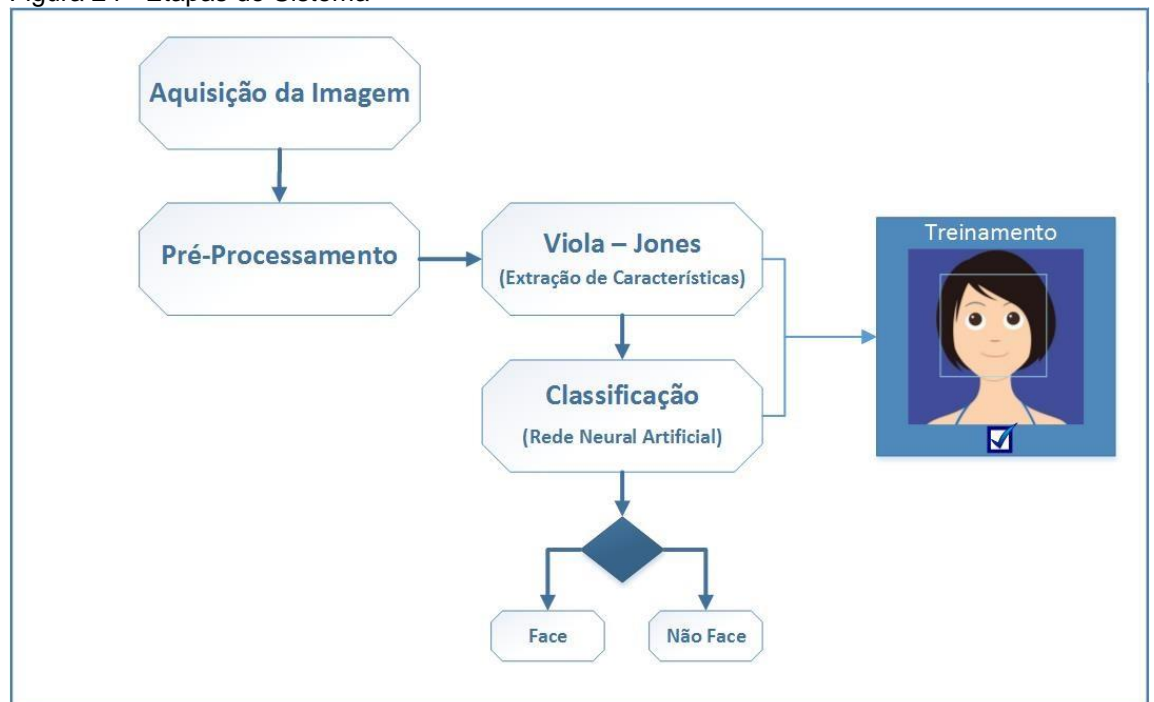
5.3 DESENVOLVIMENTO DO PROTÓTIPO DE SOFTWARE DE ANÁLISE DE CARACTERÍSTICAS EM RECONHECIMENTO FACIAL APLICANDO REDES NEURAS ARTIFICIAIS E O ALGORITMO DE VIOLA-JONES

Na literatura sobre o algoritmo de Viola-Jones são frequentes os problemas relatados acerca da luminosidade ideal para o reconhecimento de uma face. Segundo

³ Disponível em: <http://note.sonots.com/>

Pereira e Viana (2013) a eliminação de informações redundantes devido a problemas de iluminação é fundamental para a obtenção de boas taxas de classificação. O desenvolvimento deste software foi focado para capacitá-lo a analisar uma determinada cena retirada de uma foto ou vídeo em ambientes onde a luminosidade não seja a ideal, que segundo Bandala et al (2014, tradução nossa) é cerca de 480 lúmens. Para atingir este objetivo o projeto foi dividido em quatro etapas (figura 25) com duas fases de treinamento. A primeira etapa é referente a aquisição de imagem que pode ser obtida por meio de uma Webcam para reconhecimento em tempo real. A segunda etapa é o pré-processamento da imagem obtida, onde são feitas as transformações de imagem necessárias a fim de normalizá-la. Na terceira etapa é aplicado o algoritmo Viola-Jones na imagem, antes de dar prosseguimento ao desenvolvimento foi realizado o treinamento das *harr-cascades* da face, dos olhos e da boca, e recolhidos os dados do conjunto de testes.

Figura 24 - Etapas do Sistema



Fonte: do Autor.

Na segunda etapa foi desenvolvido uma RNA e interligada ao detector de Viola-Jones, nas etapas de treinamento e testes da rede usou-se os mesmos

conjuntos aplicados anteriormente. A seguir são descritas mais detalhadamente as etapas e a construção do software.

5.3.1 Preparação da Imagem

Antes de iniciar a aplicação do algoritmo se faz necessário preparar a imagem, para que o processo de extração de características seja otimizado, o *framework* OpenCV conta com uma biblioteca específica para o tratamento de imagens: a *imgproc*. Esta biblioteca possibilita transformações de alta complexidade bem como mais básicas como cortar, girar, esticar e encolher uma imagem.

Para a preparação da imagem foram criadas duas cópias da mesma, em uma delas, foi aplicado o filtro *CV_BGR2GRAY* através da função *cvCvtColor()*, a qual transforma os canais RGB para a escala de cinza da imagem passada como parâmetro para a função *transformImgs()* apresentada na figura 25, e a outra imagem foi criada em uma escala menor, de 1.3 que segundo Bradski e Kaehler (2008, tradução nossa) é a ideal para a aplicação do algoritmo de Viola-Jones, utilizando a função *cvCreateImage()*. Através da função *cvResize()* a primeira imagem, a qual foi aplicada a transformação em escala de cinza, é passada para a imagem de escala menor, por meio de uma interpolação bilinear afim de preservar todos os seus detalhes, esta interpolação é definida pela constante *CV_INTER_LINEAR*.

Figura 25 - Função de transformação de imagens

```

IplImage* transformImgs(IplImage* amframe){
    double scale = 1.3;

    IplImage* gray = cvCreateImage(cvSize(amframe->width,
                                         amframe->height),
                                  8, 1);

    IplImage* small_img = cvCreateImage(cvSize(cvRound(amframe->width/scale),
                                              cvRound(amframe->height/scale)),
                                       8, 1);

    cvCvtColor( amframe, gray, CV_BGR2GRAY );
    cvResize( gray, small_img, CV_INTER_LINEAR );

    cvEqualizeHist( small_img, small_img );
    cvSmooth(small_img, small_img, CV_GAUSSIAN);
    cvThreshold(small_img, small_img, 140, 255, CV_THRESH_BINARY);

    amframe = cvCloneImage(small_img);

    cvReleaseImage(&gray);
    cvReleaseImage(&small_img);

    return amframe;
}

```

Fonte: do Autor.

Logo em seguida é aplicado o realce da imagem para maximizar o contraste dos tons de cinza por meio das técnicas de histograma implementada na função *cvEqualizeHist()*, pois como as *features* se baseiam em diferenças da região a ser analisada o resultado pode ser distorcido caso a imagem possua um desequilíbrio de iluminação ou exposição das imagens.

A função *cvSmooth()* foi utilizada para aplicar a remoção de ruído através do filtro Gaussian Blur definido pela constante *CV_BLUR*⁴, alguns outros filtros também estão disponíveis como o *CV_GAUSSIAN*⁵, que representa o filtro Gaussian original, o *CV_BLUR_NO_SCALE*⁶ que cria uma divisão na imagem e aplica o filtro Gaussian em pontos específicos, o *CV_MEDIAN*⁷ o qual seleciona os pontos médios para aplicar a suavização das bordas e o *CV_BILATERAL*⁸ que propõe uma suavização extrema da imagem. O filtro Gaussian Blur foi escolhido, pois promove uma suavização e um desfoque menos agressivo e constante em comparação com os outros filtros apresentados. Por fim é aplicado um filtro de segmentação de imagens por meio da função *cvThreshold()* que agrupa pixels com níveis análogos

⁴ Desfoca uma imagem usando o filtro Blur normalizado (OpenCV, 2016, tradução nossa).

⁵ Desfoca uma imagem usando um filtro de Gaussian (OpenCV, 2016, tradução nossa).

⁶ Aplica um desfoque utilizando convolução linear (OpenCV, 2016, tradução nossa).

⁷ Desfoca uma imagem usando o filtro mediano (OpenCV, 2016, tradução nossa).

⁸ Desfoca uma imagem utilizando a técnica de desfoque bilateral (OpenCV, 2016, tradução nossa).

visando tornar o objeto menos complexo (PARKER, 2010, tradução nossa). Após estas etapas a imagem já está apta a fornecer dados das características de interesse, no caso faces, por meio do algoritmo de Viola-Jones.

5.3.2 Desenvolvimento do algoritmo de Viola-Jones

Para desenvolver o algoritmo de Viola-Jones, utilizou-se a biblioteca *objdetect* do *framework* OpenCV voltada para a extração de características, que já possui uma implementação do algoritmo englobando todos os seus conceitos chave: como a extração das *features*, a definição da imagem integral sobre as mesmas, o filtro *harr-cascades* e o algoritmo Adaboost.

Além da biblioteca o *framework* conta com alguns arquivos xml de *harr-cascades* prontas para o desenvolvimento, estes arquivos já possuem a configuração básica para encontrar faces, olhos, bocas e corpos inteiros em um *frame*.

Para definir qual o *harr-cascades* a ser utilizado, foram copiados os arquivos *haarcascade_frontalface_alt.xml*, *haarcascade_eyes.xml* e *haarcascade_mouth.xml* responsáveis pelo reconhecimento da face, olhos e bocas respectivamente, para o diretório do projeto, afim de facilitar a chamada no programa.

Após ser referenciado no programa é chamada a função *cvLoad()* para carregar o arquivo para uma estrutura chamada *CVHaarClassifierCascade* (figura 26), responsável por aplicar as métricas definidas no *harr-cascades* nas *features* do *frame*.

Figura 26 - Carregamento do *haar-cascade*

```

106     char *FACE_CASCADE = "haarcascade_frontalface_alt.xml";
107
108     CvHaarClassifierCascade *lcFace;
109     lcFace = (CvHaarClassifierCascade*)cvLoad(FACE_CASCADE, 0, 0, 0);
110

```

Fonte: do Autor.

Em seguida é chamada a função *detectAndDisplay()*, componente principal do algoritmo, pois é nela que está a implementação do algoritmo Viola-Jones (figura 27). Para tal é necessário definir alguns parâmetros como: a imagem a ser analisada, o *harr-cascades* definindo, a escala de *pixels* à ser utilizado na procura, a *flag* *CV_HAAR_DO_CANNY_PRUNING* que determina que o algoritmo rejeite objetos que

possuam muitas bordas (Opencv.org, 2016, tradução nossa), e o limitador de número de vizinhos para a busca na aplicação do algoritmo Viola-Jones.

Figura 27 - Função principal do algoritmo Viola-Jones

```

101  /** função principal de detecção
102  encontra a face e direciona para as outras funções de detecção*/
103
104  void detectAndDisplay( IplImage* amFrame, int aimg_no) {
105
106      char *FACE_CASCADE ="haarcascade_frontalface_alt.xml";
107      char image[100];
108
109      CvRect          *r;
110      CvSeq           *csFace ;|
111      CvMemStorage    *storage = cvCreateMemStorage(0);
112      CvHaarClassifierCascade *lcFace;
113
114      cvClearMemStorage( storage );
115
116      lcFace = (CvHaarClassifierCascade*)cvLoad(FACE_CASCADE, 0, 0, 0);
117      csFace = cvHaarDetectObjects(amFrame,
118                                lcFace,
119                                storage,
120                                1.1, 3,
121                                CV_HAAR_DO_CANNY_PRUNING,
122                                cvSize(20, 20));
123
124      for(int i = 0 ; i < ( csFace ? csFace->total : 0 ) ; i++ ) {
125
126          r = (CvRect*)cvGetSeqElem(csFace, i);
127
128          cvRectangle( amFrame,
129                    cvPoint( r->x, r->y ),
130                    cvPoint( r->x + r->width, r->y + r->height ),
131                    CV_RGB( 0, 255, 255 ),
132                    2, 8, 0 );
133
134          addlog(aimg_no, "FACE", csFace->total, i);
135
136          detectEyes(amFrame,r,storage, aimg_no);
137          cvResetImageROI(amFrame);
138          detectMouth(amFrame, r, storage, aimg_no);
139          cvResetImageROI(amFrame);
140      }
141
142      cvReleaseHaarClassifierCascade( &lcFace );
143      cvReleaseMemStorage( &storage );
144  }
145

```

Fonte: do Autor.

A *CvRect* é uma estrutura utilizada para armazenar os resultados do reconhecimento, a propriedade *CvRect->total* retorna a quantidade de faces existentes da imagem analisada, caso exista uma ou mais imagens ela possuirá outras informações sobre a face como altura, largura e os pontos onde a imagem está localizada. Depois de construída esta estrutura para a face, foi necessário construir uma estrutura semelhante para os olhos e para a boca, para obter-se a posição dos mesmos. O diferencial nestas funções (figura 28), é que elas recebem a estrutura definida da face, já com os pontos delimitadores, através dela é possível definir, por

meio de um cálculo matemático, a provável região a qual se encontra os olhos ou a boca. Estes valores proporcionais são armazenados na estrutura *Rect*, uma estrutura que forma uma área retangular e passados para a função *cvSetImageROI()* a qual define a região de interesse em uma área da imagem e através dela, o reconhecedor concentra sua busca na área definida, economizando recursos de processamento e diminuindo o tempo de busca.

Figura 28 - Estrutura de detecção dos olhos

```

63 void detectEyes(IplImage* iFature, CvRect* arPos, CvMemStorage* aStorage, int aimg_no){
64
65     const char* EYE_CASCADE = "haarcascade_eyes.xml";
66
67     CvSeq          *csEyes;
68     CvHaarClassifierCascade *lcEyes;
69
70     lcEyes = (CvHaarClassifierCascade*)cvLoad(EYE_CASCADE, 0, 0, 0);
71
72     Rect rEyes(arPos->x,
73              arPos->y + (arPos->height/5.5),
74              arPos->width,
75              arPos->height/3.0);
76
77     cvSetImageROI(iFature, rEyes);
78
79     csEyes = cvHaarDetectObjects(iFature,
80                                lcEyes,
81                                aStorage,
82                                1.05, 3, 0,
83                                cvSize(25, 15));
84
85     for( int i = 0; i < (csEyes ? csEyes->total : 0); i++ )
86     {
87         CvRect *crEye = (CvRect*)cvGetSeqElem(csEyes, i);
88
89         cvRectangle(iFature,
90                   cvPoint(crEye->x, crEye->y),
91                   cvPoint(crEye->x + crEye->width, crEye->y + crEye->height),
92                   CV_RGB(0, 0, 255),
93                   2, 8, 0 );
94
95         addlog(aimg_no, "EYE", csEyes->total, 0);
96     }
97
98     cvReleaseHaarClassifierCascade( &lcEyes );
99 }

```

Fonte: do Autor.

Para chegar a estes cálculos de proporção foi utilizado o conceito de Proporção Áurea ou a chamada Regra de Ouro de Vitruvius, a qual descreve que o homem possui um rosto dividido horizontalmente em três partes simétricas: uma que vai desde o início do cabelo até as sobrancelhas, outra desde as sobrancelhas até o acúleo do nariz, e a terceira que vai do acúleo do nariz até o queixo. Segundo esta proporção, a largura do rosto deve medir dois terços de seu comprimento. A distância entre os olhos deve corresponder, no plano horizontal, exatamente à largura do

diâmetro nasal (CAMARGO; MENDONCA; DUARTE, 2009). Partindo deste princípio chegou-se nas equações (5) e (6) a fim de determinar a altura desta região e a posição de início da área desejada:

(5)

$$h = \frac{h}{3}$$

(6)

$$y = y + \frac{h}{c}$$

Sendo h a altura do rosto na imagem, que de acordo com o princípio da proporção áurea, qualquer área de interesse em um rosto corresponde a $1/3$ da altura do rosto, e que y que representa o ponto onde inicia a área da figura no plano cartesiano y , a área do ponto inicial de interesse, ou seja, a região dos olhos ou da boca estaria entre o valor do próprio y somado à altura (h) dividido por uma constante (c). Os valores de x e a largura da imagem permanecem as mesmas. No caso dos olhos esta constante atingiu o valor de 5.5 e a da boca um percentual de 68% em relação ao y .

Ao longo do desenvolvimento desta etapa foram feitos alguns testes alterando o número de vizinhos (de 0 á 5) e a escala de pixels da procura (de 0.5 á 5), assim como alterar o modelo de *harr-cascades*, até chegar aos valores utilizados.

Quando todas as funções foram ajustadas, deu-se início a etapa de treinamento, esta etapa durou 7 dias, aos quais foram expostas as imagens de treinamento para o detector 200 vezes. O conjunto positivo possuía 4.342 exemplos, já o negativo contava com 2.035 exemplos. Para cada conjunto de imagens, positivas e negativas, o detector gerava um arquivo .log (figura 29) com informações relativas as imagens obtidas, como número do frame, o tipo e quantidade de características detectadas e o tipo de atributo corrente, além do tempo gasto em cada extração. Essas informações refletem o que foi encontrado de características nos frames de teste dos dois conjuntos.

Figura 29 - Exemplo da imagem de .log

```

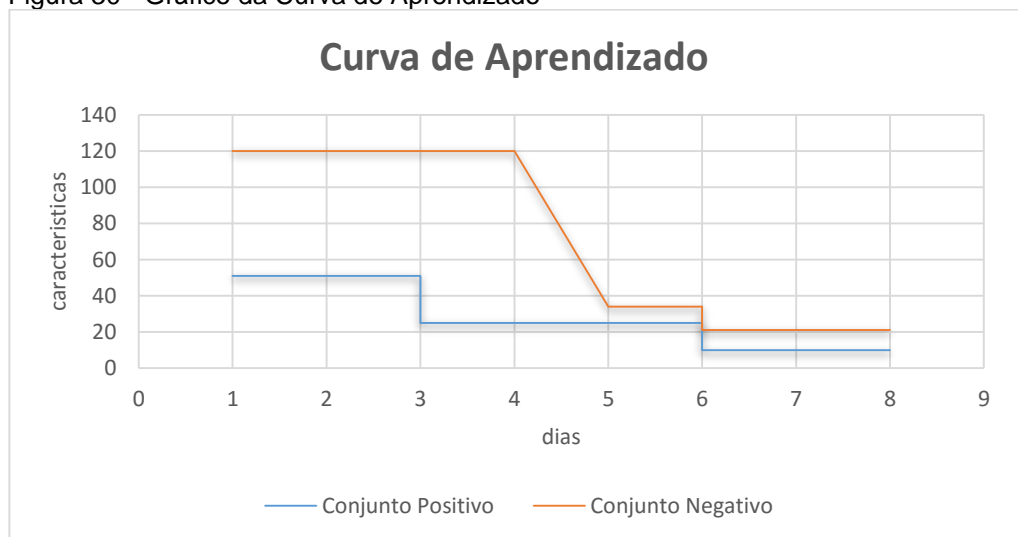
trainamento_66.log x
tempo de treinamento: 1444s
-----
imagem;tipo_caract;nmTotal;nmCorrente;
1;FACE;1;0;
2;FACE;1;0;
5;FACE;1;0;
8;FACE;1;0;
11;FACE;1;0;
13;FACE;1;0;
15;FACE;1;0;
16;FACE;1;0;
17;FACE;1;0;
18;FACE;1;0;
19;FACE;1;0;
20;FACE;1;0;
21;FACE;1;0;
22;FACE;1;0;

```

Fonte: do Autor.

A tendência é que quanto mais treinamento a *harr-cascades* receber, mais rápido e mais efetivo o detector ficará, na figura 30 é demonstrada a curva de aprendizado do detector construída por meio da compilação de informações obtidas por meio do log do sistema.

Figura 30 - Gráfico da Curva de Aprendizado



Fonte: do Autor.

Conforme a figura 30 pode-se observar que apesar do intervalo pequeno em que as *harr-cascades* foram treinadas, já é nítido o efeito do treinamento nas mesmas. Com a finalização do processo de treinamento dos *harr-cascades* pode-se

submeter o algoritmo ao conjunto de teste, afim de se validar a sua eficiência. Com esses dados coletados deu-se início à implementação da rede neural artificial.

5.3.3 Aplicação e desenvolvimento da rede neural artificial MLP

Após concluir-se as etapas do algoritmo de Viola-Jones, foi adicionada uma RNA do tipo MLP ao protótipo com o intuito de contornar os problemas de iluminação e performance descritos por Duarte et al (2009), Naderi et al (2012, tradução nossa) e Roy e Bhattacharjee (2016, tradução nossa). O *framework* OpenCV conta com uma biblioteca voltada para aprendizado de máquina a *ml*, nela são disponibilizadas as funções de ativação *IDENTITY*, *SIGMOID_SYM* e *GAUSSIAN*, e o algoritmo de treinamento *backpropagation*, juntamente com uma de suas variações: o algoritmo *Rprop*⁹.

Para criar a arquitetura da rede (figura 31), utilizou-se a estrutura *ANN_MLP*, disponibilizada na biblioteca *ml* do *framework* OpenCV, e determinou-se o número de camadas, a função de ativação e o tipo de treinamento por meio de suas funções. Como MLP utilizam camadas intermediárias se faz necessário a definição de uma função de ativação, nos algoritmos de *backpropagation* é necessária uma função de ativação contínua, diferenciável e não decrescente (FACELI et al, 2015). A função do tipo sigmoide definida no *framework* através da função *SIGMOID_SYM* obedece a esses requisitos, sendo a empregada nesta pesquisa.

⁹ *Resilient backpropagation* ou *Rprop* é um algoritmo de aprendizagem derivado de uma variação do algoritmo de *backpropagation* que elimina a necessidade da função de ativação (RIEDMILLER E BRAUN, 1994).

Figura 31 - Criação da RNA com o *framework* OpenCV

```

62
63 cv::Ptr<cv::ml::ANN_MLP> mlp;
64
65 cv::Mat layers = cv::Mat(4, 1, CV_32SC1);
66 layers.row(0) = cv::Scalar(4);
67 layers.row(1) = cv::Scalar(88);
68 layers.row(2) = cv::Scalar(30);
69 layers.row(3) = cv::Scalar(1);
70
71 mlp->setLayerSizes(layers);
72
73 mlp->setActivationFunction(ml::ANN_MLP::SIGMOID_SYM);
74
75 mlp->setTermCriteria(criteria);
76
77 mlp->setTrainMethod(ml::ANN_MLP::BACKPROP);
78
79 mlp->setBackpropMomentumScale(0.05f);
80
81 mlp->setBackpropWeightScale(0.05f);
82
83 mlp = cv::ml::ANN_MLP::create();
84

```

Fonte: do Autor.

Para Faceli et al (2015) escolher a arquitetura de uma MLP não é uma tarefa simples e normalmente é realizada por um processo de tentativa e erro, neste processo cada arquitetura é investigada e avaliada de acordo com sua acurácia para o conjunto de dados de treinamento. Para definir a arquitetura da rede utilizou-se uma abordagem construtiva, inserindo novos neurônios e conexões em uma RNA contendo uma topologia simples, formada por uma camada de entrada, uma camada oculta e uma camada de saída, a fim de melhorar seu desempenho. Na tabela 3 é apresentada uma tabela com as topologias utilizadas a fim de encontrar a melhor arquitetura.

Tabela 3 – Arquiteturas utilizadas na construção da topologia da RNA

Topologia	Entrada	Camada Oculta		Saída	Taxa de Acerto
		Camada 1	Camada 2		
Topologia 1	4	76	X	1	50%
Topologia 2	4	15	17	1	55%
Topologia 3	4	15	20	1	60%
Topologia 4	4	100	99	1	45%
Topologia 5	4	32	128	1	44%
Topologia 6	4	88	30	1	75%
Topologia 7	8	143	200	1	15%

Fonte: do Autor.

Iniciou-se os testes da rede com uma topologia simples, contendo apenas uma camada, no entanto esta topologia não diferenciou imagens classificadas incorretamente, foi então feita uma outra topologia, agora com duas camadas. Esta segunda topologia teve uma boa taxa de acerto referente ao grupo de imagens

positivo, porém ainda não foi capaz de diferenciar imagens negativas. Ao longo do treinamento foi aumentado o número de neurônios, sem ter uma grande variação de resultado. Foi percebendo-se que se uma das camadas tivesse mais que 100 neurônios, a rede classificava erroneamente o conjunto de imagens que obtiveram uma boa classificação na aplicação do algoritmo de Viola-Jones, entretanto os resultados incorretos a rede classificava corretamente. Observando-se os resultados obtidos no treinamento, a rede de Topologia 6 apresentada na tabela teve um melhor desempenho na classificação geral, pois foi a topologia que rendeu o melhor modelo de resultados nos dois conjuntos.

Para Rezende (2005) o aprendizado de uma RNA está associada a habilidade das mesmas adaptarem seus parâmetros como decorrência de sua iteração com o meio externo. O critério de desempenho que determina a qualidade do modelo e os critérios de parada são pré-estabelecidos pelos parâmetros de treinamento. Os critérios da rede são definidos por meio da estrutura *CvTermCriteria* (figura 32), nesta estrutura é definida a quantidade de iterações máximas da rede por meio do parâmetro *max_iter*, a precisão desejada para que o algoritmo pare pelo parâmetro *epsilon*, e o tipo de critério adotado definido no parâmetro *type*.

Figura 32 - Parâmetros da RNA

```

60
61 CvTermCriteria criteria;
62
63 criteria.max_iter = 200;
64
65 criteria.epsilon = 0.0000001f;
66
67 criteria.type = CV_TERMCRIT_ITER | CV_TERMCRIT_EPS;
68
69 mlp->setTermCriteria(criteria);
70

```

Fonte: do Autor.

Foi encontrada uma vasta literatura acerca do hibridismo entre VJ e RNA, entretanto as mesmas não descreveram satisfatoriamente a maneira a qual a rede foi alimentada na etapa de treinamento. Em estudos feitos por Fernandes e Gonzaga (2007) e Oliveira (2006) utiliza-se a avaliação dos *pixels* de uma imagem para fazer o treinamento e utilizar como entrada na rede. Partindo do princípio de que o algoritmo de Viola-Jones utiliza as características contidas na imagem, optou-se por utilizar estas características no treinamento. No OpenCV o treinamento da rede se dá pela

função *train()* (figura 33), esta função recebe a estrutura *Ptr<TrainData>* contendo um *array* de pontos que serão utilizados no treinamento. A estrutura *TrainData* recebe a matriz de dados de entrada da rede, a constante *ROW_SAMPLE* indicando que os valores a serem considerados no treinamento são os valores de cada linha do *array*, e uma matriz contendo o resultado de cada um dos dados, no caso se existe uma face ou não nos pontos da linha indicados na matriz. Após o treinamento, é iniciado a avaliação de cada linha da matriz, esta avaliação é feita por meio da função *predict()* que recebe uma matriz contendo o valor da linha corrente, e a estrutura matemática *Mat* vazia, essa estrutura recebe o valor da resposta da rede referente a característica corrente.

Figura 33 - Treinamento da RNA

```

82
83 Ptr<ml::TrainData> trainData = ml::TrainData::create(matInputLayers,
84                                                     ml::ROW_SAMPLE,
85                                                     matTrainLabels);
86
87 mlp->train(trainData);
88
89 for (int i = 0; i < matInputLayers.rows; i++){
90     Mat sample = Mat(1, matInputLayers.cols, CV_32F, matInputLayers.at<int>(i));
91     Mat result;
92     mlp->predict(sample, result);
93     predicted.at<float>(i,0) = result.at<float>(0,0);
94     addlogNN(sample, result);
95 }
96
97
98
99
100
101

```

Fonte: do Autor.

Conforme descrito anteriormente o treinamento foi iniciado com uma rede contendo uma arquitetura de baixa complexidade, foi utilizada uma topologia que continha 4 neurônios na camada de entrada, 10 neurônios na camada oculta e 1 neurônio na camada de saída. Esta configuração de rede foi submetida a um pequeno grupo de imagens, a fim de validar o código, o arquivo .log e avaliar se era possível realizar o treinamento no Raspberry Pi, com essa configuração validada, aumentou-se a complexidade da rede para 4 neurônios de camada de entrada, 15 neurônios na camada oculta e um neurônio na camada de saída e submeteu-se a rede o conjunto total de 6.377 imagens de treinamento. Nesta etapa pode-se observar que a rede não foi treinada, foram realizados mais alguns testes modificando a arquitetura, parâmetros e critérios da rede, entretanto o treino não foi efetuado.

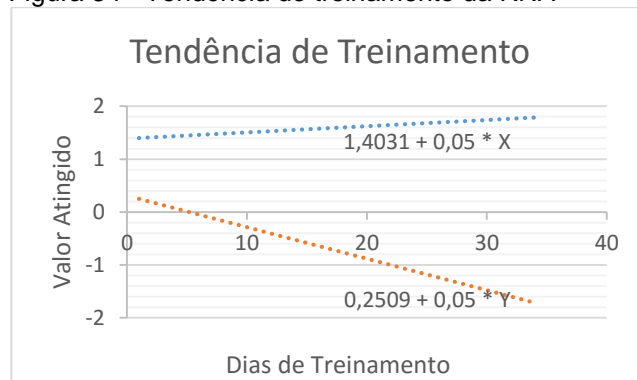
Embora autores como Janard e Maruringsith (2015, tradução nossa) e Hajari e Andurkar (2015, tradução nossa) relatem em seus estudos a aplicação e treinamento de reconhedores faciais em um Raspberry Pi, em nenhum momento é mencionado que a rede foi de fato treinada neste *hardware*. Para avaliar a possibilidade de que o problema consistia na falta de recursos da placa, o software do protótipo foi submetido a um treinamento em um PC com 8 Gb RAM, 650Gb de HD e processador Intel i5, onde foi comprovada a teoria de que os problemas relatados na etapa de treinamento da RNA no Raspberry Pi ocorreram por falta de recursos no hardware da placa.

Segundo Faceli et al (2009) se as taxas de erro para os dados de treinamento e de validação forem exibidas em um gráfico, poderá ser observado que no início do treinamento as duas taxas tendem a subir e que em um dado momento, a taxa de erro de validação pode começar a cair, sendo um indicio de que a rede parou de aprender e está ocorrendo *overfitting*, ou seja a rede está se tornando extremamente adaptada aos dados de treinamento devendo, neste ponto o treinamento da rede deve ser finalizado. Nesta aplicação as taxas de erro alcançaram um valor 1.4031 para uma imagem contendo faces e de 0.2509 para uma imagem que não apresentava faces, pois em decorrência dos problemas citados teve-se um tempo menor do que o esperado para o treinamento. Na documentação do *framework* OpenCV as taxas de erro obtidas para o algoritmo *backpropagation* para as entradas esperadas, no caso a presença de uma face, seriam de 1.7159 e de -1.7159 para entradas que não contivessem o objeto de estudo.

Considerando o pouco tempo que se teve para treinar a rede (aproximadamente quatro dias), observando o padrão de aprendizado, foi feita uma previsão através do método de tendência, que calcula uma previsão de longo prazo, a partir de uma taxa indicativa de movimento. Através do cálculo de tendência, foi criado um gráfico do tempo a qual a rede levaria para ser treinada (figura 34), tanto relacionado ao grupo positivo, quanto ao grupo negativo, neste cálculo foi considerando uma taxa de aprendizado de 0,05 por dia, ou seja a rede apresentaria uma evolução no reconhecimento de 5% ao dia, o qual levaria em torno de 28 dias para a rede atingir o resultado ótimo em relação as amostras positivas e 34 dias para

atingir um resultado ótimo para amostras negativas, entretanto não temos como afirmar a veracidade destas informações, pois algoritmos de aprendizado, aprendem de uma forma geométrica, ou seja, não possui um padrão linear de aprendizado. No entanto, segundo Faceli et al (2015) uma das desvantagens do algoritmo *backpropagation* é sua lentidão na convergência para um bom conjunto de pesos e a sua queda de desempenho quando usado em grandes conjuntos de dados, pois pode ser necessário apresentar o conjunto de treinamento centenas ou milhares de vezes, o que demonstra, novamente que o tempo necessário para treinar a rede pode ser muito superior ao imaginado inicialmente.

Figura 34 - Tendência de treinamento da RNA



Fonte: do Autor.

Depois de parado o treinamento da rede, pode-se fazer a coleta de dados de teste, estes resultados são apresentados a seguir.

5.3 ANÁLISE DOS RESULTADOS OBTIDOS

Após a aplicação de qualquer técnica, para demonstrar sua efetividade geralmente são feitos experimentos a fim de validar a sua eficácia. Estes experimentos geralmente ocorrem por meio da aplicação de modelos estatísticos como especificidade, sensibilidade e acurácia calculada sobre as taxas de erro. No conjunto de testes foram utilizadas 2.143 imagens positivas e 1.000 imagens negativas demonstradas na figura 35. Estas imagens foram escolhidas aleatoriamente do conjunto de dados obtidos para treinamento.

Figura 35 – Amostra de Faces reconhecidas



Fonte: do Autor.

Para avaliação de dois métodos normalmente é montada uma matriz de confusão (tabela 4), nesta matriz uma das classes é positiva e a outra negativa. Para a avaliação do reconhecedor a classe positiva é representada pelo número de imagens classificadas corretamente do conjunto de teste positivo, chamada de Verdadeiros Positivos (VP), e a classe negativa representada pelo número de imagens classificadas corretamente no conjunto de teste negativo, chamada de Verdadeiros Negativos (VN). Dentre estas imagens diz-se que são Falsos Positivos (FP), imagens que pertencem ao conjunto negativo e são classificadas como pertencentes a classe positiva e Falsos Negativos (FN), imagens que pertencem ao conjunto positivo, porém foram classificadas como pertencentes ao conjunto negativo.

Tabela 4 - Matriz de Confusão de Viola-Jones

	Grupo Positivo	Grupo Negativo
Classificação Positiva	2100	19
Classificação Negativa	35	981

Fonte: do Autor.

A partir da construção da matriz de confusão, outras medidas de desempenho podem ser obtidas para determinar a efetividade do teste realizado, como a Sensibilidade (S) (FACELI et al, 2015). A sensibilidade de um detector é a proporção de verdadeiros positivos entre todas as amostras e expressa a capacidade

do detector para encontrar faces, esta medida é estimada através da fórmula matemática (7) (MEDRONHO et al, 2009):

$$S = \frac{VP}{VP + FN} \quad (7)$$

No detector desenvolvido neste trabalho a taxa de sensibilidade, ou seja, a taxa de acerto geral entre todos os VP atingiu o percentual de $S = 0,99103$, quando um teste obtém uma taxa alta de sensibilidade observasse que o mesmo sofre perdas no indicador de Especificidade (E). O teste de especificidade retorna à probabilidade de uma detecção de um membro do conjunto negativo, ou seja, avalia a probabilidade de uma imagem sem uma face, ser classificada corretamente (MEDRONHO et al, 2009). Este teste é feito a partir do cálculo matemático (8):

$$E = \frac{VN}{FP + VN} \quad (8)$$

A taxa de especificidade do detector demonstrou um percentual de $E = 0,981$ demonstrando que o detector desenvolvido tem uma propensão maior de reconhecer imagens positivas do que negativas. Um teste ideal seria aquele que sua especificidade e sua sensibilidade atingissem 100%, ou seja, não existisse erro no resultado, entretanto na prática isto é impossível (MEDRONHO et al, 2009).

Por fim é comparada a acurácia (A) do detector, aplicada a um intervalo de confiança de modo que se possa permitir quantificar a proporção de acertos com uma margem de erro. A acurácia é a proporção de acertos de um teste como um todo, ou seja, é a proporção entre os VP e VN em relação a todos os resultados possíveis (MEDRONHO et al, 2009), a acurácia é determinada pela equação (9):

$$A = \frac{VP + VN}{VP + FP + VN + FN} \quad (9)$$

A acurácia do detector desenvolvido é de 78,72% ($A = 0.78726$) aplicando o intervalo de confiança padrão de 95%, a acurácia do detector varia entre 73,12% e 83,12%. Em trabalhos (tabela 5) semelhantes observou-se que a taxa máxima da acurácia é de 88% e a mínima de 80%.

Tabela 5 - Acurácia de sistemas semelhantes

Autor	Acurácia	Quantidade de Imagens Treinadas
Zomer (2005)	80%	75
Fernandez et al (2014)	88,06%	1050
Fernandez e Gonzaga (2007)	80%	200
Alqudah et al (2015)	86.23%	126
Minatto (2016)	78,72%	9.518

Fonte: do Autor.

Uma série de fatores pode-se influenciar na taxa de acurácia como por exemplo a quantidade de imagens treinadas e o tempo de duração para este treinamento (FACELI et al, 2015). Embora a taxa de acurácia do detector tenha sido um pouco menor que a de outros trabalhos relacionados, é importante ressaltar que todos os conjuntos teste analisados possuem uma quantidade de amostras que varia entre 75 a 1.050, podendo ser observado que a quantidade de amostras para o treino e teste era consideravelmente superior. Outro ponto a ser levado em consideração, é o tempo ao qual foram treinados os sistemas, muitos dos autores não deixam claro a quantidade de tempo despendida para a realização do treinamento dos classificadores, entretanto é surpreendente o quão rápido alguns algoritmos classificadores aprendem, como no caso das *haar-cascades* do algoritmo Viola-Jones, e como outros precisam de mais tempo, como as RNA.

7 CONCLUSÃO

A aplicação de reconhecimento de padrões é uma área em constante evolução, as melhores técnicas de processamento de imagens, a fim de se obter características com uma maior qualidade são constantemente discutidas pela comunidade científica. Mais especificamente a área de reconhecimento facial ainda necessita de um maior aperfeiçoamento, pois existem muitos obstáculos que esta tecnologia precisa superar, no entanto atualmente a área de visão computacional ganha cada vez mais pesquisadores, e a tendência é que cada vez apareçam algoritmos mais eficientes.

O algoritmo de Viola-Jones estudado nesta pesquisa provou ser o exemplo disto, sendo publicado em 2000 atualmente é o algoritmo que conta com mais aplicações reais. Sendo um dos algoritmos mais eficientes em termos de desempenho e processamento, o algoritmo de Viola-Jones não necessita de um grande poder computacional para ser aplicado, contudo a eficiência do algoritmo fica ligada a um bom desempenho do processamento das imagens a serem analisadas, caso as técnicas aplicadas nesta imagem não retornem uma boa quantidade de características a eficácia do algoritmo fica severamente comprometida. Sendo uma aplicação de alta sensibilidade, algumas falhas como efeitos de luminosidade, ruído, distorção e generalização de padrões necessitam ser estudadas mais profundamente.

A aplicação desenvolvida nesta pesquisa apresentou uma boa taxa de acurácia considerando o tempo de treinamento limitado e a grande quantidade de amostras treinadas, observou-se uma tendência da aplicação em classificar com uma melhor taxa de acerto imagens que não contenham faces.

Com o intuito de contornar os problemas de iluminação e performance foi aplicada uma RNA em conjunto com o algoritmo de Viola-Jones, entretanto durante a fase de testes foi constatado que o Raspberry Pi não possuía poder computacional para o treinamento de uma RNA. Em testes feitos com um *hardware* mais robusto o software desenvolvido executou com sucesso o treinamento da rede, infelizmente pelo tempo limitado do desenvolvimento não foi possível estudar uma forma de contornar

o problema. Desta maneira, pode-se dizer que os objetivos, geral e específicos do trabalho proposto foram parcialmente atingidos.

Como sugestão de continuação para este trabalho de pesquisa propõe-se:

- a) um estudo mais aprofundado de hardware levando em consideração a necessidade de processamento de uma RNA;
- b) efetuar o treinamento da RNA, coletar e comparar os resultados com o detector;
- c) aprimorar o protótipo para que o mesmo possa ser utilizado em ambientes reais;
- d) implementar um identificador por meio das características obtidas, a fim de capacitar o reconhecer a definir a identidade de uma pessoa.

REFERÊNCIAS

ASHBOURN, J. **Biometrics: Advanced Identity Verification - The Complete Guide**. New York. Springer-Verlag, 2002.

ASHBOURN, J. **Biometrics in the New World: The Cloud, Mobile Technology and Pervasive Identity**. 2. ed. New York. Springer, 2014. 258 p.

BARRETO, J. M. **Inteligência Artificial no Limiar do Século XXI**. 4. ed. Florianópolis: Duplic, 2002. 392 p.

BRADSKI, G.; KAEHLER, A. **Learning OpenCV computer vision with the OPENCV Library**. New York. O'Reilly Media, 2008. 555 p.

BRAGA, L. F. Z. **Sistemas de Reconhecimento Facial**. 2013. 84 pg. TCC (Graduação) - Curso de Sistemas de Informação, Universidade de São Paulo, São Carlos, 2013.

BRAHMBHATT, S. **Practical OpenCV**. New York. Apress, 2013. 244 p.

BRUNELLI, R. **Template Matching Techniques in Computer Vision: Theory and Practice**. New York. Wiley, 2009. 348 p.

CAMARGOS, C. N; MENDONCA, C. A; DUARTE, S. M. **Da imagem visual do rosto humano: simetria, textura e padrão**. Saude soc. 2009, vol.18, n.3, pp.395-410.

COLIC, A.; MARQUES, O.; FURHT, B. **Driver Drowsiness Detection: Systems and Solutions**. New York. Springer, 2014.

COPPIN, B. **Inteligência Artificial**. Rio de Janeiro: LTC, 2012. 636 p.

CHEN, T. et al. **Grayscale Image Matting And Colorization**. PROCEEDINGS OF ASIAN CONFERENCE ON COMPUTER VISION. Korea. 2004. p. 1164-1169.

DAVIES, E. R. **Computer and Machine Vision: Theory, Algorithms, Practicalities**. 4. ed. New York. Academic Press, 2012. 912 p.

DAWSON-HOWE, K. **A Practical Introduction to Computer Vision with OpenCV**. New York. Wiley, 2014. 234 p.

DELIGIANNIDIS, L.; ARABNIA, H. R. **Emerging Trends in Image Processing, Computer Vision and Pattern Recognition**. New York. Morgan Kaufmann, 2014. 640 p.

DICKINSON, S.; PIZLO, S. **Shape Perception in Human and Computer Vision:**

An Interdisciplinary Perspective. New York. London: Springer, v. 1, 2013. 502 p.

FARINELLA, M.; BATTIATO, S.; CIPOLLA, A. **Advanced Topics in Computer Vision.** 3. ed. New York. Springer, 2013. 433 p.

FERNANDES, R. A. S.; GONZAGA, A. **Reconhecimento de Faces Utilizando Redes Neurais Artificiais com Fusão por Meio de Votação.** III Workshop de Visão Computacional. São Paulo: SBC. 2007. p. 296-300.

FACELI, K; LORENA, A. C; GAMA, J; CARVALHO, A. **Inteligência Artificial. Uma Abordagem de Aprendizado de Máquina.** São Paulo: LTC. 2015. 394 pg.

FERNANDEZ, C; GOB, K.J.E; LEONIDAS, A.R.M; RAVARA, R.J.J; BANDALA, A.A; DADIOS, E.P. **Simultaneous face detection and recognition using Viola-Jones Algorithm and Artificial Neural Networks for identity verification.** Region 10 Symposium, 2014 IEEE, IEEE, 2014, pp. 672-676

GAVRILOVA, M. L.; MONWAR, M. **Multimodal Biometrics and Intelligent Image Processing for Security Systems.** New York. IGI Global, 2013. 232 p.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de Imagens Digitais.** São Paulo. Edgard Blücher Ltda, 2000.

GOOCH, A. et al. **Color2Gray: Saliency-preserving color removal.** ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2005, New York, 24, 3 Junho 2005. 634-639.

GRAUPE, D. **Principles of Artificial Neural Networks.** 3. ed. New York. World Scientific Publishing Company, 2013. 500 p.

HAYKIN, S. **Redes Neurais - Princípios e Práticas.** 2. ed. Porto Alegre: Bookman, 2001.

HEGG, R.; LUONGO, J. **Elementos de Biometria Humana.** 2ª. ed. São Paulo: Livraria Nobel, v. 1, 1975. 168 p.

HWU, W. **GPU Computing Gems Emerald Edition.** 1ª. ed. New York. Morgan Kaufmann, 2011. 886 p.

JAIN, A. K.; PRABHAKAR, S; PANKANTI, S. **On the similarity of identical twin fingerprints.** Pattern Recognition Society, 2002. p. 2653–2663.

JAIN, A. K; BOLLE, R; PANKANTI, S. **Biometrics: Personal Identification in Networked Society.** Norwell, Massachusetts: Kluwer Academic Publishers, 1999. 411 pp. Second printing 1999.

JANARD , K; MARURNGSITH,W. **Accelerating real-time face detection on a raspberry pi telepresence robot**. Innovative Computing Technology (INTECH), 2015 Fifth International Conference on, Galcia, 2015, pp. 136-141.

JAIN, S.; VAIBHAV, A.; GOYAL, L. **Raspberry Pi based Interactive Home Automation System through E-mail**. International Conference on Reliability, Optimization and Information Technology. India: IEEE. 2014. p. 277-280.

KLETTE, R. **Concise Computer Vision: An Introduction into Theory and Algorithms**. 1^a. ed. New York. Springer, v. 1, 2014. 429 p.

KOUROUPETROGLOU, G. **Assistive Technologies and Computer Access for Motor Disabilities**. 1^a. ed. New York. IGI Global, 2013. 417 p.

KRIG, S. **Computer Vision Metrics: Survey, Taxonomy, and Analysis**. 1^a. ed. Apress, 2014. 510 p.

LANIER, L. **Professional Digital Compositing: Essential Tools and Techniques**. 1^a. ed. Sybex, v. 1, 2009. 480 p.

LIENHART, R; KURANOV, A; PISAREVSKY, V. **Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection**. MRL Technical Report, Intel Labs, 2002.

LIU, H. **Face Detection and Recognition on Mobile Devices**. 1^a. ed. Morgan Kaufmann, 2014. 440 p.

LUGER, G. **Inteligência Artificial**. 6^a. ed. São Paulo: Pearson, v. 1, 2013. 632 p.

MALEPATI, H. **Digital Media Processing: DSP Algorithms Using C**. 1^a. ed. Newnes, v. 1, 2010. 768 p.

MARQUES FILHO, O.; VIEIRA NETO, H. **Processamento Digital de Imagens**. 1^a ed. Rio de Janeiro: Brasport, 1999. 408 p.

MODI, S. K. **Biometrics in Identity Management: Concepts to Applications**. 1^a. ed. New York. Artech House, 2011. 250 p.

MONTABONE, S. **Beginning Digital Image Processing: Using Free Tools for Photographers**. 2^a ed. New York. Apress, v. 1, 2009. 312 p.

NANAVATI, S.; THIEME, M.; NANAVATI, R. **Biometrics: Identity Verification in a Networked World**. 1^a ed. New York. Wiley, 2002. 320 p.

NAPPI, M.; TISTARELLI, M.; MARISCO, M. D. **Face Recognition in Adverse Conditions**. 1^a ed. New York. IGI Global, 2014. 480 p.

NADERI, S; NASIRI, J. A; CHARKARI, N. M; KABIR, E. **Visual illumination compensation for face images using light mapping matrix.** *IET Image Processing*, vol. 7, no. 5, pp. 514-522, July 2013.

NUNES, F. T. **Técnicas de biometria baseadas em padrões faciais e sua utilização na segurança pública.** Pós-Graduação da Universidade Federal de Santa Catarina. Araranguá. 2015.65 p.

OBERMEIER, B; PADOVA, T. **Photoshop Elements 7 All-in-One For Dummies.** 1ª ed. New York. John Wiley & Sons, 2009. 672 p.

OLIVEIRA, D. R. **Reconhecimento de Faces Usando Redes Neurais e Biometria.** Instituto Nacional de Pesquisas Espaciais. São José dos Campos, p. 166. 2006.

OPENCV. **OpenCV - Open Source Computer Vision.** Disponível em: <<http://opencv.org/>>. Acesso em: 21 set. 2015.

PALLAVI, V; ASHWINI, H; ANDURKAR, G. **Review Paper on System for Voice and Facial Recognition using Raspberry Pi.** International Journal of Advanced Research in Computer and Communication Engineering. Vol. 4, 2015.

RASBPERRYPI.ORG. Raspberry PI. Disponível em: <<https://www.raspberrypi.org/>>. Acesso em: 06 Novembro 2015.

RUSSELL, S.; NORVIG, P. **Inteligência Artificial: Uma Abordagem Moderna.** Rio de Janeiro: Elsevier, 2013. 988 p.

ROY, H; BHATTACHARJEE, D. **Local-Gravity-Face (LG-face) for Illumination-Invariant and Heterogeneous Face Recognition.** IEEE Transactions on Information Forensics and Security, vol. 11, no. 7, pp. 1412-1424, July 2016.

RIEDMILLER, H. BRAUN, A. **A direct adaptive method for faster backpropagation learning: the RPROP algorithm.** IEEE International Conference on Neural Networks, IEEE Press, New York, 1994, pp. 586–591.

SERASA Experian. Serasa Experian, 2016. Disponível em: <<http://noticias.serasaexperian.com.br/janeiro-registra-mais-de-150-mil-tentativas-de-fraude-aponta-serasa/>>. Acesso em: 12 Abril 2015.

SRIVASTAVA, R.; SINGH, S. K.; SHUKLA, K. K. **Research Developments in Biometrics and Video Processing Techniques.** IGI Global, v. 1ª, 2013. 279 p.

SUNG, K; POGGIO, T. **Exemple-Based Learning for View-Based Human Face Detection.** IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 20, p. 39-51, Janeiro 1998.

TRAORE, I.; AHMED, A. **A Continuous Authentication Using Biometrics: Data, Models and Metrics**. 1ª ed. IGI Global, 2011. 294 p.

TRASLAVIÑA, C. M. G. **Introducción a la biometría**. Disponível em: <<http://www.monografias.com/trabajos43/biometria/biometria.shtml>> acesso 12 maio 2016.

TSO, F. P. et al. **The Glasgow Raspberry Pi Cloud: A Scale Model for Cloud Computing Infrastructures**. First International Workshop on Resource Management of Cloud Computing. Philadelphia: IEEE. 2013. p. 108-112.

THEODORIDIS, S. KOUTROUMBAS, K. **Pattern Recognition**. New York. 2 ed. Elsevier, 2003.

VIANA, O. D; PEREIRA, T. E. **Análise da Influência de Diferentes Condições de Iluminação na Classificação de Faces Humanas em Imagens Digitais**. Revista De Tecnologia Da Informação E Comunicação, Vol. 3, Número 2, Porto Alegre, 2013. p9-19.

VIGLIAZZI, D. **Biometria - Medidas de Segurança**. 2ª ed. Florianópolis. Visual Books, 2006.

VIOLA, P.; JONES, M. **Rapid Object Detection using a Boosted Cascade of Simple Features**. IEEE Computer Society Conference on Computer Vision Recognition. Cambridge: IEEE. 2001. p. 511-518.

VIOLA, P.; JONES, M. **Robust Real-Time Face Detection**. International Journal of Computer Vision, Netherlands, IEEE. 2004. p. 137-154.

YANG, M.; ROTH, D.; AHUJA, N. A SNoW-Based Face Detector. Department of Computer Science and the Backman Institute, Urbana. 2000.

ZOMER, E. **BIOID – Aplicação Experimental de Identificação Biométrica Baseada em impressão digital**. 106p. TCC (Graduação) – Curso de Ciência da Computação – Universidade do Extremo Sul Catarinense, Criciúma, 2005.

APÊNDICE(S)

APÊNDICE A – Artigo

Análise de Características em Redes Neurais Artificiais Aplicadas em Visão Computacional com Ênfase em Reconhecimento Facial

Daniele V. Minatto¹, Sergio Coral¹, Merisandra C. de Mattos Garcia¹

¹Curso de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC)
Criciúma – SC – Brasil

daniele.minatto@gmail.com, sergiocoral@unesc.net, mem@unesc.net

***Abstract.** With the increasing application of recognition technology in daily life, both in areas such as security authentication, facial recognition techniques have been studied to be one of the least invasive technologies and the only one that does not require the user cooperation. However, for the face is a highly deformable element are still studied more effective techniques to identify a face discarding external elements such as image quality, brightness, occlusion and guidance. Thus, this article explores the result of a facial recognition prototype based on a study on the Viola - Jones algorithm and the inclusion of an artificial neural network in an integrated hardware, Raspberry PI*

***Resumo.** Com a crescente aplicação da tecnologia de reconhecimento no cotidiano das pessoas, tanto em áreas de autenticação como segurança, técnicas de reconhecimento facial vem sendo estudadas por ser uma das tecnologias menos invasivas e a única que não necessita da cooperação do usuário. Entretanto, por a face ser um elemento altamente deformável ainda são estudadas técnicas mais eficazes de identificar uma face descartando elementos externos como: qualidade da imagem, luminosidade, oclusão e orientação. Desta forma, este artigo explora o resultado de um protótipo de reconhecimento facial baseado em um estudo feito sobre o algoritmo de Viola-Jones e a inclusão de uma rede neural artificial em um hardware integrado, o Raspberry PI.*

1. Introdução

Em janeiro de 2016 foram registradas 150.643 tentativas de roubo de identidade de acordo com o Indicador' Serasa Experian (2016), o equivalente a uma tentativa de fraude a cada 17,8 segundos no país. Diante desse panorama torna-se claro necessário o uso de tecnologias para autenticação de indivíduos. O procedimento de identificação facial é algo natural para o ser humano, pois todos os dias, reconhecem pessoas, aprende-se características novas e armazenam-se novas faces no cérebro, no entanto, pesquisas referentes a reconhecimento facial são relativamente recentes, este tema começou a despertar interesse no final dos anos 80, Sung e Poggio (1998) propuseram um detector de faces a base de distribuição, Rowley et al (1998) relataram um detector de faces com base em redes neurais artificiais inspirados em Féraud et al (1997) que desenvolveu um detector baseado em um novo modelo de rede neural artificial que foi nomeado de Constrained Generative Model (CGM), Viola e Jones (2001) propuseram o

primeiro método de detecção de faces em tempo real, utilizando um conjunto de características simples em vez de extrair informações dos pixels da imagem.

Aplicações de reconhecimento facial requerem um alto nível de sensibilidade para a detecção e comparação de características, desta maneira além de padronizar as imagens que servirão de entrada para o sistema torna-se imprescindível o uso de técnicas como as das redes neurais artificiais (RNA) para que o sistema possa aprender quais recursos são mais eficazes, comparando características informadas com a base de dados e modificando os pesos/valores de cada característica a fim de determinar se existe uma correspondência (HAYKIN, 2001). A abordagem adotada nesta pesquisa é a implementação de um protótipo em um hardware integrado, do algoritmo Viola-Jones, fazendo o uso de uma rede neural artificial do tipo *Multilayer Perceptron* (MLP) utilizando o algoritmo backpropagation para aprimorar o desempenho da detecção em condições difíceis. A Seção 2 descreve os componentes necessários para a execução do projeto.

2. Construção do Protótipo de Reconhecimento Facial

Na literatura sobre o algoritmo de Viola-Jones são frequentes os problemas relatados acerca da luminosidade ideal para o reconhecimento de uma face. Segundo Pereira e Viana (2013) a eliminação de informações redundantes devido a problemas de iluminação é fundamental para a obtenção de boas taxas de classificação. O desenvolvimento do protótipo analisado aqui foi focado para capacitá-lo a analisar uma determinada cena retirada de uma foto ou vídeo em ambientes onde a luminosidade não seja a ideal. Para a execução do protótipo foi utilizado um hardware integrado, o Raspberry Pi.

2.1. Estrutura Física

O Raspberry Pi 2 Modelo B, o qual foi utilizado neste protótipo, é a segunda geração Raspberry Pi. Possui um processador ARMv7, podendo executar sistemas operacionais como o Ubuntu, Raspbian, Windows 10 IoT, entre outras distribuições personalizadas (Raspberrypi.org, 2016, tradução nossa). Durante a elaboração do projeto foi pesquisado qual Sistema Operacional (SO) seria o mais eficiente em termos de desempenho, quanto praticidade. Foram feitos testes nas distribuições de SO Windows 10 IoT, Ubuntu e Raspbian, entretanto foi escolhido o Raspbian por se tratar de um sistema operacional livre baseado em Debian otimizado para o hardware Raspberry Pi. Foi adicionada no Raspberry, via conexão USB uma Webcam de 1,3 Megapixels, com a inserção desta Webcam na placa, foi criada uma conexão de vídeo em tempo real, permitindo que a placa obtivesse imagens para serem analisadas pelo *software* desenvolvido com o framework OpenCV.

2.2. OpenCV

O OpenCV é o mais conhecido *framework* para o desenvolvimento de aplicações utilizando visão computacional e possui várias extensões (BRADSKI; KAEHLER, 2008, tradução nossa). A biblioteca tem mais de 2500 algoritmos otimizados, à qual inclui um conjunto abrangente de algoritmos de aprendizado clássicos e especializados em visão computacional. Esses algoritmos podem ser usados para detectar e reconhecer rostos, identificar objetos, entre outros (Opencv.org, 2016, tradução nossa). O *framework* OpenCV foi utilizado tanto no

desenvolvimento do algoritmo Viola-Jones, quanto na rede neural artificial. Á seguir é detalhado a base de testes utilizada para o treinamento e avaliação da acurácia do *software* desenvolvido neste protótipo.

2.3. Base de Dados

Para o treinamento do Viola-Jones e da RNA foi utilizado o banco de faces do Dr. Libor Spacek¹⁰, Mestre em inteligência computacional e pesquisador da área de visão computacional, disponibilizado de forma gratuita na internet, este banco conta com 7.900 imagens de indivíduos de sexo, etnias e idades variadas, contendo a presença de óculos, barba e objetos no fundo, e são divididas em baixa, média e alta complexidade de detecção. A base de dados de imagens negativas utilizada foi obtida por meio da página de Naotoshi Seo¹¹ em sua página sobre treinamento de cascatas de classificadores. Esta base contém 4979 imagens cinza com resolução 640x480 pixels, disponibilizadas de forma gratuita e que não contém faces de pessoas e foram obtidas em diversos ambientes e condições de iluminação. Algumas destas imagens estavam corrompidas e foram descartadas da base.

Ao final estes dois conjuntos de imagens somaram 9.518 imagens as quais foram subdivididas em um conjunto de teste e em um conjunto de treino (figura 1), para realizar este particionamento foi usado o método de Holdout (THEODORIDIS; KOUTROUMBAS, 2003, tradução nossa), utilizando 2/3 da base de dados inicial como conjunto de testes, o qual contava com 3.140 imagens e 1/3 para o conjunto de treino contando com 6.378 imagens, após essa definição foi dado início ao desenvolvimento do protótipo, descrita na próxima seção.

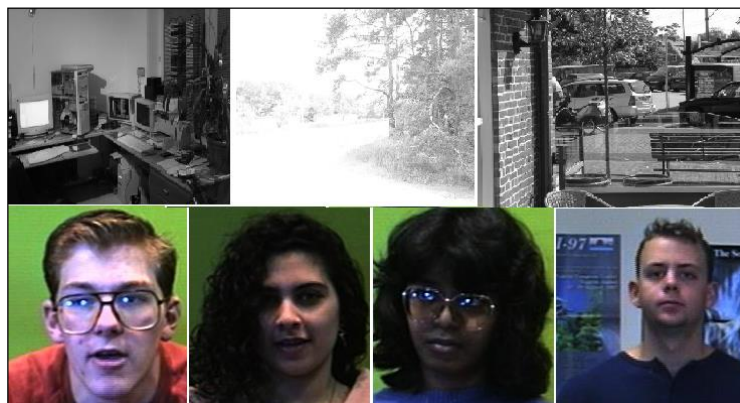


Figura 1. Base de Testes Positivas e Negativas

3. Desenvolvimento do Protótipo

O desenvolvimento do protótipo foi em quatro etapas (figura 2) com duas fases de treinamento. A primeira etapa é referente a aquisição de imagem que pode ser obtida por meio de uma Webcam para reconhecimento em tempo real. A segunda etapa é o pré-processamento da imagem obtida, onde são feitas as transformações de imagem necessárias a fim de normalizá-la. Na terceira etapa é aplicado o algoritmo Viola-Jones na imagem, antes de dar

¹⁰ Disponível em: <http://cswww.essex.ac.uk/mv/allfaces/>

¹¹ Disponível em: <http://note.sonots.com/>

prosseguimento ao desenvolvimento foi realizado o treinamento das harr-cascades da face, dos olhos e da boca, e recolhidos os dados do conjunto de testes.

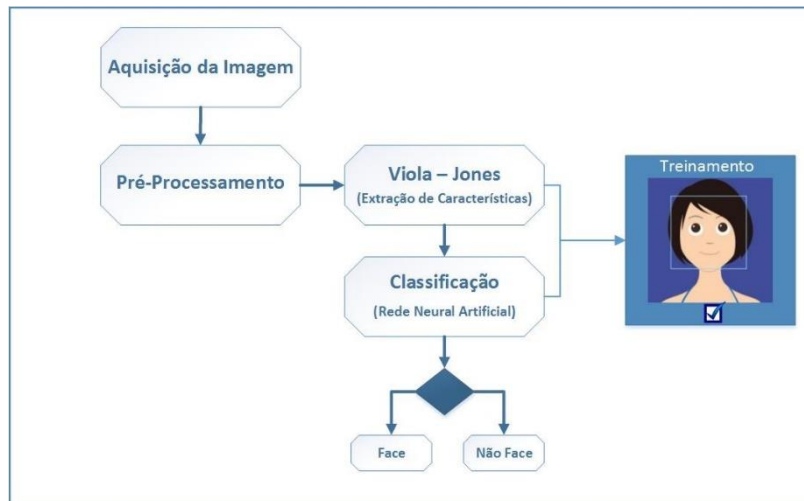


Figura 2. Etapas do Sistema

Na quarta etapa foi desenvolvido uma RNA e interligada ao detector de Viola-Jones, nas etapas de treinamento e testes da rede usou-se os mesmos conjuntos aplicados anteriormente.

Antes de iniciar a aplicação do algoritmo se faz necessário preparar a imagem, para que o processo de extração de características seja otimizado, o *framework* OpenCV conta com uma biblioteca específica para o tratamento de imagens: a *imgproc*. Esta biblioteca possibilita transformações de alta complexidade bem como mais básicas, foi aplicado por meio do *framework* as seguintes transformações: escala de tons de cinza, realce de imagem, remoção de ruído e segmentação.

Depois desta etapa concluída, a imagem já está apta para fornecer dados das características de interesse, no caso faces, por meio da aplicação do algoritmo Viola-Jones.

3.1. Desenvolvimento do algoritmo Viola-Jones

Para desenvolver o algoritmo de Viola-Jones, utilizou-se o *framework* OpenCV, que já possui uma implementação do algoritmo englobando todos os seus conceitos chave: a extração das *features*, a definição da imagem integral, o filtro *harr-cascades* e o algoritmo Adaboost.

O grande diferencial do algoritmo é que através dele é possível definir, após a localização da face, por meio de um cálculo matemático, a provável região a qual se encontra os olhos ou a boca. Estes valores proporcionais são armazenados em uma estrutura que forma uma área retangular e passados para o *framework*, por meio dela é definida a região de interesse em uma área da imagem e através dela, o reconhecedor concentra sua busca nesta área, economizando recursos de processamento e diminuindo o tempo de busca. Ao longo do desenvolvimento desta etapa foram feitos alguns testes alterando o número de vizinhos (de 0 a 5) e a escala de pixels da procura (de 0.5 a 5), assim como alterar o modelo de harr-cascades, até chegar aos valores utilizados. Quando todas as funções foram ajustadas, deu-se início a etapa de treinamento, a qual pode ser analisada na Seção 4, e posteriormente deu-se início a implementação da rede neural artificial.

3.2. Aplicação da rede neural artificial Multilayer perceptron

Após concluir-se as etapas do algoritmo de Viola-Jones, foi adicionada uma rede neural artificial do tipo *multilayer perceptron* (MLP) ao protótipo com o intuito de contornar os problemas de iluminação e performance descritos por Duarte et al (2009), Naderi et al (2012, tradução nossa) e Roy e Bhattacharjee (2016, tradução nossa). O *framework* OpenCV conta com uma biblioteca voltada para aprendizado de máquina a *ml*, nela são disponibilizados as funções de ativação e o algoritmo de treinamento *backpropagation*, juntamente com uma de suas variações: o algoritmo *Rprop*¹².

Para criar a arquitetura da rede, utilizou-se a biblioteca *ml* do *framework* OpenCV, e determinou-se o número de camadas, a função de ativação e o tipo de treinamento por meio de suas funções. Como MLP utilizam camadas intermediárias se faz necessário a definição de uma função de ativação, nos algoritmos de *backpropagation* é necessária uma função de ativação contínua, diferenciável e não decrescente (FACELI et al, 2015). A função do tipo sigmoide obedece a esses requisitos, sendo a empregada na pesquisa. Para definir a arquitetura da rede utilizou-se uma abordagem construtiva, inserindo novos neurônios e conexões em uma RNA contendo uma topologia simples, formada por uma camada de entrada, uma camada oculta e uma camada de saída, a fim de melhorar seu desempenho. Iniciou-se os testes da rede com uma topologia simples, contendo apenas uma camada, no entanto esta topologia não diferenciou imagens classificadas incorretamente, ao longo dos testes foi aumentado o número de neurônios, sem ter uma grande variação de resultado. Foi percebendo-se que se uma das camadas tivesse mais que 100 neurônios, a rede classificava erroneamente o conjunto de imagens que obtiveram uma boa classificação na aplicação do algoritmo de Viola-Jones, entretanto os resultados incorretos a rede classificava corretamente.

4. Resultados Experimentais

Após a aplicação de qualquer técnica, para demonstrar sua efetividade geralmente são feitos experimentos a fim de validar a sua eficácia. Estes experimentos geralmente ocorrem por meio da aplicação de modelos estatísticos como especificidade, sensibilidade e acurácia calculada sobre as taxas de erro. No conjunto de testes foram utilizadas 2.143 imagens positivas e 1.000 imagens negativas demonstradas na figura 3.

¹² Resilient backpropagation ou Rprop é um algoritmo de aprendizagem derivado de uma variação do algoritmo de *backpropagation* que elimina a necessidade da função de ativação (RIEDMILLER E BRAUN, 1994).



Figura 3. Amostra de faces reconhecidas

A seguir são detalhadas as etapas de treinamento e teste do algoritmo Viola-Jones e da rede neural artificial do tipo MLP.

4.1. Análise dos resultados do algoritmo Viola-Jones

Quando todas as funções referentes ao algoritmo foram ajustadas, deu-se início a etapa de treinamento, esta etapa durou 7 dias, aos quais foram expostas as imagens de treinamento para o detector 200 vezes. O conjunto positivo possuía 4.342 exemplos, já o negativo contava com 2.035 exemplos. Para cada conjunto de imagens, positivas e negativas, o detector gerava um arquivo .log com informações relativas as imagens obtidas, como número do frame, o tipo e quantidade de características detectadas e o tipo de atributo corrente, além do tempo gasto em cada extração. Essas informações refletem o que foi encontrado de características nos frames de teste dos dois conjuntos. A tendência é que quanto mais treinamento a harr-cascades receber, mais rápido e mais efetivo o detector ficará, na figura 4 é demonstrada a curva de aprendizado do detector construída por meio da compilação de informações obtidas por meio do log do sistema.

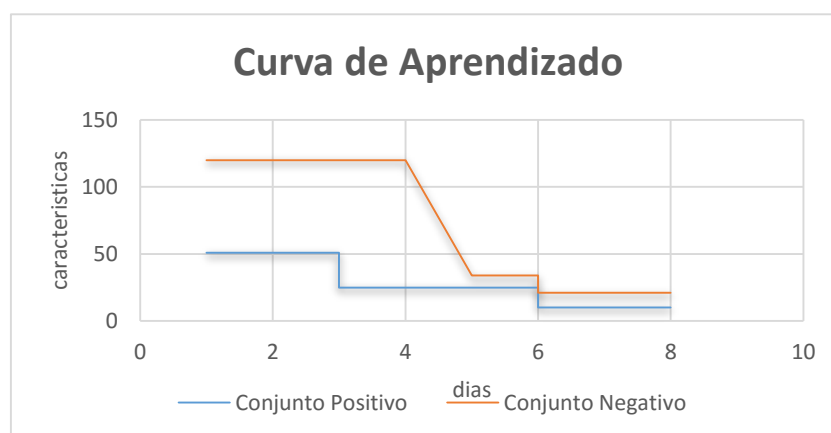


Figura 4. Gráfico da Curva de Aprendizado

Conforme a figura 4 pode-se observar que apesar do intervalo pequeno em que as *harr-cascades* foram treinadas, já é nítido o efeito do treinamento nas mesmas. Com a finalização do processo de treinamento dos *harr-cascades* pode-se submeter o algoritmo ao conjunto de teste, afim de se validar a sua eficiência. Com esses dados coletados deu-se início à coleta de resultados do conjunto de testes e a partir deles aplicar métodos estatísticos afim de comprovar sua efetividade. No detector desenvolvido neste trabalho a taxa de sensibilidade, ou seja, a taxa de acerto geral entre todos os VP atingiu o percentual de $S=0,99103$, quando um teste obtém uma taxa alta de sensibilidade observasse que o mesmo sofre perdas no indicador de Especificidade (E) o qual obteve um percentual de $E= 0,981$ demonstrando que o detector desenvolvido tem uma propensão maior de reconhecer imagens positivas do que negativas. Um teste ideal seria aquele que sua especificidade e sua sensibilidade atingissem 100%, ou seja, não existisse erro no resultado, entretanto na prática isto é impossível (MEDRONHO et al, 2009). Por fim é comparada a acurácia (A) do detector, aplicada a um intervalo de confiança de modo que se possa permitir quantificar a proporção de acertos com uma margem de erro, esta taxa obteve um percentual de 78,72% ($A=0.78726$) aplicando o intervalo de confiança padrão de 95%, a acurácia do detector varia entre 73,12% e 83,12%. Em trabalhos (tabela 2) semelhantes observou-se que a taxa máxima da acurácia é de 88% e a mínima de 80%.

Tabela 2. Acurácia de sistemas semelhantes

Autor	Acurácia	Quantidade de Imagens Treinadas
Zomer (2005)	80%	75
Fernandez et al (2014)	88,06%	1050
Fernandez e Gonzaga (2007)	80%	200
Alqudah et al (2015)	86.23%	126
Minatto (2016)	78,72%	9.518

Uma série de fatores pode-se influenciar na taxa de acurácia como por exemplo a quantidade de imagens treinadas e o tempo de duração para este treinamento (FACELI et al, 2015). Embora a taxa de acurácia do detector tenha sido um pouco menor que a de outros trabalhos relacionados, é importante ressaltar que todos os conjuntos teste analisados possuem uma quantidade de amostras que varia entre 75 a 1.050, podendo ser observado que a quantidade de amostras para o treino e teste era consideravelmente superior. Outro ponto a ser levado em consideração, é o tempo ao qual foram treinados os sistemas, muitos dos autores não deixam claro a quantidade de tempo despendida para a realização do treinamento dos classificadores, entretanto é surpreendente o quão rápido alguns algoritmos classificadores aprendem, como no caso das *harr-cascades* do algoritmo Viola-Jones, e como outros precisam de mais tempo, como as RNA.

4.1. Análise do desempenho da Rede Neural Artificial

Conforme descrito anteriormente na Seção 3, o treinamento foi iniciado com uma rede contendo uma arquitetura de baixa complexidade, foi utilizada uma topologia que continha 4 neurônios na camada de entrada, 10 neurônios na camada oculta e 1 neurônio na camada de saída. Esta configuração de rede foi submetida a um pequeno grupo de imagens, a fim de validar o código,

o arquivo .log e avaliar se era possível realizar o treinamento no Raspberry Pi, com essa configuração validada, aumentou-se a complexidade da rede para 4 neurônios de camada de entrada, 15 neurônios na camada oculta e um neurônio na camada de saída e submeteu-se a rede o conjunto total de 6.377 imagens de treinamento. Nesta etapa pode-se observar que a rede não foi treinada, foram realizados mais alguns testes modificando a arquitetura, parâmetros e critérios da rede, entretanto o treino não foi efetuado.

Embora autores como Janard e Marurngsith (2015, tradução nossa) e Hajari e Andurkar (2015, tradução nossa) relatem em seus estudos a aplicação e treinamento de reconhedores faciais em um Raspberry Pi, em nenhum momento é mencionado que a rede foi de fato treinada neste hardware. Para avaliar a possibilidade de que o problema consistia na falta de recursos da placa, o software do protótipo foi submetido a um treinamento em um PC com 8 Gb RAM, 650Gb de HD e processador Intel i5, onde foi comprovada a teoria de que os problemas relatados na etapa de treinamento da RNA no Raspberry Pi ocorreram por falta de recursos no hardware da placa.

Segundo Faceli et al (2009) se as taxas de erro para os dados de treinamento e de validação forem exibidas em um gráfico, poderá ser observado que no início do treinamento as duas taxas tendem a subir e que em um dado momento, a taxa de erro de validação pode começar a cair, sendo um indicio de que a rede parou de aprender e está ocorrendo *overfitting*, ou seja a rede está se tornando extremamente adaptada aos dados de treinamento devendo, neste ponto o treinamento da rede deve ser finalizado. Nesta aplicação as taxas de erro alcançaram um valor 1.4031 para uma imagem contendo faces e de 0.2509 para uma imagem que não apresentava faces, pois em decorrência dos problemas citados teve-se um tempo menor do que o esperado para o treinamento. Na documentação do framework OpenCV as taxas de erro obtidas para o algoritmo backpropagation para as entradas esperadas, no caso a presença de uma face, seriam de 1.7159 e de -1.7159 para entradas que não contivessem o objeto de estudo.

Considerando o pouco tempo que se teve para treinar a rede (aproximadamente quatro dias), observando o padrão de aprendizado, foi feita uma previsão através do método de tendência, que calcula uma previsão de longo prazo, a partir de uma taxa indicativa de movimento. Através do cálculo de tendência, foi criado um gráfico do tempo a qual a rede levaria para ser treinada (figura 6), tanto relacionado ao grupo positivo, quanto ao grupo negativo, neste cálculo foi considerando uma taxa de aprendizado de 0,05 por dia, ou seja a rede apresentaria uma evolução no reconhecimento de 5% ao dia, o qual levaria em torno de 28 dias para a rede atingir o resultado ótimo em relação as amostras positivas e 34 dias para atingir um resultado ótimo para amostras negativas, entretanto não temos como afirmar a veracidade destas informações, pois algoritmos de aprendizado, aprendem de uma forma geométrica, ou seja, não possui um padrão linear de aprendizado.

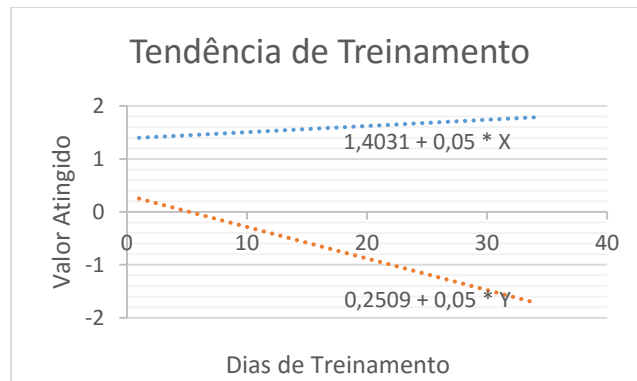


Figura 6. Tendência de treinamento da RNA

No entanto, segundo Faceli et al (2015) uma das desvantagens do algoritmo backpropagation é sua lentidão na convergência para um bom conjunto de pesos e a sua queda de desempenho quando usado em grandes conjuntos de dados, pois pode ser necessário apresentar o conjunto de treinamento centenas ou milhares de vezes, o que demonstra, novamente que o tempo necessário para treinar a rede pode ser muito superior ao imaginado inicialmente.

6. Conclusão

A aplicação de reconhecimento de padrões é uma área em constante evolução, e as melhores técnicas de processamento de imagens, a fim de se obter características com uma maior qualidade são constantemente discutidas pela comunidade científica. Sendo um dos algoritmos mais eficientes em termos de desempenho e processamento, o algoritmo de Viola-Jones não necessita de um grande poder computacional para ser aplicado, contudo a eficiência do algoritmo fica ligada à um bom desempenho do processamento das imagens a serem analisadas, caso as técnicas aplicadas nesta imagem não retornem uma boa quantidade de características a eficácia do algoritmo fica severamente comprometida. A aplicação desenvolvida nesta pesquisa apresentou uma boa taxa de acurácia considerando o tempo de treinamento limitado e a quantidade de amostras treinadas. Com o intuito de contornar os problemas de iluminação foi aplicada uma RNA em conjunto com o algoritmo de Viola-Jones, entretanto durante a fase de testes foi constatado que o Raspberry Pi não possuía poder computacional para o treinamento de uma RNA, entretanto em testes feitos com um hardware mais robusto o software desenvolvido executou com sucesso o treinamento da rede.

Referencias

- BRADSKI, G.; KAEHLER, A. Learning OpenCV computer vision with the OPENCV Library. New York. O'Reilly Media, 2008. 555 p.
- CAMARGOS, C. N; MENDONCA, C. A; DUARTE, S. M. Da imagem visual do rosto humano: simetria, textura e padrão. Saude soc. 2009, vol.18, n.3, pp.395-410.
- FACELI, K; LORENA, A. C; GAMA, J; CARVALHO, A. Inteligência Artificial. Uma Abordagem de Aprendizado de Máquina. São Paulo: LTC. 2015. 394 pg.
- HAYKIN, S. Redes Neurais - Princípios e Práticas. 2. ed. Porto Alegre: Bookman, 2001.

- NADERI, S; NASIRI, J. A; CHARKARI, N. M; KABIR, E. Visual illumination compensation for face images using light mapping matrix. IET Image Processing, vol. 7, no. 5, pp. 514-522, July 2013.
- OPENCV. OpenCV - Open Source Computer Vision. Disponível em: <<http://opencv.org/>>. Acesso em: 21 set. 2015.
- RASBPERRYPI.ORG. Raspberry PI. Disponível em: <<https://www.raspberrypi.org/>>. Acesso em: 06 Novembro 2015.
- ROY, H; BHATTACHARJEE, D. Local-Gravity-Face (LG-face) for Illumination-Invariant and Heterogeneous Face Recognition. IEEE Transactions on Information Forensics and Security, vol. 11, no. 7, pp. 1412-1424, July 2016.
- SERASA Experian. Serasa Experian, 2016. Disponível em: <<http://noticias.serasaexperian.com.br/janeiro-registra-mais-de-150-mil-tentativas-de-fraude-aponta-serasa/>>. Acesso em: 12 Abril 2015.
- SUNG, K; POGGIO, T. Exemple-Based Learning for View-Based Human Face Detection. IEEE Transactions on Pattern Analysis and Machine Inteligence, v. 20, p. 39-51, Janeiro 1998.
- THEODORIDIS, S. KOUTROUMBAS, K. Pattern Recognition. New York. 2 ed. Elsevier, 2003.
- VIANA, O. D; PEREIRA, T. E. Análise da Influência de Diferentes Condições de Iluminação na Classificação de Faces Humanas em Imagens Digitais. Revista De Tecnologia Da Informação E Comunicação, Vol. 3, Número 2, Porto Alegre, 2013. p9-19.
- VIOLA, P.; JONES, M. Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE Computer Society Conference on Computer Vision Recognition. Cambridge: IEEE. 2001. p. 511-518.
- VIOLA, P.; JONES, M. Robust Real-Time Face Detection. International Journal of Computer Vision, Netherlands, IEEE. 2004. p. 137-154.