

UNIVERSIDADE FEDERAL DE SANTA CATARINA

Elder Dominghini Tramontin

**ANÁLISE E APLICAÇÃO DE RECONHECIMENTO
FACIAL EM SISTEMA EMBARCADO**

Araranguá

2016

Elder Dominghini Tramontin

**ANÁLISE E APLICAÇÃO DE RECONHECIMENTO
FACIAL EM SISTEMA EMBARCADO**

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina, como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Engenharia de Computação.

Orientador: Prof. Fabrício de Oliveira Ourique, Dr.

Araranguá

2016

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Tramontin, Elder Dominghini

Análise e aplicação de reconhecimento facial em sistema embarcado / Elder Dominghini Tramontin ; orientador, Fabrício de Oliveira Ourique - Araranguá, SC, 2016.
77 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Campus Araranguá.
Graduação em Engenharia de Computação.

Inclui referências

1. Engenharia de Computação. 2. Reconhecimento facial embarcado. 3. Análise de Componentes Principais. 4. Autofaces. I. Ourique, Fabrício de Oliveira. II. Universidade Federal de Santa Catarina. Graduação em Engenharia de Computação. III. Título.

Elder Dominghini Tramontin

ANÁLISE E APLICAÇÃO DE RECONHECIMENTO FACIAL EM SISTEMA EMBARCADO

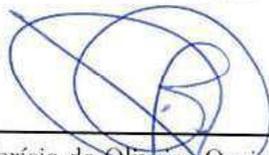
Este Trabalho de Conclusão de Curso foi julgado aprovado para a obtenção do Título de “Bacharel em Engenharia de Computação”, e aprovado em sua forma final pela Universidade Federal de Santa Catarina.

Araranguá, 22 de julho 2016.



Prof. Anderson Luiz Fernandes Perez, Dr.
Coordenador do curso

Banca Examinadora:



Prof. Fabrício de Oliveira Ourique, Dr.
Presidente



Prof. Anderson Luiz Fernandes Perez, Dr.



Prof. Mauricio Girardi, Dr.

À minha família, por ser a base que dá
suporte ao meu crescimento.

AGRADECIMENTOS

À Universidade Federal de Santa Catarina, pelas oportunidades que me foram concedidas e à toda a comunidade do campus Araranguá, que se tornou como uma família durante estes bons anos. Ao Professor e orientador Fabrício de Oliveira Ourique, pelos conhecimentos compartilhados, pelas orientações, pela paciência em relação aos meus métodos de trabalho e por ser um exemplo a ser seguido. Ao Professor Anderson Luiz Fernandes Perez por todas as oportunidades de crescimento por ele oferecidas, pela confiança em mim, tantas vezes depositada, e por todos os ensinamentos e conselhos, sobre o curso e, principalmente, sobre a vida. Aos caríssimos amigos do Laboratório de Automação e Robótica Móvel, que muito colaboraram no meu aprendizado e foram imprescindíveis para a conclusão desta etapa da minha vida acadêmica. À Ana Claudia Fernandes de Souza, pelo apoio, motivação e compreensão. À meus pais, Adeodato Tramontin e Maria Inês Dominghini Tramontin e ao meu irmão, Eduardo Dominghini Tramontin, por sempre me darem subsistência para realizar meus estudos e, especialmente, pela compreensão durante a realização deste trabalho. Aos demais colegas e amigos que tornaram esta jornada divertida e prazerosa.

Any particular face can be economically represented in terms of a best coordinate system that we term eigenpictures.

(Sirovich , 1987)

RESUMO

É desejável que a identificação de pessoas a partir de seus rostos seja realizada por máquinas. Isto permite uma gama de aplicações em áreas como biometria, segurança, controle de acesso, *Big Data*, etc. Porém, há muita complexidade na implementação de algoritmos para este fim. Dentre as várias abordagens existentes, a utilização da técnica PCA representa uma alternativa de simples implementação e bons resultados. Para que estes resultados sejam alcançados corretamente, faz-se necessária a utilização de técnicas de pré-processamento. Com o objetivo de imbuir sistemas de reconhecimento em diversos contextos, faz-se desejável o desenvolvimento de sistemas embarcados voltados a este fim. Este trabalho visa apresentar a técnica PCA para reconhecimento de faces, juntamente com os conceitos de processamento digital de imagens imprescindíveis para esta tarefa, e, no final, apresentar o desenvolvimento e avaliação de um sistema embarcado utilizando estes conhecimentos.

Palavras-chave: Reconhecimento facial embarcado. Análise de Componentes Principais. Autofaces.

ABSTRACT

It is desirable that the people identification from their faces be performed by machines. This allows a range of applications in areas such as biometrics, security, access control, context adaptation, Big Data, etc. However, there is much complexity in the algorithms implementation for this purpose. Among the various existing approaches, the PCA is alternative with simple implementation and good results. In order to these results be achieved properly, is necessary the use of preprocessing techniques. Aiming to imbue recognition systems in several contexts, it is desirable the development of embedded systems targeted to this end. This paper presents the PCA technique for face recognition using digital image processing concepts for this task. It presents the development and evaluation of an embedded system.

Keywords: Embedded face recognition. Eigenfaces. Principal Component Analysis.

LISTA DE FIGURAS

Figura 1	Aquisição e representação de imagens de fora do espectro de visível.....	30
Figura 2	Matriz e sua representação como imagem.....	31
Figura 3	Aplicação da Equalização de Histograma.....	33
Figura 4	Projeção de uma imagem com apenas 1 <i>pixel</i> em espaço unidimensional.....	36
Figura 5	Projeção de uma imagem com 2 <i>pixels</i> em plano 2D ...	37
Figura 6	Projeção de 3 imagens com 3 <i>pixels</i> cada em espaço 3D	37
Figura 7	Pontos em espaço 2D e linha que melhor captura a variância destes (ou que minimiza os erros).....	39
Figura 8	Autoespaço com 16 autofaces, em ordem decrescente de acordo com o autovalor associado.....	43
Figura 9	Reconstrução de duas faces a partir de uma fração das autofaces.....	44
Figura 10	Distâncias em relação ao autoespaço.....	47
Figura 11	Detecção e classificação de faces em tempo real.....	51
Figura 12	Exemplo de classificadores <i>Haar-like</i>	52
Figura 13	Detecção por minimização de DFFS.....	53
Figura 14	Interface gráfica do sistema desenvolvido.....	56
Figura 15	Exemplo de conversão de uma imagem no espaço de cores RGB para escala de cinza.....	57
Figura 16	Etapas comuns ao treinamento e ao reconhecimento ...	58
Figura 17	Curvas ROC do sistema (a) sem e (b) com equalização de histograma.....	62
Figura 18	Curvas de correspondência cumulativa (CMC) do sistema com (vermelho) e sem (azul) a etapa de equalização de histograma.....	64
Figura 19	Raspberry Pi modelo B.....	65

LISTA DE ABREVIATURAS E SIGLAS

Pixel	<i>Picture Element</i>
DIP	<i>Digital Image Processing</i>
VC	Visão Computacional
PCA	<i>Principal Component Analysis</i>
RF	Reconhecimento Facial
DFFS	<i>Distance From the Face Space</i>
DIFS	<i>Distance Within the Face Space</i>
ROC	<i>Receiver Operating Characteristic</i>
MATLAB	<i>MATrix LABoratory</i>
CMC	<i>Cumulative Match Characteristic</i>

LISTA DE EQUAÇÕES

3.1	Média amostral.....	39
3.2	Variâncias de cada amostra.....	39
3.3	Cálculo da matriz de covariância.....	39
3.4	Autovetores e autovalores da matriz de covariância, C	40
3.5	Autovetores e autovalores da matriz L	41
3.6	Rearranjo da Equação 3.5.....	41
3.7	Rearranjo da Equação 3.6.....	41
3.8	Cálculo dos autovetores de C por meio dos autovetores de L	41
3.9	Relevância do autovetor.....	42
3.10	Pontuações de uma face.....	42
3.11	Projeção de uma face no autoespaço.....	45
3.12	Distância em relação ao autoespaço.....	45
3.13	Distância da faces de prova a cada classe.....	45
3.14	Distância de Mahalanobis.....	47
3.15	DFFS Mahalanobis.....	48
3.16	DIFS Mahalanobis.....	48
3.17	Cálculo da Sensibilidade de um diagnóstico.....	49
3.18	Cálculo da Especificidade de um diagnóstico.....	49
3.19	Cálculo da Precisão de um diagnóstico.....	50
4.1	Decomposição das amostras da Galeria.....	59
4.2	Sensibilidade de um sistema de RF.....	61
4.3	Especificidade de um sistema de RF.....	61
4.4	Precisão de um sistema de RF.....	61

SUMÁRIO

1	INTRODUÇÃO	23
1.1	JUSTIFICATIVA E MOTIVAÇÃO	25
1.2	OBJETIVOS	27
1.2.1	Geral	27
1.2.2	Específicos	27
1.3	ORGANIZAÇÃO DO TRABALHO	28
2	VISÃO COMPUTACIONAL	29
2.1	IMAGEM	29
2.1.1	Imagem Analógica	29
2.1.2	Imagem Digital	30
2.2	PROCESSAMENTO DIGITAL DE IMAGENS	32
2.2.1	Equalização de Histograma	32
2.2.2	Segmentação	32
2.3	VISÃO COMPUTACIONAL	33
3	RECONHECIMENTO FACIAL	35
3.1	ANÁLISE DE DADOS	36
3.2	ANÁLISE DE COMPONENTES PRINCIPAIS - PCA ...	38
3.2.1	Redução do Espaço Dimensional	40
3.2.2	Decomposição em Autofaces	42
3.2.3	Classificação	43
3.3	BASES DE DADOS	48
3.4	DETECÇÃO DE FACE	50
3.4.1	Algoritmo de Viola-Jones	51
3.4.2	Detecção por Minimização da DFFS	53
4	SISTEMA EMBARCADO DE RECONHECIMENTO FACIAL	55
4.1	DESENVOLVIMENTO	55
4.2	VALIDAÇÃO	60
4.3	IMPLEMENTAÇÃO EM SISTEMA EMBARCADO	64
5	CONSIDERAÇÕES FINAIS	67
5.1	TRABALHOS FUTUROS	67
	REFERÊNCIAS	69

1 INTRODUÇÃO

Desde as últimas décadas o homem busca técnicas de aproximar cada vez mais as capacidades das máquinas às humanas. Com a Inteligência Artificial, visa simular aprendizado em tarefas específicas. Com a Robótica, dota os computadores de locomoção, capacidade de manipular objetos e interagir com o meio. Os sentidos humanos também são objetos de estudo, seja para a criação de robôs com estas aptidões ou para a correção de deficiências físicas em pessoas.

A visão é o mais avançado dos sentidos humanos (GONZALEZ; WOODS, 2002). O aprendizado é, em sua maioria, baseado em informações visuais. Esta afirmação pode ser validada ao se avaliar: quantas informações do dia-a-dia provém de estímulos visuais e quantas dos outros sentidos? Pense nos livros, relatórios, anotações e, enfim, todo o nosso sistema de escrita. Placas, gestos e expressões também são importantes fontes de dados. Desta forma, é possível concluir que dotar computadores da capacidade de visão significa viabilizar a aquisição e interpretação desta gama de informações.

Realizar a obtenção de imagens e traduzí-las em imagens virtuais é um processo chamado Conversão Analógico-Digital ou Aquisição de Imagem. O resultado deste processo é chamado de Imagem Digital. Da mesma forma que outras conversões analógico-digitais, esta etapa é de grande influência nas etapas subsequentes. Quanto maiores os níveis de detalhes desejados, maior a quantidade de bytes usados para representar a imagem. Portanto há um dilema e, dependendo de cada aplicação, deve-se balancear o quanto priorizam-se os detalhes em detrimento da capacidade de armazenamento e do custo para realizar o processamento.

Uma imagem pode ser interpretada como uma função bidimensional, relacionando a posição com a intensidade de luz aferida. Quando esta análise é realizada em espaço contínuo ela refere-se à imagem real e quando ela é feita em espaço discreto refere-se à imagem digital (GONZALEZ; WOODS, 2002). Esta última, por ser descontínua, possui um número finito de elementos que a compõem, denominados elementos de imagem, ou *picture elements*, ou ainda *pixels*.

Cada *pixel* possuirá um valor associado, cujo entendimento depende da origem da imagem. Em imagens adquiridas por câmeras que capturam somente o espectro de cores visível ao olho humano, estes valores corresponderão a uma intensidade de cor, variando da ausência total daquela cor até a presença total. Para valores discretos que repre-

sentem esta intensidade, esta faixa entre ausência e presença é dividida em níveis.

Na área de computação gráfica existem duas subdivisões que naturalmente se confundem: Visão Computacional e Processamento Digital de Imagens (do inglês *Digital Image Processing*, ou DIP). A Visão Computacional (VC), cujo objetivo é replicar a visão humana em máquinas, consiste em realizar operações sobre imagens e obter como resultado alguma informação. Ela é um ramo da Inteligência Artificial. Já o DIP consiste em realizar operações sobre imagens digitais de maneira a obter outras imagens como saída, sendo que estas últimas possuem, geralmente, alguma característica ressaltada para que seja mais facilmente perceptível aos humanos.

Com o crescimento do uso da tecnologia, há cada vez mais dados pessoais armazenados em computadores e servidores, e são necessários mecanismos de autenticação para impedir que pessoas indevidas tenham acesso a estas informações. Segundo Prazeres (2010), estas soluções podem ser divididas em 3 grupos: baseadas em conhecimentos (senhas), baseadas em propriedade (chips, tokens ou cartões) e baseadas em características (biometria). Em comparação às outras, as soluções de biometria são mais difíceis de se roubar, replicar ou perder e são carregadas naturalmente sem nenhum ônus adicional, como é o caso das senhas que devem ser memorizadas ou dos cartões que devem ser transportados consigo. Além disto, a presença do portador é imprescindível para a autenticação, o que evita fraudes.

Entre os diversos tipos de autenticação por biometria está o reconhecimento a partir de imagens (fotos) da face dos usuários. Existem várias abordagens para a realização da identificação destas faces: Redes Neurais Artificiais (PRAZERES, 2010), (JALILMAZLOUM et al., 2012), Lógica Nebulosa (KHANAM; SHAFIQ; AKRAM, 2008), Descritores e Momentos (SINGH; CHHABRA, 2014), Padrões Binários Locais (HUANG et al., 2011), Análise de Componentes Principais (*Principal Component Analysis* ou PCA, em inglês) (LEARNED-MILLER et al., 2015), (TURK; PENTLAND, 1991), (SOBRINHO, 2010), Análise de Discriminantes Lineares (MARTÍNEZ; KAK, 2001), (ZHAO et al., 2003), entre outras.

Na prática, as imagens contém mais componentes do que apenas rostos. Há todo um plano de fundo, podendo conter objetos, paredes, animais, fontes de luz e, inclusive, outras faces. Com toda esta complexa composição torna-se necessário isolar os objetos de interesse, neste caso, as faces, para realizar as operações somente com as regiões de interesse (HJELMÅS; LOW, 2001). Para realizar esta segmentação pode-se utilizar algoritmos de detecção facial.

É recomendado a adoção de algumas etapas de pré-processamento das imagens para realizar a normalização, em relação a luminosidade e tamanho da face, e a extração, por meio da segmentação, dos fragmentos da imagem que contenham faces. Diante de tantas técnicas, é comum a utilização de bancos de imagens para que os resultados nos diferentes trabalhos sejam comparados. Não há, atualmente, uma técnica que se sobressaia em todos os quesitos em relação às outras. Dependendo da metodologia utilizada para criar cada tipo de base de faces (quantidade de amostras por pessoa, diferentes orientações da cabeça, diferentes fontes de luz, etc) os resultados obtidos podem diferir, devendo o desenvolvedor utilizar a abordagem que melhor se adequa ao seu problema.

Neste trabalho será apresentada a formulação matemática da técnica PCA, bem como sua aplicação em sistemas de Reconhecimento Facial. Conforme Turk e Pentland (1991), Huang et al. (2011) e Zhao et al. (2003), há um aumento significativo na precisão do reconhecimento se forem realizadas normalizações nas imagens para compor o banco de faces. Portanto, este trabalho também busca explicar sobre alguns procedimentos de pré-processamento que podem ser realizados para este fim. Por fim, visando validar a plausibilidade dos conceitos abordados, há a implementação em sistema embarcado.

1.1 JUSTIFICATIVA E MOTIVAÇÃO

A Visão Computacional é uma área que busca replicar a visão humana em máquinas. Dentre as suas diversas aplicações, uma que está sendo bastante explorada ultimamente é o reconhecimento facial (RF). Diante de outras abordagens para autenticação de pessoas, como reconhecimento por voz, impressão digital, retina, DNA, entre outros (SOBRINHO, 2010), a utilização de faces tem a vantagem de não ser intrusiva, ou seja, o usuário não tem a necessidade de interagir com o sistema, inclusive, não é necessário nem o consentimento do mesmo (PRAZERES, 2010). Além disso, várias pessoas podem ser distinguidas ao mesmo tempo, diferentemente dos outros métodos.

Segundo Hjelmås e Low (2001), a detecção de face é considerada uma tarefa fácil para humanos, porém complexa de ser implementada computacionalmente. Esta etapa é crucial para a etapa de reconhecimento pois elimina informações desnecessárias para este fim. Dentre as diversas técnicas conhecidas, todas necessitam realizar a aquisição de amostras para compor uma base de dados, as quais devem ser subme-

tidas a um treinamento para possibilitar o reconhecimento. Contudo, pela quantidade de memória necessária para representar e manipular imagens, sistemas desta finalidade possuem grande complexidade e, portanto, requerem muito processamento.

Tratando-se de estado da arte em reconhecimento facial, não há como se referir a apenas uma técnica pois a variedade das aplicações não favorece a descoberta de uma solução ótima para todas. É preciso levar em conta a problemática para então realizar a escolha do método que atende requisitos como utilização de memória, taxa de acerto, tempo de processamento, etc.

Os trabalhos de Sirovich e Kirby (1987) e Turk e Pentland (1991) evidenciam que o uso da técnica PCA traz ganhos em armazenamento e processamento, possibilitando a criação de sistemas de reconhecimento em tempo real. Também é relatada uma abordagem para realizar a detecção de faces em imagens utilizando o mesmo conceito. Assim, detecção e autenticação podem ser feitos com um único algoritmo.

Dentre a vasta variedade de metodologias para reconhecimento facial, a escolha da técnica PCA é embasada na frequente utilização desta na bibliografia, sendo esta comumente utilizada como base de comparação. Além disto, há algumas outras abordagens similares que são baseadas na Análise de Componentes Principais que na teoria deveriam superar o desempenho da PCA mas na prática o fazem apenas sob certas restrições (MARTÍNEZ; KAK, 2001), fazendo com que ela ainda seja preferível para situações genéricas.

De acordo com Moghaddam e Pentland (1995), este tipo de abordagem (decomposição em subespaços) é particularmente recomendada para problemas de reconhecimento de objetos. No contexto de reconhecimento facial, a técnica PCA, assim como outros métodos estatísticos, obtém a informação global da imagem, sem utilizar-se de cálculos envolvendo a geometria das faces (SINGH; CHHABRA, 2014).

Com relação a implementação, PCA é relativamente simples de implementar e entender, diferentemente de métodos como Redes Neurais Artificiais ou Lógica Nebulosa, e por isto, permite um melhor estudo do seu procedimento. Isto possibilita realizar pequenas alterações de modo a avaliar comparativamente as suas implicações. Além disto, trata-se de um método de aprendizagem não-supervisionado, permitindo sua utilização (após a implementação do algoritmo) por leigos (MOGHADDAM; PENTLAND, 1995).

De acordo com Braga (2013), ainda que diversos trabalhos tenham sido desenvolvidos no campo de reconhecimento facial, até então nenhum possui eficiência equivalente ou superior ao ser humano. Isto é,

realizar o reconhecimento em ambientes distintos, de diferentes perspectivas e independente da expressão facial (BRAGA, 2013). Deste modo, pode-se afirmar que ainda há aprimoramentos a serem realizados na problemática de reconhecimento de faces.

Buscando seguir os conceitos de *Smart Cities* e *Smart Home*, há um desenvolvimento crescente de tecnologias com o objetivo de aumentar a integração entre humanos e dispositivos. Algumas possibilidades de aplicação de reconhecimento facial são: sistemas de segurança, sistemas de controle de acesso, veículos inteligentes, validação de passaportes e documentos, além das diversas áreas emergentes ainda não exploradas. Deste ponto de vista, é imprescindível a viabilização da identificação de pessoas de forma eficiente, segura e não-invasiva. Diante desta problemática, é desejável a utilização de um sistema embarcado que realize o reconhecimento. Isto significa desenvolver sistemas específicos, que possam ser aplicados em ambientes distintos, sem um custo muito elevado de adaptação. Desta forma, este trabalho visa encontrar uma solução de RF aplicada em um sistema embarcado.

1.2 OBJETIVOS

Esta seção apresenta o objetivo geral e os objetivos específicos deste trabalho.

1.2.1 Geral

Propor e avaliar um sistema embarcado de identificação por reconhecimento facial.

1.2.2 Específicos

1. Levantar o estado da arte a respeito de sistemas de reconhecimento facial;
2. Estudar técnicas de processamento de imagens digitais;
3. Levantar algoritmos para detecção de face;
4. Implementar e validar a técnica PCA;
5. Propor um sistema embarcado para identificar e reconhecer faces;

6. Implementar e validar o sistema proposto em 5.

1.3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está dividido em 5 capítulos, contando com a introdução.

O **Capítulo 2** apresenta alguns conceitos de Visão Computacional que servem de base para a compreensão do restante do documento.

O **Capítulo 3** explana sobre o reconhecimento facial, focando na técnica PCA e utilização de autofaces, dividido de acordo com suas etapas.

O **Capítulo 4** aborda o desenvolvimento e a implementação de um sistema embarcado de reconhecimento de faces.

O **Capítulo 5** apresenta as considerações finais deste trabalho e algumas propostas para trabalhos futuros.

2 VISÃO COMPUTACIONAL

Neste capítulo serão abordados alguns conceitos básicos do Processamento Digital de Imagens e da Visão Computacional. Estes são imprescindíveis para entender as implicações da etapa de pré-processamento em sistemas de reconhecimento. Segundo (HJELMÁS; LOW, 2001), uma face pode encontrar-se em um fundo complexo, sendo necessário a aplicação de algumas técnicas de PDI para empregar mais eficientemente os algoritmos de reconhecimento.

2.1 IMAGEM

É sabido que dos sentidos o mais importante é a visão. Uma cena pode conter muita informação de uma só vez. Alguns fatos que confirmam isto: a leitura é principal base para estudos, placas de trânsito e recursos visuais são fundamentais para o correto funcionamento do sistema de transporte e assistir a uma partida de futebol sem áudio provê uma melhor fidelidade com a realidade do que apenas ouvi-lo pelo rádio. É possível, portanto, concluir que uma imagem nada mais é do que uma forma de representação da informação.

Uma imagem pode conter diversos elementos que a compõem. O trabalho de distinguí-los é feito naturalmente pelo cérebro humano, porém para computadores é uma tarefa relativamente complexa. A capacidade de absorver estas informações é crucial em muitíssimas atividades do dia-a-dia das pessoas, por isso é desejável que sistemas computacionais também a possuam.

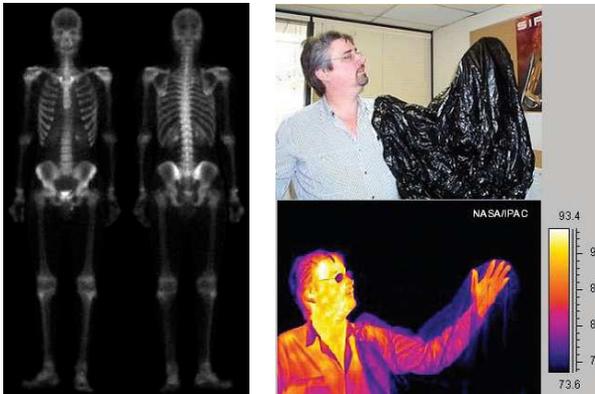
Após vislumbrar uma imagem, o cérebro a analisa de maneira a obter as características relevantes da mesma. Ou seja, é realizado um processamento da imagem de maneira a perceber e realizar a abstração das informações que realmente são úteis. Voltando ao exemplo do jogo de futebol, este é o processo realizado pelo locutor quando ele observa o jogo e narra pelo rádio somente uma síntese do que está sendo visualizado.

2.1.1 Imagem Analógica

Uma imagem, do ponto de vista físico, é a projeção dos fótons emitidos por diversos átomos à um observador. A energia do fóton é o

que determina a qual banda do espectro eletromagnético ele pertence. Em outras palavras, dependendo da energia do fóton, uma determinada cor é observada.

Os olhos humanos conseguem identificar somente uma pequena faixa de todo espectro eletromagnético, chamada de espectro visível ou, apenas, luz (CONCI; AZEVEDO; LETA, 2008). Porém, outras bandas também podem fornecer informações importantes (GONZALEZ; WOODS, 2002), como é o caso das imagens de Raio-X, como a Figura 1a, ultravioleta ou das imagens térmicas (banda infravermelha), como a Figura 1b. Estas imagens não podem ser diretamente analisadas por humanos. Uma alternativa é realizar sua aquisição por câmeras que captam esta banda de frequência e representá-las por cores do espectro visível.



(a) Aplicação típica da utilização da banda de Raio-X

(b) Imagem do espectro visível (cima) e da banda Infravermelha Termal (baixo)

Figura 1 – Aquisição e representação de imagens de fora do espectro de visível

Extraído de: (GONZALEZ; WOODS, 2002) e (NASA/IPAC, 2002).

2.1.2 Imagem Digital

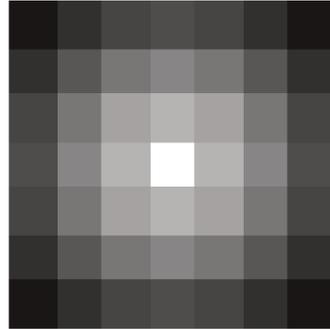
Uma imagem digital é uma estrutura de dados que armazena elementos pertencentes a uma imagem analógica (ou real), de modo a permitir sua reprodução no meio digital. Assim como outras transformações analógico-digitais, a quantidade de bits utilizados para a

representação das imagens está intimamente relacionada à fidelidade (semelhança) entre a imagem virtual e a real.

Dois conceitos relevantes na etapa de aquisição são a amostragem e a quantização. O primeiro se refere a quantidade de *pixels* e o segundo a quantidade de bits utilizados para representar o valor de intensidade de cada *pixel*. Em relação aos seus efeitos: com uma maior quantidade de bits para representar a intensidade de um *pixel*, maior será a precisão da informação digital em relação a analógica. Da mesma forma, quanto mais *pixels* uma imagem possuir, melhor será a sua resolução, ou seja, menor serão as discontinuidades entre *pixels* vizinhos, fazendo com que a imagem se pareça mais com a sua originária do espaço contínuo.

Para armazenar os valores de intensidade juntamente com seu relativo valor posicional na imagem, a estrutura mais adequada é a matriz. Isto porque ela permite guardar as intensidades de cada *pixel* em uma representação análoga ao espaço de 2 dimensões, como pode ser observado na Figura 2. Desta forma, é comum a utilização dos conceitos de Álgebra Linear para manipulação de imagens digitais.

27	50	73	83	73	50	27
50	94	136	155	136	94	50
73	136	199	225	199	136	73
83	155	225	255	225	155	83
73	136	199	225	199	136	73
50	94	136	155	136	94	50
27	50	73	83	73	50	27



(a) Valores de intensidade de uma imagem¹ 7×7 .

(b) Representação visual de (a) utilizando 8 bits por *pixel*.

Figura 2 Matriz e sua representação como imagem

Segundo Gonzalez e Woods (2002), a aquisição de dados é uma das etapas onde surgem as principais fontes de ruído. Por se tratar do estágio inicial do DIP, todas as fases seguintes estarão sujeitas a estes ruídos, o que pode gerar erros que se propagam exponencialmente nos resultados obtidos (SANTANA; GOMES; SANTOS, 2014). Por estes problemas, muitas vezes, serem difíceis de inibir, faz-se necessária a utilização de técnicas de pré-processamento, para, por exemplo, eliminação

¹Imagem de um Filtro Passa-baixa Gaussiano 2D com $\sigma = 2$

ou diminuição de ruídos ou realce de contraste.

2.2 PROCESSAMENTO DIGITAL DE IMAGENS

Processamento Digital de Imagens é a ação de realizar operações sobre uma imagem, cujo resultado é outra imagem. Estas transformações tem como objetivo ressaltar características de forma a facilitar a extração das informações desejadas, fornecendo subsídios para sua interpretação pelo analista humano (SOBRINHO, 2010). O DIP é comumente utilizado em conjunto com técnicas de Visão Computacional.

Nas Subseções a seguir serão tratadas dois tipos de processamento que afetam positivamente os resultados de sistemas de reconhecimento facial.

2.2.1 Equalização de Histograma

Histograma é uma ferramenta estatística que expõe, para cada nível de intensidade, a proporção dos *pixels* que a apresentam. Ou seja, é um gráfico da frequência de ocorrência de cada intensidade (CONCI; AZEVEDO; LETA, 2008). A análise de um histograma permite identificar atributos como nível de contraste e luminosidade média.

Realizar a equalização de um histograma permite distribuir, de forma mais uniforme, a intensidade dos *pixels* dentro de todo o intervalo de níveis possível. Esta técnica visa melhorar o contraste da imagem. As Figuras 3a e 3b ilustram o resultado da aplicação da equalização de histograma em uma imagem com baixo contraste e as Figuras 3c e 3d são seus respectivos histogramas.

2.2.2 Segmentação

O processo de segmentação consiste em agrupar e ressaltar *pixels* que tenham relação entre si, como é o caso daqueles pertencentes a um mesmo objeto ou região. A segmentação é uma etapa que pode ou não estar associada a extração da região de interesse.

No caso específico do tema abordado neste trabalho, a segmentação deve ocorrer na etapa de detecção de face. Fazendo isto, o resultado ideal da segmentação deve ser a face sem *background*. Porém, esta tarefa acaba sendo muito complexa, então na maioria dos casos é tolerável realizar apenas um recorte retangular onde a face esteja contida.

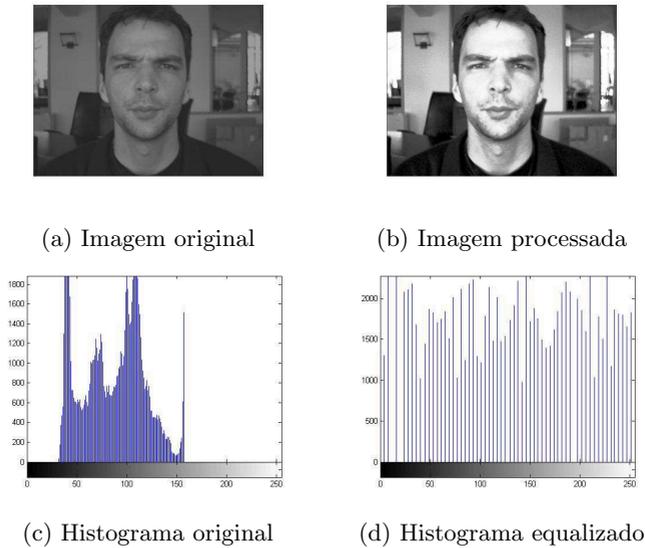


Figura 3 – Aplicação da Equalização de Histograma
 Extraído e adaptado de: (BIOID, 2001).

2.3 VISÃO COMPUTACIONAL

A Visão Computacional é uma área da Inteligência Artificial e tem como objetivo replicar a visão humana. De acordo com Hjelms e Low (2001), antigamente ela era aplicada em tarefas repetitivas e enfadonhas em linhas de montagem. Hoje em dia nesta área vêm sendo desenvolvidos trabalhos mais complexos, como o reconhecimento de padrões e a modelagem tridimensional.

Diferentemente do Processamento Digital, cujas entradas e saídas são imagens, a Visão Computacional possui como entrada uma ou mais imagens e produz como resultado algum tipo de informação. Alguns exemplos são: detectar a posição e a quantidade de determinado objeto (MOGHADDAM; PENTLAND, 1995), detectar olhos (MILANO; BARROZO; HONORATO, 2010), discernir entre formas geométricas, aferir a velocidade de deslocamento de um corpo em uma sequência de quadros (SHAH, 1997) ou até interpretar o relevo de determinado território.

Como exemplos de aplicação da Visão Computacional é possível citar: sistemas de segurança contra-terrorista de aeroportos, mani-

pulação de braços robóticos em fábricas, diagnósticos médicos por imagem, rastreamento de mísseis, controle de qualidade em fábricas, exploração espacial, previsão do tempo, identificação de incêndios, etc (MILANO; BARROZO; HONORATO, 2010). A diversidade de áreas em que pode ser aplicada a Visão Computacional faz com que seja crescente o número de estudos neste campo.

3 RECONHECIMENTO FACIAL

Realizar o reconhecimento facial significa dotar um computador da capacidade de, a partir de informações prévias sobre algumas pessoas, inferir se uma imagem qualquer é pertencente a algum indivíduo conhecido. Esta capacidade pode ser aproveitada em diversas áreas, para muitas aplicações, tais como as listadas na Tabela 1.

Áreas	Aplicações Específicas
Biometria	Passaportes, Programas de Imigração
Segurança da Informação	Logon, segurança de aplicação, segurança de banco de dados, arquivos médicos, terminais de auto-atendimento
Cumprimento da Lei e Segurança	Videomonitoramento avançado, rastreamento e investigação de suspeitos
Cartões inteligentes	Autenticação de usuário
Controle de acesso	Facilitador de acesso, Acesso veicular

Tabela 1 – Aplicações típicas de Reconhecimento Facial
Extraído e adaptado de: (ZHAO et al., 2003)

De acordo com Zhao et al. (2003), o Reconhecimento Facial é muito próximo de técnicas de reconhecimento de objetos. Um dos seus pontos em comum é a necessidade de realizar a extração das características da região de interesse. Portanto, muitas considerações feitas neste capítulo são válidas em reconhecimento de objetos ou padrões.

Os métodos de detecção e reconhecimento de faces podem ser divididos em duas classes: baseados em características e baseados em imagens (HJELMÅS; LOW, 2001). Na primeira, estão os métodos de análise de baixo nível, utilizando como características: cor, contornos, níveis de cinza e distâncias entre elementos faciais. Na segunda classe, estão as redes neurais e as abordagens estatísticas. A problemática deste trabalho é reconhecer pessoas pelo aspecto global da sua face.

Turk e Pentland (1991) defendem que, utilizando o conceito de autofaces, os componentes principais de uma face podem ou não ter relação direta com suas feições. Com isto, esta abordagem leva em

consideração a face de forma holística, realizando traços que não seriam considerados inter-relacionados, de acordo com a percepção humana.

Considera-se cada *pixel* como sendo uma característica de cada imagem. Para realizar uma análise holística é necessário descobrir se há uma inter-relação entre as características do conjunto de amostras. Diferentemente da maioria das operações de processamento de imagens, a distância entre *pixels* não tem influência nesta análise. Na prática, se determinados *pixels* vizinhos seguirem sempre um padrão, na análise de dados haverá uma forte relação entre estas então chamadas características do conjunto amostral.

3.1 ANÁLISE DE DADOS

Uma imagem composta de apenas 1 *pixel* ao ser representada em um sistema de coordenadas, levando em conta a sua intensidade de luz, seria um ponto em um espaço de 1 dimensão. A posição neste espaço unidimensional depende apenas do seu valor de nível de cinza. Quando esta abordagem é feita em relação a um par, 2 *pixels*, sua projeção como um ponto em um gráfico necessita de 2 dimensões para projetar todas as possibilidades de valores. De forma análoga, para representar a combinação de n *pixels* são necessárias n dimensões. A interpretação pode-se dar como a de um ponto em um espaço n -dimensional ou de um vetor de n dimensões.

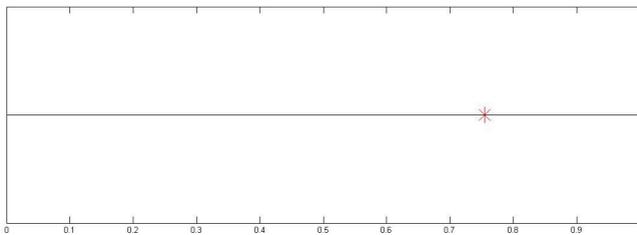


Figura 4 – Projeção de uma imagem com apenas 1 *pixel* em espaço unidimensional (ou um vetor de 1 posição) em um espaço unidimensional. $P = 0,7549$

Considerando uma matriz $N \times M$ que contém os níveis de intensidade de cada *pixel* de uma imagem, esta possui n *pixels*, onde $n = N \cdot M$. Esta imagem qualquer é um ponto em um espaço n -dimensional. Se-

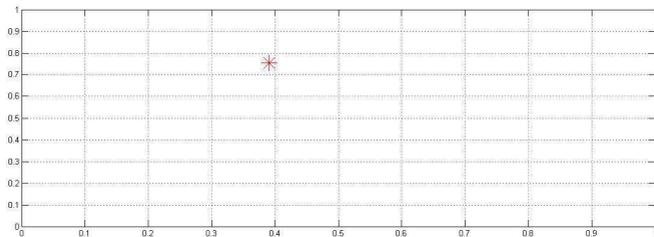


Figura 5 – Projeção de uma imagem com 2 *pixels* em plano 2D (ou um vetor de 2 posições) em um plano 2D. $P = [0,3912 \ 0,7528]$

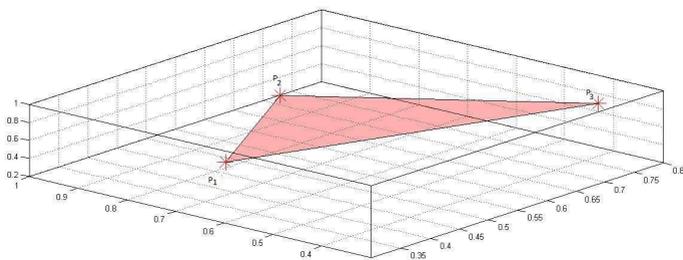


Figura 6 – Projeção de 3 imagens com 3 *pixels* cada em espaço 3D (ou 3 vetores de 3 posições) em um espaço 3D. $P_1 = [0,3452 \ 0,7643 \ 0,6953]$; $P_2 = [0,6523 \ 0,3915 \ 0,9591]$; $P_3 = [0,7129 \ 0,8163 \ 0,3205]$

gundo (MARTÍNEZ; KAK, 2001), estes espaços são grandes demais para realizar um reconhecimento de forma rápida. Para solucionar este problema pode-se utilizar técnicas de redução de dimensionalidade.

Sirovich e Kirby (1987) perceberam que imagens que sigam algum tipo de padrão, como o caso de imagens de faces, quando projetadas neste espaço, sua distribuição não apresenta muita dispersão, apresentando características de uma distribuição normal (ou gaussiana). Isto significa que os pontos não estão dispersos em todo o espaço, mas sim concentrados em uma pequena parte. Intrinsecamente, a problemática de reconhecimento de objetos apresenta distribuições concentradas em poucas dimensões (MOGHADDAM; PENTLAND, 1995).

Desta forma, para representar este tipo de dados pode-se utilizar alguma técnica que não necessariamente seja capaz de remontar todo o espaço euclidiano. Ou seja, é possível utilizar apenas um subespaço

vetorial que contenha as faces. Para isto, altera-se a base da canônica para outra que maximize a variância das amostras.

Apenas a mudança de base, que não representa uma perda de informação. Porém, ainda que a utilização de uma base menor, com os vetores que correspondam as direções de maior variância, resulte em uma perda de informação, ela é relativamente pequena em detrimento de custos de armazenamento e processamento.

3.2 ANÁLISE DE COMPONENTES PRINCIPAIS - PCA

A Análise de Componentes Principais foi inventada em 1901 pelo Britânico Karl Pearson, que realizou várias contribuições para a área de teoria estatística, além desta. Ela também é conhecida como Expansão Karhunen-Loeve, no contexto de reconhecimento de padrões, ou Transformada de Hotelling. É comumente usada na análise exploratória de dados e na construção de modelos preditivos (ABDI; WILLIAMS, 2010). Esta técnica consiste em encontrar um subespaço cujas bases vetoriais correspondam às direções de máxima variância no espaço original (MARTÍNEZ; KAK, 2001). Ou seja, calcular os componentes que representam a máxima variância em um conjunto de amostras (Figura 7). Para fins discriminativos, este conjunto será chamado “Galeria”.

Conforme Shlens (2014), a técnica PCA baseia-se na seguinte pergunta: “Há alguma outra base, que seja uma combinação linear da base original, que melhor re-expressa nosso conjunto de dados?”. Para responder esta pergunta, um primeiro passo é calcular a matriz de covariância das amostras em relação a elas mesmas. Esta matriz indica o grau de relacionamento entre cada *pixel*. A magnitude destes valores indica o grau de redundância, afirma Shlens (2014). Ou seja, valores com magnitude elevada implicam em uma redundância entre 2 *pixels*. Por “redundância” deve-se entender: “informação repetida, logo, desnecessária”.

A partir da matriz de covariância são calculados seus autovalores e autovetores. Cada autovetor é um vetor base do espaço vetorial gerado pela Galeria. O autovalor associado a cada autovetor indica a significância daquele vetor para o espaço. Desta forma, o autovetor associado ao maior autovalor é o que representa a maior variância.

Para aplicar PCA à problemática de reconhecimento facial, deve-se realizar algumas operações iniciais. As imagens, computadas como matrizes, devem ser transformadas em vetores. Considerando que as faces possuem tamanho $N \times N$ *pixels* e considerando M a quantidade

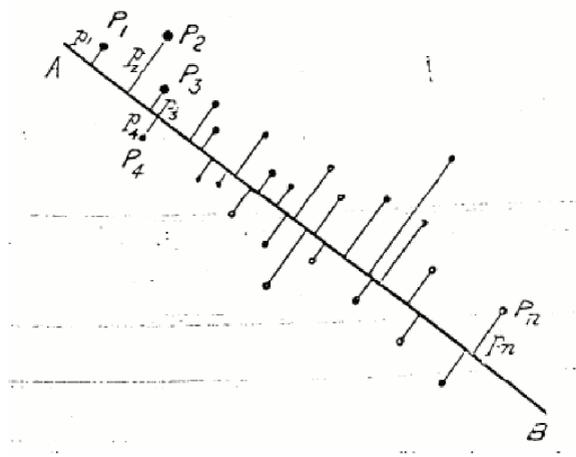


Figura 7 Pontos em espaço 2D e linha que melhor captura a variância destes (ou que minimiza os erros)

Extraído de: Pearson (1901).

de amostras e $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M$ o conjunto de faces para treinamento. O vetor médio deve ser calculado e subtraído de cada amostra, de acordo com as Equações 3.1 e 3.2, respectivamente.

$$\Psi = \frac{1}{M} \sum_{j=1}^M \Gamma_j \quad (3.1)$$

$$\Phi_j = \Gamma_j - \Psi \quad (3.2)$$

Então, compõe-se uma matriz $A = [\Phi_1 \ \Phi_2 \ \Phi_3 \dots \ \Phi_M]$, com cada amostra sendo uma coluna. Esta matriz possui configuração $N^2 \times M$, onde N^2 é a quantidade de *pixels* de cada imagem e M é a quantidade de amostras. Então, a matriz de covariância é determinada por:

$$C = AA^T \quad (3.3)$$

O próximo passo é calcular os autovetores que compõem esta matriz. Tomando u_i e λ_i como os autovetores e autovalores de C ,

respectivamente, é definido que

$$AA^T \mathbf{u}_i = \lambda_i \mathbf{u}_i \quad (3.4)$$

Como a matriz de covariância é simétrica, seus autovetores possuem a característica de serem ortonormais (CONCI; AZEVEDO; LETA, 2008). Estes compõem uma base ortonormal do espaço de amostras. Os autovetores possuem uma melhor representação como imagem e, por isto, são chamados de autofaces, ou autofiguras (ou também *eigenfaces*, derivado de *eigenvectors*). Contudo, a matriz C é $N^2 \times N^2$, e, apesar de ser uma operação matematicamente válida, determinar N^2 autovalores é uma tarefa inviável, do ponto de vista computacional (SIROVICH; KIRBY, 1987).

3.2.1 Redução do Espaço Dimensional

Para resolver o problema da complexidade das operações, é factível realizar algumas simplificações algébricas. Se a quantidade de amostras, M , for menor que a dimensão do espaço, N^2 , então haverá apenas $M - 1$ autovetores associados a autovalores não nulos. Ou seja, somente $M - 1$ autovetores representam alguma informação. Isto porque são necessários apenas $M - 1$ vetores para compor um subespaço que contenha todas as amostras. Por mais não-lineares que as imagens estejam no espaço, o maior subespaço formado por elas será a quantidade de pontos no espaço menos 1.

Na Figura 6 é possível observar que com 3 amostras (3 pontos no espaço) há um subespaço de 2 dimensões que as compreende. De fato, se o espaço fosse de maiores dimensões (o exemplo possui 3), porém ainda houver apenas 3 pontos neste espaço, sempre haverá um plano¹ que compreenda todas as amostras. Generalizando, com M amostras, a quantidade de dimensões do subespaço necessária para compreender todas as amostras é $M - 1$.

Diante destas constatações, para aplicações onde a quantidade de amostras na Galeria é muito maior que a quantidade de *pixels* em cada imagem ($n \gg M$) há a possibilidade da redução das dimensões das operações matriciais, sem perda de informação. Isto porque só é necessário um pequeno subespaço, em relação ao espaço original, para abranger todos os pontos. Esta condição, $n \gg M$, é bastante comum. Por exemplo: considerando faces de tamanho 200×200 *pixels*,

¹Espaço de 2 dimensões

se forem utilizadas até 40.000 amostras para treinamento este método é vantajoso.

Considerando uma matriz $L = A^T A$, de tamanho $M \times M$, tem-se que:

$$A^T A v_i = \mu_i v_i \quad (3.5)$$

onde v_i são os autovetores e μ_i os autovalores de $A^T A$. Multiplicando ambos os lados por A , obtém-se:

$$A(A^T A v_i) = A(\mu_i v_i) \quad (3.6)$$

Reorganizando a Equação 3.6, produz-se:

$$(A A^T) A v_i = \mu_i (A v_i) \quad (3.7)$$

Analisando a Equação 3.7 e a Equação 3.4 é possível observar que $A v_i$ são os autovetores, u_i , da matriz $C = A A^T$ (TURK; PENTLAND, 1991).

$$u_i = A v_i \quad (3.8)$$

Estes autovetores, pela natureza da aplicação, se assemelham a faces e por isso são também chamados de *autofaces*. Portanto, partindo da matriz L , deve-se calcular seus autovetores, v_i , e realizar a multiplicação de A por eles, para então obter as autofaces, u_i , que representam o conjunto de amostras. Esta simplificação reduz a quantidade de operações de acordo com a relação: $(\frac{N^2}{M})^2$, visto que C é uma matriz do tipo $N^2 \times N^2$ e L é $M \times M$. Desta forma, um problema de complexidade $O(n^2)$ transforma-se então em dois problemas separados de complexidade $O(n)$ (RAO, 2015).

Outra estratégia, ressaltada por Sirovich e Kirby (1987) e utilizada por Turk e Pentland (1991), é a utilização de um menor número de autofaces. De acordo com Moon e Phillips (2001), utilizar um conjunto muito reduzido ou utilizar o conjunto completo de autofaces prejudicam o resultado, sendo sugerido o uso de 40% das autofaces mais significantes². M' é o índice do último autovetor utilizado, então $M' = M \cdot 0,4$. Sendo assim, utilizando esta técnica há uma redução de armazenamento e de cálculo que, possivelmente, de acordo com trabalhos semelhantes, implica no aumento do desempenho do algoritmo.

²as autofaces mais significantes são as que possuem um maior autovalor associado.

A raiz de um autovalor indica o quanto as amostras seguem ao longo da direção do autovetor referente (SIROVICH; KIRBY, 1987). Em outras palavras, $\sqrt{\lambda_n}$ mede o quanto do conjunto de amostras é representado pelo n -ésimo autovetor. Desta forma, uma maneira de calcular a significância de um autovetor em relação ao espaço de faces gerado é calculando:

$$p_n = \frac{\sqrt{\lambda_n}}{\sum_{i=0}^{M'} \sqrt{\lambda_i}} \quad (3.9)$$

3.2.2 Decomposição em Autofaces

As autofaces são a base do espaço dimensional formado pelo conjunto de faces conhecidas. Isto significa que a partir da combinação linear destas autofaces é possível obter qualquer face do conjunto de treinamento. Isto permite reconstruir todas as faces amostradas a partir das autofaces e dos coeficientes da combinação linear das mesmas. Para cada nova imagem no conjunto, o espaço de armazenamento necessário para sua reconstrução é apenas de M coeficientes, não sendo necessário armazenar N^2 *pixels*.

Conforme ressaltado na Subseção 3.2.1 e ilustrado na Figura 8, de acordo com o aumento do índice do autovetor, ou autoface, menor a significância por ele apresentada em relação ao conjunto de treinamento. Outra observação interessante de se fazer nesta Figura é que a primeira autoface carrega grande parte das informações comuns a maioria das amostras. Ou seja, conforme esperado, ela representa o eixo onde há maior variância, sendo seus *pixels* mais claros os que mais influenciam na caracterização de uma face.

Muitos trabalhos, como os de Martínez e Kak (2001), Moon e Phillips (2001) e Zhao et al. (2003), revelam que a não utilização das autofaces de menor ordem causa uma melhora nos resultados do sistema. Isto se deve a grande probabilidade de que estas contenham informações relacionadas ao *background*, que são dispensáveis, e até inconvenientes, para o reconhecimento facial.

A Equação 3.10 calcula os pesos, ou pontuações, ou ainda *scores*, de cada autoface $(1, \dots, M')$ para uma determinada amostra Φ_j , onde j é o índice da amostra.

$$\omega_{ji} = \mathbf{u}_i^T \Phi_j \quad (3.10)$$

Realizando esta operação para todas as faces do conjunto de treina-



Figura 8 – Autoespaço com 16 autofaces, em ordem decrescente de acordo com o autovalor associado

mento, obtém-se um vetor de pesos, $\Omega_j = [\omega_{j1} \ \omega_{j2} \ \dots \ \omega_{jM'}]$, para cada face j (TURK; PENTLAND, 1991).

Na Figura 9 é possível ver duas amostras pertencentes ao conjunto de treinamento. Estas, quando projetadas no subespaço gerado por uma parte do total de autofaces, diferem das imagens originais. Considerando a Subseção 3.2.1, onde foram explanadas as vantagens de se utilizar um menor número de autofaces, haverá um erro associado à reconstrução das amostras do conjunto. A vantagem desta técnica é que mesmo com um reduzido número de autofaces (apenas 10% nas Figuras 9c e 9g) o erro apresentado é relativamente baixo. Isto porque a maior parte da variância se concentra na menor parte das autofaces.

3.2.3 Classificação

Para realizar o reconhecimento de uma face é preciso coletar uma amostra que não pertença ao conjunto de treinamento. Esta é comumente chamada de imagem de prova. Conforme também realizado com a Galeria, esta imagem deve ser normalizada, sendo necessária a subtração do vetor médio calculado anteriormente, segundo a Equação 3.2. O resultado da operação anterior será chamado, para fins explicativos,

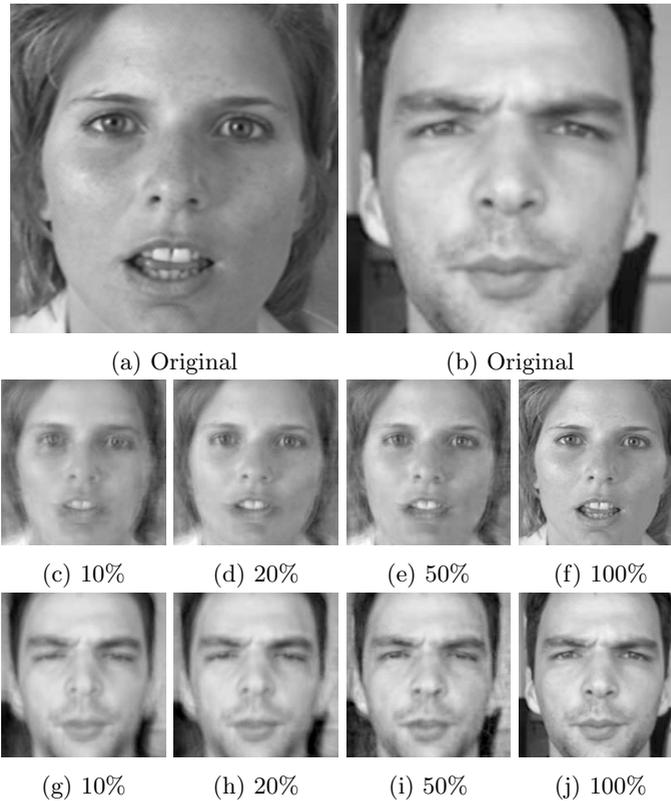


Figura 9 – Reconstrução de duas faces a partir de uma fração das autofaces. As proporções de 10%, 20%, 50%, 100% de quantidade de autofaces correspondem, respectivamente, a 44%, 57%, 80% e 100% da variância total, aproximadamente (Equação 3.9)

Adaptado de: (BIOID, 2001).

de Φ_P . Após este passo, aplicando a Equação 3.10 é possível adquirir os pesos desta imagem. Analogamente, estes pesos, ω'_i , quando multiplicados por cada autoface, u_i , produzem a projeção da imagem de prova no espaço de faces (TURK; PENTLAND, 1991):

$$\Phi'_P = \sum_{i=1}^{M'} \omega_{j_i} u_i \quad (3.11)$$

sendo i o índice do respectivo autovetor, Φ'_P a projeção da imagem de prova e ω'_i seus pesos. Esta projeção pode diferir da imagem original (como pode ser observado na Figura 9), uma vez que ela não esteja completamente no espaço de faces. Uma forma de computar esta medida é calculando a distância Euclidiana entre a imagem original e a projeção, também conhecida como *distance from the face space (DFFS)*. A Equação 3.12 apresenta este cálculo.

$$\epsilon^2 = \|\Phi - \Phi'\|^2 \quad (3.12)$$

O valor de ϵ é a medida de quanto a imagem-teste se assemelha a uma face, de acordo com o espaço gerado pelo conjunto de treinamento Hjelmås e Low (2001). Para automatizar esta etapa, pode ser escolhido um parâmetro θ para o qual se $\epsilon < \theta$ considera-se que a imagem de prova representa uma face, conforme será melhor abordado na Seção 3.4.

Comparando os pesos da imagem de prova com os pesos de cada face conhecida, é possível obter uma medida de semelhança. Se uma classe (pessoa) possuir mais de uma face no conjunto de treinamento, os valores usados na comparação devem ser os pesos da face média da classe. Para realizar a medida de semelhança existem alguns métodos, sendo que a escolha de um ou outro é considerada uma decisão crítica para os resultados (MOON; PHILLIPS, 2001). O método mais simples consiste em calcular a distância Euclidiana entre os pesos de cada classe e os pesos da imagem de prova. Esta distância é chamada de *Distance Within the Face Space (DIFS)*. Este processo de aferir a imagem de prova como pertencente a uma classe é chamado de classificação por vizinho mais próximo. A distância entre a face de prova e uma determinada classe k é dada pela seguinte equação:

$$\epsilon_k^2 = \|\Omega - \Omega_k\|^2 \quad (3.13)$$

onde k é o índice das classes.

Similar ao realizado com a distância em relação ao espaço de faces, pode-se aplicar um limiar, θ_ϵ , sendo que se $\min \epsilon_k < \theta_\epsilon$ considera-se que a imagem de prova pertence a k -ésima classe. Caso nenhum ϵ_k seja menor que o limiar, a face é considerada pertencente à uma pessoa desconhecida.

O reconhecimento pode ser realizado de duas maneiras: para verificar a autenticidade de uma pessoa ou para identificá-la (BRAGA, 2013). Na primeira, o cálculo dá-se da seguinte forma:

1. Calculam-se os pesos da face de teste;
2. Calcula-se a DIFS entre a face de teste e a classe objetivo;
3. Verifica-se se a DIFS é menor que um determinado limiar, indicando que ela é suficientemente semelhante e pode então ser (ou não) aferida como pertencente à pessoa avaliada.

No segundo método, a abordagem, mostrada a seguir, é um pouco diferente:

1. Calculam-se os pesos da face de teste;
2. Calcula-se a DIFS entre a face de teste e cada classe conhecida;
3. A classe que distar menos da projeção da face avaliada é considerada a mais semelhante. Ou seja, a que minimizar a Equação 3.13;
4. Verifica-se se a DIFS da classe escolhida é menor que um determinado limiar, indicando que ela é suficientemente semelhante e pode então ser (ou não) aferida como uma face conhecida;
5. Caso a face seja considerada desconhecida, pode-se realizar um novo cadastro no banco.

Conforme os conceitos estabelecidos nesta seção, a Figura 10 ilustra as distâncias DFFS e DIFS no espaço de faces F . Na ilustração, m representa a média das amostras, x representa a imagem de prova e \tilde{x} é a sua projeção no espaço de faces. A Tabela 2 lista os possíveis resultados da etapa de classificação de uma amostra de prova.

É possível verificar que o desempenho do algoritmo está intimamente relacionado com a escolha dos valores de θ e θ_ϵ (MOON; PHILLIPS, 2001). Turk, Pentland et al. (1991) analisa que para um perfeito reconhecimento, mais estes parâmetros devem se aproximar de zero, porém ao custo de muitas imagens serem rejeitadas como desconhecidas. O

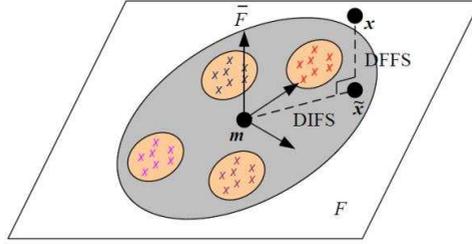


Figura 10 – Distâncias em relação ao autoespaço
 Extraído de: (TURK; PENTLAND et al., 1991)

DFFS	DIFS	Conclusão
$\epsilon < \theta$	$\min(\epsilon_k) < \theta_\epsilon$	Amostra é uma face pertencente a uma classe conhecida
$\epsilon < \theta$	$\min(\epsilon_k) > \theta_\epsilon$	Amostra é uma face desconhecida pelo sistema
$\epsilon > \theta$		Amostra não é uma face e é não está próxima de nenhuma classe conhecida

Tabela 2 – Classificação de uma imagem de prova de acordo com DFFS e DIFS

oposto também ocorre: ao aumentar estes fatores, mais imagens serão reconhecidas, porém isto aumenta o número de falsos positivos. Desta forma, estas constantes devem ser escolhidas de maneira a balancear estes dois cenários. É possível concluir também que não há uma solução ótima para todos os casos, estes parâmetros devem ser especialmente escolhidos consonante com cada aplicação.

Outras técnicas para medir a DFFS e a DIFS podem gerar melhor resultado que a distância Euclidiana (MOON; PHILLIPS, 2001). Uma técnica que é bastante utilizada, (LEARNED-MILLER et al., 2015), (HJELMÁS; LOW, 2001), por levar em consideração as variações em cada autoespaço para relacionar a distância entre dois vetores é a distância de Mahalanobis.

$$d(x, y) = \sum_{i=1}^t \frac{|x_i - y_i|}{\sqrt{\lambda_i}} \quad (3.14)$$

A Equação 3.14 relaciona a distância entre os vetores X e Y . λ_i são os autovalores da matriz de covariância e t é a quantidade de autovetores deste espaço. Substituindo a distância Euclidiana por esta técnica nas Equações 3.12 e 3.13, obtém-se:

$$\epsilon = \sum_{i=1}^{M'} \frac{|\Phi_P - \Phi'_P|}{\sqrt{\lambda_i}} \quad (3.15)$$

$$\epsilon_k = \sum_{i=1}^{M'} \frac{|\omega_P i - \omega_{k_i}|}{\sqrt{\lambda_i}} \quad (3.16)$$

onde k é o índice de cada classe e P identifica valores pertencentes à face de prova.

Após o correto reconhecimento da face de prova, pode-se adicioná-la ao conjunto de treinamento da classe referente. Esta prática visa a constante atualização e melhora do sistema, visto que algumas classes podem ter poucas amostras e já levando em conta a esporádica mudança do rosto humano. Porém, conforme Braga (2013), há a desvantagem de ter que recalculer os autovetores novamente a cada nova face adicionada.

3.3 BASES DE DADOS

Apesar de haver muitos algoritmos para realização de reconhecimento facial, é difícil obter uma métrica de avaliação a respeito dos resultados se os bancos de imagens forem significativamente diferentes. Ou seja, as restrições utilizadas para aquisição das amostras de um determinado sistema devem ser as mesmas de outro sistema para que seus resultados sejam comparáveis. Isto se deve ao fato de que variações na etapa de aquisição, como diferentes orientações da cabeça, incidência de luz ou expressão facial afetam o desempenho da maioria dos algoritmos (BRAGA, 2013). Além disto, decisões como quantidade de amostras por pessoa, quantidade de autofaces utilizadas e limiares de DFFS e DIFS também são muito influentes nos resultados.

Para possibilitar a comparação entre as diversas implementações de algoritmos para reconhecimento, existem algumas bases de imagens disponíveis para o uso. Cada base é composta a partir de restrições e, sendo assim, apresenta imagens em ambientes controlados. Alguns exemplos de bases de dados utilizadas como padrão por pesquisadores são: *The Color FERET database*, *The Yale Face Database B*, *The AR Face Database*, *BioID Face Database* e *Labeled Faces in the Wild* (GRGIC; DELAC, 2005-2016).

Conforme Learned-Miller et al. (2015), em 2006, os resultados

obtidos a partir de bases de dados já estavam saturados, porém em aplicações reais estas soluções apresentavam baixo desempenho. Então, faz-se necessário o desenvolvimento de técnicas que se apliquem em bases de dados mais condizentes com situações reais. A base *Labeled Faces in the Wild* não possui restrições no seu banco de imagens. Ela consiste em mais de 13.000 imagens de faces coletadas na web. Cada uma está rotulada com a identificação de a quem ela pertence. Aplicações que tenham bons resultados neste tipo de base de dados estão mais propensas a terem bons resultados em situações reais, como, por exemplo, reconhecimento em tempo real por câmeras de segurança.

Mesmo compartilhando o mesmo conjunto de imagens de treinamento, ainda faz-se necessário determinar um método para comparar resultados entre as diversas bases existentes. Isto pode ser realizado calculando a curva Característica de Operação do Receptor, ou *Receiver Operating Characteristic* (ROC), em inglês. A curva ROC é um gráfico que representa o desempenho do sistema, quando submetido a diferentes cenários ou parâmetros. O parâmetro alterado, neste trabalho, é θ_ϵ , diminuindo ou aumentando a distância máxima entre a face de avaliação e a face média de uma classe para que a ela seja considerada pertencente a esta classe.

Como métrica de desempenho são utilizados os conceitos de Sensibilidade e Especificidade. Seus cálculos são apresentados nas Equações 3.17 e 3.18:

$$\text{Sensibilidade} = \frac{VP}{P} \quad (3.17)$$

$$\text{Especificidade} = \frac{VN}{N} \quad (3.18)$$

onde VP é a quantidade de faces corretamente identificadas (verdadeiros positivos), P é a quantidade total de faces conhecidas, VN é a quantidade de faces ausentes no conjunto de treinamento consideradas como desconhecidas (verdadeiros negativos) e N é a quantidade total de faces desconhecidas. Sensibilidade, no contexto deste trabalho, é a capacidade do sistema classificar corretamente faces de classes contidas no conjunto de treinamento. Especificidade é a capacidade do sistema corretamente identificar classes desconhecidas. A curva ROC é obtida por meio da projeção da Sensibilidade *versus* $(1 - \text{Especificidade})$ para variados cenários.

Outra forma de qualificar um sistema é medir sua Precisão:

$$Precisao = \frac{VP + VN}{P + N} \quad (3.19)$$

Precisão é a estimativa de acerto do sistema, considerando reconhecimentos corretos e rejeições corretas.

3.4 DETECÇÃO DE FACE

Segundo Hjelmås e Low (2001), a detecção de faces em imagens é o primeiro passo para realizar seu reconhecimento, além de ser o emprego mais importante da detecção. O objetivo deste estágio é localizar e extrair, por segmentação, a região de interesse do restante da cena. Isto é necessário visto que uma face pode dar-se em um ambiente complexo e em diferentes poses. De acordo com (BRAGA, 2013), esta etapa é importante pois elimina da imagem informações desnecessárias. As Figuras 11(a-b) exemplificam os passos deste processo.

Os esforços em relação a detecção facial datam do início da década de 70, porém estas técnicas apresentavam bons resultados apenas em condições bastante específicas, o que vai contra o objetivo de detectar faces em situações reais. Até os anos 90, este cenário continuou praticamente estagnado. A partir daí, novas técnicas surgiram, dando mais flexibilidade aos sistemas. Isto permitiu a realização da detecção de forma mais eficiente, viabilizando sua utilização como etapa inicial para o reconhecimento.

Para o correto funcionamento da detecção de faces, é de fundamental importância seu treinamento. Para que isto ocorra, devem ser utilizadas tanto imagens de faces distintas, quanto imagens que não são faces. Então, pode-se aferir que a detecção facial é um problema de reconhecimento de padrões de duas classes, uma relativa a faces e outra a não-faces (BRAGA, 2013).

Nas subseções a seguir serão apresentados dois algoritmos para realizar esta tarefa. O algoritmo de Viola-Jones se baseia em características locais, enquanto a Detecção por Minimização da DFFS se aproveita da própria composição espacial utilizada na técnica PCA.

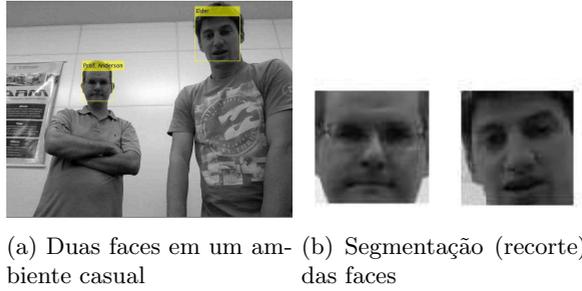


Figura 11 – Detecção e classificação de faces em tempo real

3.4.1 Algoritmo de Viola-Jones

O algoritmo de Viola e Jones (2001) baseia-se em três conceitos: integral de imagem, treinamento de classificadores por *boosting* e cascadeamento de classificadores (BRAGA, 2013). Sua completa descrição não faz parte do escopo deste trabalho e pode ser melhor compreendida em “*Rapid object detection using a boosted cascade of simple features*” (VIOLA; JONES, 2001).

A integral de imagem visa calcular eficientemente a soma dos valores dos *pixels* de uma área retangular em uma imagem. Esta etapa permite o reuso dos valores de integral de cada *pixel*, evitando cálculos repetitivos.

Boosting é um método de aprendizagem utilizado no treinamento de classificadores, que cria um classificador “forte” a partir da combinação de vários classificadores “fracos”. Desta forma, operações simples executam uma classificação complexa.

O cascadeamento combina vários classificadores obtidos no estágio anterior em sequência, de maneira que as amostras são submetidas ao primeiro classificador e somente serão submetidas aos próximos se forem avaliadas positivamente nos anteriores. Cada classificador tende a ser mais complexo e específico que a etapa anterior. Isto evita que seja despendido tempo em observações que já falhem em classificadores simples, eliminando imagens que divergem muito de faces já nos primeiros estágios.

Este método de detecção visa utilizar máscaras para analisar diferenças de intensidade entre regiões da imagem. Isto torna-se útil quando percebe-se alguns padrões, como a região dos olhos, onde o nível de intensidade dos *pixels* é menor que os da área logo abaixo, das

bochechas. Uma máscara aplicada para esta situação é a apresentada na Figura 12 (B). Assim, realizando a passagem desta máscara em toda a imagem, obtém-se resultados que indicam o quanto aquela posição se assemelha com o aspecto típico dos olhos.

Empregando outros operadores, como o da Figura 12 (A), é possível identificar locais da imagem que se pareçam com a relação entre nariz e bochechas. Aplicando este classificador em cascata com o citado no parágrafo anterior, melhora-se a precisão em localizar traços faciais. Estes classificadores, por causa da sua simplicidade, são considerados classificadores fracos. A junção de vários destes, em cascata, formam um classificador forte.

Portanto, o algoritmo de Viola-Jones faz tarefas simples, em sequência e apresenta boa precisão com um tempo de processamento muito baixo. Outra vantagem dele é que não necessita ser treinado exaustivamente, com variadas amostras de classes distintas (faces e não-faces) para que tenha um resultado satisfatório, afirma Braga (2013).

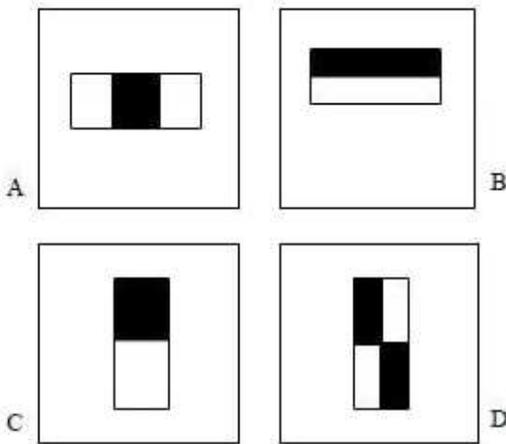


Figura 12 – Exemplo de classificadores *Haar-like*
Extraído de: (BRAGA, 2013)

3.4.2 Detecção por Minimização da DFFS

Conforme apresentada na Subseção 3.2.3, uma maneira de classificar se uma imagem é semelhante a uma face ou não é realizar o cálculo da distância em relação ao espaço de faces (DFFS). Proposto por Turk e Pentland (1991), na etapa de detecção de face é possível realizar uma varredura por janela, aplicando o cálculo da distância sobre regiões menores de uma imagem. O resultado é uma matriz de distâncias, de acordo com a posição escolhida para a aplicação do cálculo.

A Figura 13 ilustra este efeito. As regiões mais escuras indicam uma maior semelhança destas com o espaço facial. É possível verificar que, corretamente, a imagem indica a presença de uma face no quadrante inferior esquerdo, onde se concentram as menores intensidades (destacado na Figura 13).

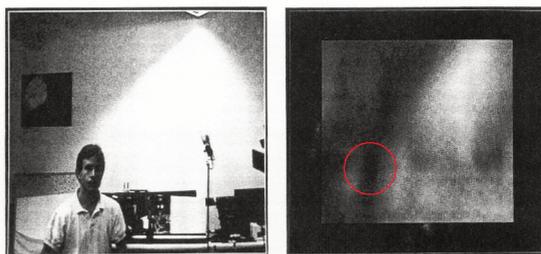


Figura 13 Detecção por minimização de DFFS
Extraído de: (TURK; PENTLAND, 1991)

Este algoritmo subdivide a imagem em diversas outras imagens menores, de acordo com o tamanho definido da janela. Porém, em um ambiente real, uma face pode estar mais afastada ou menos afastada do dispositivo de aquisição (câmera), sendo representada em diferentes tamanhos na imagem. Realizar a aplicação desta técnica com tamanhos diferentes de janela e encontrar a escala que representa menor distância entre a imagem e o espaço de faces possibilita a determinação do tamanho da face, para posterior segmentação. Apesar dos bons resultados, esta técnica demanda muito mais processamento em comparação a utilização do algoritmo de Viola-Jones, que, atualmente, é o estado da arte em Detecção Facial.

4 SISTEMA EMBARCADO DE RECONHECIMENTO FACIAL

Este Capítulo tem como objetivo abordar o desenvolvimento de um sistema embarcado para reconhecimento facial baseado na síntese dos conceitos vistos nos capítulos anteriores. Inicialmente será tratado da implementação do algoritmo para sua validação e avaliação de desempenho. Posteriormente há a explanação a respeito da implantação do sistema em hardware embarcado.

4.1 DESENVOLVIMENTO

O treinamento foi dividido em etapas consequentes, as quais são elencadas a seguir:

1. Aquisição
2. Detecção de face
3. Segmentação da face
4. Equalização de histograma
5. Inclusão de todas as faces no banco
6. Treinamento

O reconhecimento foi segmentado nas seguintes etapas:

1. Aquisição
2. Detecção de face
3. Segmentação da face
4. Equalização de histograma
5. Cálculo de DIFS
6. Classificação

A plataforma escolhida para a implementação do sistema foi o MATLAB (*MATrix LABoratory*). Este ambiente foi escolhido por facilitar a manipulação de matrizes, em operações como redimensionamento e multiplicação. Além disto, ele possui algumas ferramentas

para trabalho com imagens e interface gráfica que agilizam o desenvolvimento. A interface gráfica foi desenvolvida por meio da ferramenta GUIDE (*Graphical User Interface Design Editor*) do MATLAB e pode ser observada na Figura 14. Ela possibilita a realização de todas as etapas acima listadas, por meio de programação orientada a eventos.



Figura 14 – Interface gráfica do sistema desenvolvido

Na etapa de aquisição, as imagens podem ser obtidas através de *webcams* ou carregadas a partir arquivos. Estas são, então, carregadas em matrizes tridimensionais, as quais possuem dois eixos referentes a posição do *pixel* e o terceiro, um vetor de tamanho 3, guarda a intensidade de luz das componentes Vermelho, Verde e Azul do *pixel* em questão.

Conforme Sirovich e Kirby (1987), Turk e Pentland (1991), Viola e Jones (2001), Jesorsky, Kirchner e Frischholz (2001), entre outros autores, imagens em níveis de cinza são ideais para este tipo de aplicação pois ocupam menos espaço, comparadas há imagens coloridas. Desta forma, optou-se por converter, nesta etapa, todas as imagens para este formato, reduzindo a dimensão que apresentava 3 eixos para apenas 1, resultando em matrizes bidimensionais (Figura 15). Afim de realizar normalização e obter maior precisão nos cálculos seguintes, os valores de intensidade de cada *pixel* são armazenados como variáveis do tipo *double*, com *range* de valores de 0 a 1. Com esta abordagem, *pixels* com conteúdo 0 representam a cor preta, com conteúdo 1 representam a cor branca, e valores intermediários são tons de cinza proporcionais a esta variação.

Após a aquisição, é realizada a detecção da face (Figura 16a).



(a) Imagem RGB

(b) Imagem em níveis de cinza

Figura 15 – Exemplo de conversão de uma imagem no espaço de cores RGB para escala de cinza

Extraído e adaptado de (GONZALEZ; WOODS, 2002)

Para esta etapa foi escolhido utilizar o algoritmo de Viola-Jones (VIOLA; JONES, 2001), visto que este não necessita ter um exaustivo treinamento antes de obter resultados satisfatórios (BRAGA, 2013). O mesmo já possui implementação na biblioteca de funções do MATLAB, portanto isto torna a execução da detecção mais rápida. Além disso, a outra abordagem citada no Capítulo 3 necessita que a base de dados seja previamente preenchida com algumas faces para então poder distinguir outras faces.

A saída do algoritmo de detecção são as coordenadas de um retângulo que compreende a face na imagem. Com isto, é possível realizar a segmentação desta área e ignorar tudo que estiver exterior a este retângulo. De forma a padronizar as imagens obtidas nesta etapa, estas são redimensionadas para um mesmo tamanho. Para fins matemáticos, serão consideradas estas dimensões como N_1 e N_2 . Ou seja, o resultado desta fase são matrizes $N_1 \times N_2$.

Em seguida ao redimensionamento (Figura 16b), os *pixels* podem possuir valores de intensidade mal distribuídos. Então, também buscando a normalização de todas as faces, é realizada a equalização de histograma. Desta maneira, espalha-se mais uniformemente os *pixels* nos níveis de cinza, dando maior contraste aos detalhes (Figura 16c). Esta etapa tem o objetivo de deixar o sistema mais robusto à variações globais na iluminação (SARIYANIDI; GUNES; CAVALLARO, 2015).



(a) Imagem com a face detectada demarcada por um retângulo



(b) Segmentação da(s) face(s) detectada(s) e (c) Face após a equalização de histograma

Figura 16 Etapas comuns ao treinamento e ao reconhecimento
Adaptado de (BIOID, 2001)

Um número mínimo de amostras de cada pessoa deve ser cadastrado para que o desempenho do sistema seja satisfatório. Estas devem apresentar pequenas variações em relação a pose e expressão facial para que o software seja tolerante à estas mudanças no momento do reconhecimento. Na solução desenvolvida, após a captura das amostras, estas são adicionadas ao conjunto de treinamento, conjuntamente com a identificação da pessoa a qual pertencem.

Com as classes devidamente cadastradas, é possível realizar o treinamento. Esta etapa consiste em, inicialmente, dispor todas as imagens do conjunto de treinamento em forma de vetores. Conforme estabelecido anteriormente, estes vetores possuem dimensão $(N_1 \cdot N_2) \times 1$. Considerando M a quantidade total de amostras, somando-se todos

os vetores e dividindo por M , obtém-se a face média. Então, é realizada a subtração da face média do conjunto em cada amostra, de forma a obter vetores apenas com o desvio em relação a face média.

Para realizar as operações descritas no Capítulo 3 referentes à técnica PCA, as amostras são agrupadas, cada uma compondo uma coluna, de modo a formar a matriz A , de dimensão $(N_1 \cdot N_2) \times M$. O próximo passo é encontrar a matriz de covariância entre as amostras. Para isto, seguindo o solução de menor custo computacional apresentada no Capítulo anterior, deve-se primeiro calcular a matriz L .

Conforme demonstrado anteriormente, multiplicando $A^T \cdot A$ obtém-se a matriz L . Então são calculados seus autovalores e autovetores. De modo a dispor os autovetores de L por ordem de maior relevância para a variância do espaço, estes são reordenados em ordem descendente de autovalor associado. Isto porque um autovalor de maior módulo indica que seu autovetor representa uma maior variância. Desta forma, após a reordenação, uma porção dos autovetores de menor índice são suficientes para representar a maioria da informação útil para distinção entre amostras. Desta forma, mantém-se apenas M' autovetores, descartando-se o restante. O valor de M' utilizado, conforme a sugestão mencionada no capítulo anterior, é de 40% do total de amostras. Esta consideração é importante para a redução do espaço dimensional e possui implicação direta no tempo de execução do algoritmo.

De posse dos autovetores selecionados de L , estes são agrupados como colunas, de maneira a formar a matriz $V = [v_1 \ v_2 \ v_3 \ \dots \ v'_M]$ de ordem $M \times M'$. Seguindo a Equação 3.8, multiplica-se A por V para então obter a matriz $U = [u_1 \ u_2 \ u_3 \ \dots \ u'_M]$ de ordem $(N_1 \cdot N_2) \times M'$, onde cada coluna é uma autoface do subespaço.

Com a obtenção do subespaço cujas autofaces são base, é necessário projetar as faces médias de cada classe no subespaço. A face média de cada classe j é computada pela média aritmética de suas amostras. Sua projeção é obtida por

$$\Omega_j = U^T \Phi_j \quad (4.1)$$

onde Ω_j é o vetor de pesos da classe j .

Partindo para o reconhecimento, as etapas de Aquisição, Detecção de Face, Segmentação e Equalização de Histograma nada diferem do Treinamento. A etapa de cálculo da distância em relação ao espaço face (DIFS) é realizada primeiramente subtraindo a face média da face de prova. O resultado desta subtração, aqui chamado de Φ_p , é então projetado no espaço de faces. Isto é efetuado pela multiplicação

da face, Φ_p , pela matriz de autofaces U , semelhante a Equação 4.1.

De posse das pontuações, Ω_P e Ω_j , a identificação por vizinho mais próximo é realizada de acordo com as Distâncias de Mahalanobis entre a face projetada e cada classe, conforme a Equação 3.16. Um limiar, θ_k , é definido para que classificações com distâncias acima deste sejam consideradas como “Desconhecido”. Como não há consenso na escolha de um valor ótimo para θ , o sistema permitirá a sua variação em tempo de execução, possibilitando uma avaliação direta de sua influência.

4.2 VALIDAÇÃO

Para a validação da corretude do sistema, foram realizadas algumas rodadas de testes. A base de dados utilizada foi a “*Extended Yale Face Database*” (LEE; HO; KRIEGMAN, 2001). Sua composição é descrita por Lee, Ho e Kriegman (2005). Conforme Grgic e Delac (2005-2016), para validação de técnicas como PCA esta base de dados é adequada por conter várias imagens de cada pessoa.

O banco de faces utilizado possui 64 imagens de 38 indivíduos (2.432 imagens no total) com diferentes ângulos de incidência de luz, porém mesma pose (frontal). Antes de realizar o treinamento é necessário separar as imagens que serão utilizadas na etapa de classificação. Portanto, optou-se por retirar aleatoriamente 3 pessoas para que sejam os “desconhecidos” e 4 imagens de cada classe restante para verificar se são corretamente classificadas.

Foram compostos 5 conjuntos de imagens para ensaio, com diferentes restrições em relação à incidência de luz, visto que, segundo Braga (2013), o erro estimado aumenta se a iluminação ambiente for inadequada. Cada conjunto é um subconjunto do anterior sendo o primeiro a base completa e os restantes com 70%, 50%, 40% e 25% das imagens com melhores condições de iluminação. Desta maneira, é possível verificar o comportamento da precisão do sistema em relação aos critérios de moderação das imagens de entrada. Conforme o aumento nas restrições de luminosidade, a quantidade de imagens para treinamento diminuiu, porém o número de amostras para classificação e o número de indivíduos retirados permaneceram os mesmos. Os ensaios foram realizados com e sem a etapa de equalização de histograma, permitindo avaliar também sua eficácia nos resultados.

Para cada conjunto de parâmetros (restrições de luminosidade e θ_k) foram realizadas 10 execuções do algoritmo. A cada execução, as

imagens separadas foram reintegradas ao conjunto total e, aleatoriamente, foram retiradas novas imagens para posterior classificação. O treinamento também é feito a cada execução.

Classificação / Situação	Pessoa na base	Desconhecido
Pessoa certa	VP	-
Pessoa errada	FP1	FP2
Desconhecido	FN	VN

Tabela 3 – Matriz de Confusão

Reescrevendo as equações 3.17, 3.18 e 3.19, de acordo com esta tabela, obtém-se:

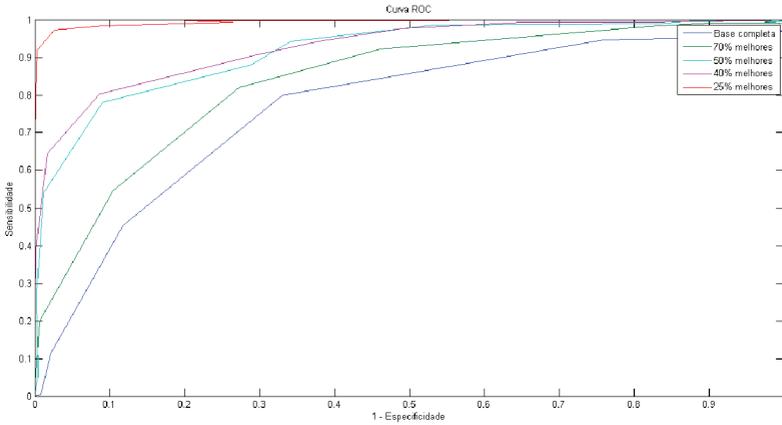
$$\text{Sensibilidade} = \frac{VP}{VP + FP1 + FN} \quad (4.2)$$

$$\text{Especificidade} = \frac{VN}{FP2 + VN} \quad (4.3)$$

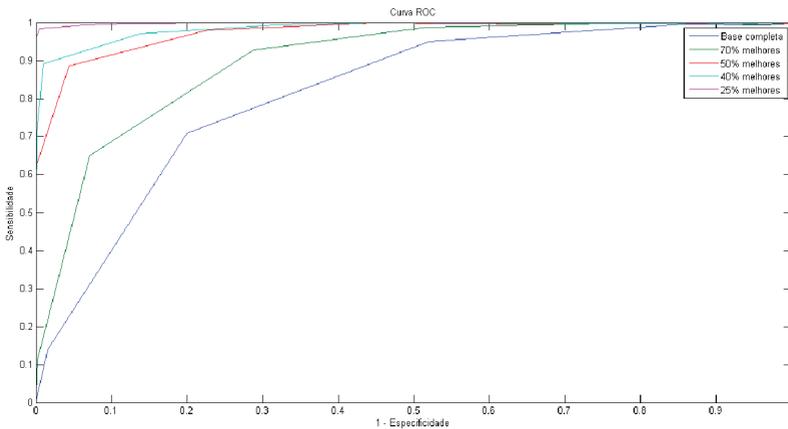
$$\text{Precisao} = \frac{VP + VN}{Total} \quad (4.4)$$

O Pseudocódigo que apresenta as etapas para a obtenção dos qualificadores “Sensibilidade” e “Especificidade” pode ser encontrado no Apêndice A. A partir da Matriz de Confusão é obtida a curva *ROC* do sistema. Esta pode ser utilizada para estimar o ponto de operação ideal, de acordo com as características desejadas (melhor taxa de acerto em detrimento de menor número de falsos positivos, por exemplo). As Figuras 17a e 17b apresentam, respectivamente, as curvas *ROC* dos 5 conjuntos de imagens sem a etapa de equalização de histograma e com a realização da equalização.

Nas Figuras 17a e 17b é possível perceber um aumento significativo na relação Sensibilidade *versus* (1-Especificidade) para os conjuntos de treinamento com mais restrições em relação as condições de luminosidade. Isto permite afirmar que o desempenho do sistema está intimamente ligado às regras de moderação na etapa de aquisição. Outra conclusão válida é que a realização da equalização de histograma, seguindo o mesmo raciocínio anterior, tende a normalizar as diferenças de luminosidade e gerar melhores resultados.



(a)



(b)

Figura 17 Curvas ROC do sistema (a) sem e (b) com equalização de histograma

De posse das curvas ROC, é possível definir o ponto de operação adequado para o sistema. Buscando uma maior Precisão, o ponto de operação escolhido deve aliar alta Sensibilidade com alta Especificidade. Calculando a melhor Precisão de cada ensaio, gera-se a Tabela 4.

	Sem Eq. Histograma	Com Eq. Histograma
Base Completa	72,46%	76,09%
70% da base	77,48%	82,20%
50% da base	83,33%	91,49%
40% da base	84,36%	92,94%
25% da base	97,31%	98,68%

Tabela 4 – Melhores valores de Precisão do sistema

Diante destes resultados é possível realizar a comparação do sistema com outras implementações da bibliografia. Considerando resultados com a mesma base de imagens, a precisão de Zhao et al. (2003) foi de 74% e de Lee, Ho e Kriegman (2005) foi de 70%. Utilizando outras bases de faces, como FERET ou conjuntos próprios, a precisão obtida foi de 90%, 76% e 97% por, respectivamente, Moon e Phillips (2001), (SANTANA; GOMES; SANTOS, 2014) e (SINGH; CHHABRA, 2014). Comparando os resultados deste trabalho com os da bibliografia pode-se considerar que o reconhecimento foi corretamente implementado, e apresenta resultados semelhantes à trabalhos similares.

Outra métrica de desempenho de sistemas de reconhecimento facial consiste em avaliar a imagem de prova de acordo com seus K-vizinhos mais próximos. Neste tipo de verificação não são inclusas imagens que não pertençam às classes conhecidas. Portanto, foram utilizadas 60 amostras de 38 classes para compor a Galeria e 4 amostras de cada classe para reconhecimento. Quando $K=1$, esta análise avalia se a classe à qual a imagem de prova pertence é a mais próxima de sua projeção no espaço de faces. Aumentando o K até o número total de faces permite obter uma métrica da probabilidade da classe correta estar entre as K classes mais próximas. Este algoritmo também é conhecido como *Cumulative Match Characteristic (CMC) curve*.

A Figura 18 apresenta as CMCs do sistema com e sem a equalização de histograma. Nela é perceptível o ganho que a etapa de equalização de histograma dá ao sistema. Sua interpretação dá-se da seguinte forma: o sistema sem a realização da equalização de histograma possui, aproximadamente, 75% de chance de informar corretamente a identidade de uma pessoa previamente cadastrada. Se considerar fornecer 5 “suspeitos” esta probabilidade aumenta para, aproximadamente, 87% de conter, dentre as 5, a pessoa correta. Já no sistema com equalização de histograma, as chances de identificar corretamente iniciam em 99,2%, aumentando ainda mais conforme a restrição de quantidade de “suspeitos” aumenta.

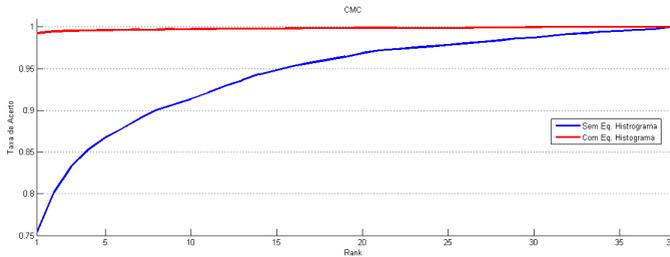


Figura 18 – Curvas de correspondência cumulativa (CMC) do sistema com (vermelho) e sem (azul) a etapa de equalização de histograma

4.3 IMPLEMENTAÇÃO EM SISTEMA EMBARCADO

Para avaliar a viabilidade de embarcar um sistema de RF o dispositivo alvo deste trabalho é um Raspberry Pi modelo B (PI, 2012). Ele foi escolhido pela relação entre sua capacidade de processamento e seu tamanho e pelo baixo custo, o que possibilita sua utilização em variadas aplicações.

Raspberry Pi é um pequeno computador que conta com processador de 700MHz, 512MB de memória RAM e GPU, custando apenas \$35 (modelo B, em 2012). Como Sistema Operacional, ele suporta algumas distribuições baseadas em *kernel* Linux. A distribuição utilizada foi a *Raspbian Jessie*. Possui interfaces USB, HDMI e Ethernet, permitindo a conexão de *webcams* e monitores e conexão com a rede. Não há nenhum disco rígido porém há um soquete para cartão SD, que realiza o armazenamento dos dados do computador. Sua alimentação é feita a 5V por conector MicroUSB, consumindo até 3.5W (PI, 2012). Alguns destes componentes de hardware podem ser vistos na Figura 19.

Para implementação do código optou-se por utilizar a biblioteca OpenCV (*Open Source Computer Vision Library*, em inglês). É uma biblioteca de código aberto baseada na licença BSD para aplicações de Visão Computacional. Ela possui interfaces para C, C++, Python, Java e MATLAB e suporta Windows, Linux, Mac OS, iOS e Android (ITSEEZ, 2000-2016). A utilização desta biblioteca minimiza os custos de desenvolvimento de código para processamento de imagens ao mesmo tempo que maximiza seu desempenho dado que otimiza a geração de código de máquina, quando possível.

Por utilizar SO Linux, pode ser utilizada uma vasta gama de linguagens de programação, *frameworks* e bibliotecas. Diante das pos-



Figura 19 – Raspberry Pi modelo B
Extraído de: (PI, 2012).

sibilidades e visando a execução mais rápida do código, optou-se pelo uso da linguagem C++. Para diminuir o tempo de desenvolvimento, inicialmente o código foi desenvolvido no ambiente Microsoft Visual Studio, em C++, com a biblioteca OpenCV. Conforme o esperado, o código foi portado sem problemas para o Raspberry Pi.

Como o treinamento é uma etapa sem restrições temporais rígidas, não mostrou-se necessário para esta avaliação a sua implementação no sistema embarcado. Desta forma, o treinamento foi realizado no MATLAB e as autofaces e os pesos de cada classe foram passados por intermédio de um arquivo de texto para o Raspberry Pi. Foram realizados experimentos referentes a etapa de reconhecimento, visando avaliar seu tempo de execução.

A câmera utilizada foi uma *webcam*, marca HP, modelo 4110 HD, com resolução máxima de 1900×1080 . Pelo experimento dispensar grandes resoluções, foi utilizada 640×360 . Ela conta com ajuste focal automático e adapta-se às condições de luz. Sua interface de comunicação é USB. Desta forma, a mesma câmera pôde ser empregada no *laptop*, na etapa de treinamento, e no Raspberry Pi, na etapa de reconhecimento.

Os resultados obtidos no sistema embarcado foram iguais aos obtidos no MATLAB, confirmando a corretude da implementação. A execução do reconhecimento, porém, mostrou-se mais lenta. Ela foi feita considerando a aquisição *frame a frame* a partir da *webcam* e realizando a detecção de faces, segmentação e reconhecimento de cada um, mostrando, ao final de cada ciclo, as faces em destaque conjuntamente com a sua identificação (Figura 11). Isto se deve à diferença de capacidade computacional entre o *laptop* utilizado na primeira parte e do Raspberry Pi.

O reconhecimento ocorreu sem atrasos à uma taxa de aproximadamente 1 *frame* analisado por segundo. Isto significa que todas as operações desde a aquisição de um *frame* até a identificação de todas as faces e a projeção da imagem com as identificações na tela ocorrem em menos de 1 segundo. Diminuir este tempo não foi objetivo deste trabalho, visto que este já é suficiente para validar o funcionamento do reconhecimento. Porém, taxas maiores podem ser obtidas se algumas funções nativas do Sistema Operacional forem desativadas. A projeção da imagem com as possíveis faces identificadas no monitor também prejudica o desempenho, aumentando a quantidade de computação necessária para iniciar a próxima verificação. Outro fator crítico observado foi a câmera utilizada. A aquisição de imagem por meio de diferentes modelos de *webcam* demora mais ou menos, dependendo da resolução escolhida, interface de comunicação utilizada pelo dispositivo, funcionalidades como foco automático, entre outras.

O sistema final pode ser empregado para realizar a identificação em tempo real de diversas pessoas em um ambiente complexo. Isto significa detectar e classificar, em imagens recém capturadas por uma câmera, uma ou mais pessoas e dispor a imagem de saída com as pessoas devidamente rotuladas. Para seu funcionamento, basta apenas realizar um treinamento prévio para cadastrar as pessoas no banco e ser alimentado por uma fonte ou bateria de 5V. As imagens de saída podem ser projetadas em uma tela a partir da conexão HDMI ou ainda podem ser armazenadas em memória para envio pela rede ou consulta futura.

5 CONSIDERAÇÕES FINAIS

PCA é uma técnica muito eficiente no campo da análise preditiva de dados. Porém, sua capacidade de extrapolação é baixa, deixando o sistema muito restrito às condições empregadas na aquisição de dados. Sua vantagem consiste em revelar estruturas subliminares simples em conjuntos de dados complexos, por meio de soluções da álgebra linear (SHLENS, 2014).

A abordagem pela técnica PCA não necessariamente possui o melhor desempenho em relação a outras técnicas. Porém, analisando sua facilidade de implementação, é uma das que melhor alia simplicidade e desempenho. Uma vez que necessita de treinamento, é necessário compor uma base de dados antes de realizar o reconhecimento. Então, de acordo com a aplicação, as imagens base devem seguir algumas restrições, visando aumentar o grau de confiabilidade da solução.

O desempenho de um sistema de reconhecimento facial está totalmente ligado a uma série de decisões de design, em todas as etapas. Isto significa que, apesar da técnica utilizada para realizar o reconhecimento, todo o sistema deve ser analisado e projetado para a aplicação específica, caso contrário, resultados insatisfatórios podem ser encontrados. Portanto, cabe ao desenvolvedor analisar suas prioridades para escolher os parâmetros que melhor as atendem.

Este trabalho apresentou uma solução de sistema embarcado voltado ao reconhecimento facial. Diante dos resultados obtidos, constatou-se que as ações de pré-processamento são importantes para alcançar um melhor desempenho. Além disto, o ganho apresentado pela realização da Equalização de Histograma foi superior ao de abordagens semelhantes (mesmo banco de faces e uso da técnica PCA). A taxa de acerto equipara-se a de outras implementações encontradas na bibliografia, qualificando este trabalho como uma solução ao problema.

O sistema embarcado resultante mostrou-se apto a realizar o reconhecimento, apesar do potencial de otimização não explorado, conforme mencionado. Pode-se afirmar que os objetivos foram concluídos com sucesso, sob as restrições adotadas.

5.1 TRABALHOS FUTUROS

São listados, nesta seção, algumas propostas para trabalhos futuros:

- Implementar outras técnicas, como LDA ou Redes Neurais e comparar os resultados;
- Implementar o reconhecimento facial em FPGA;
- Disponibilizar o sistema para uso em controle de acesso (verificação), como versão de testes;
- Utilizar câmera CMOS compatível com Raspberry Pi para avaliar possível gargalo referente à comunicação USB;
- Melhorar o algoritmo de reconhecimento em tempo real para que realize a inferência baseado nos *frames* anteriores, além do atual.

REFERÊNCIAS

- ABDI, H.; WILLIAMS, L. J. Principal component analysis. **Wiley Interdisciplinary Reviews: Computational Statistics**, Wiley Online Library, v. 2, n. 4, p. 433–459, 2010.
- BIOID. **BioID Face DataFace**. 2001. Online. Acessado em: 23/10/2015. Disponível em: <<https://www.bioid.com/About/BioID-Face-Database>>.
- BRAGA, L. F. Z. **Sistemas de Reconhecimento Facial**. 2013. 1–84 p.
- CONCI, A.; AZEVEDO, E.; LETA, F. R. **Computação gráfica**. 1. ed. Rio de Janeiro: Elsevier, 2008. ISBN 85-352-2329-0.
- GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. 2. ed. Upper Saddle River, N.J: Prentice Hall, 2002. ISBN 0201180758.
- GRGIC, M.; DELAC, K. **Face Recognition Homepage**. 2005–2016. Online. Acessado em: 08/12/2015. Disponível em: <<http://www.face-rec.org/databases/>>.
- HJELMÅS, E.; LOW, B. K. Face detection: A survey. **Computer vision and image understanding**, Elsevier, v. 83, n. 3, p. 236–274, 2001.
- HUANG, D. et al. Local binary patterns and its application to facial image analysis: A survey. **IEEE Trans. Syst., Man, Cybern. C**, Institute of Electrical & Electronics Engineers (IEEE), v. 41, n. 6, Nov 2011. ISSN 1558-2442. Disponível em: <<http://dx.doi.org/10.1109/TSMCC.2011.2118750>>.
- ITSEEZ. **OpenCV Website**. 2000–2016. Online. Acessado em: 07/12/2015. Disponível em: <<http://opencv.org/>>.
- JALILMAZLOUM et al. A novel bidirectional neural network for face recognition. In: **2012 2nd International eConference on Computer and Knowledge Engineering (ICCKE), October 18-19, 2012**. Tehran, Iran: IEEE, 2012. p. 18–23.
- JESORSKY, O.; KIRCHBERG, K. J.; FRISCHHOLZ, R. W. Robust face detection using the hausdorff distance. In: SPRINGER. **Audio**

and Video-based Biometric Person Authentication. Halmstad, Sweden, 2001. p. 90–95.

KHANAM, A.; SHAFIQ, M. Z.; AKRAM, M. U. Fuzzy based facial expression recognition. **2008 Congress on Image and Signal Processing**, Institute of Electrical & Electronics Engineers (IEEE), 2008. Disponível em: <<http://dx.doi.org/10.1109/CISP.2008.694>>.

LEARNED-MILLER, E. et al. **Labeled Faces in the Wild: A Survey.** 2015.

LEE, K.; HO, J.; KRIEGMAN, D. **The Yale Face Database B.** 2001. Online. Acessado em: 21/03/2016. Disponível em: <<http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/YaleFaceDatabase.htm>>.

LEE, K.; HO, J.; KRIEGMAN, D. Acquiring linear subspaces for face recognition under variable lighting. **IEEE Trans. Pattern Anal. Mach. Intelligence**, v. 27, n. 5, p. 684–698, 2005.

MARTÍNEZ, A. M.; KAK, A. C. Pca versus lda. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, IEEE, v. 23, n. 2, p. 228–233, 2001.

MILANO, D. de; BARROZO, L.; HONORATO. Visão computacional. 2010.

MOGHADDAM, B.; PENTLAND, A. Probabilistic visual learning for object detection. In: IEEE. **Fifth International Conference on Computer Vision.** Cambridge, Massachusetts, 1995. p. 786–793.

MOON, H.; PHILLIPS, P. J. Computational and performance aspects of pca-based face-recognition algorithms. **Perception-London**, [London: Pion Ltd.], v. 30, n. 3, p. 303–322, 2001.

NASA/IPAC. **Birth of Stars & Protoplanetary Systems.** 2002. Online. Acessado em: 18/05/2016. Disponível em: <<http://jwst.nasa.gov/birth.html>>.

PEARSON, K. On lines and planes of closest fit to systems of points in space. **The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science**, Taylor & Francis, v. 2, n. 11, p. 559–572, 1901.

PI, R. **Raspberry Pi.** 2012. 104 p.

PRAZERES, S. G. **Reconhecimento de Faces Utilizando Redes Neurais Autoassociativas**. Dezembro 2010. Monografia, Escola Politécnica de Pernambuco.

RAO, K. R. **Optimal Decorrelation and the KLT**. 2015.

SANTANA, L. M. Q. de; GOMES, F. R.; SANTOS, T. S. R. Uma análise do processo reconhecimento facial. **Caderno de Graduação-Ciências Exatas e Tecnológicas-UNIT**, v. 2, n. 2, p. 49–58, 2014.

SARIYANIDI, E.; GUNES, H.; CAVALLARO, A. Automatic analysis of facial affect: A survey of registration, representation, and recognition. **IEEE Trans. Pattern Anal. Mach. Intell.**, Institute of Electrical & Electronics Engineers (IEEE), v. 37, n. 6, p. 1113–1133, Jun 2015. ISSN 2160-9292. Disponível em: <<http://dx.doi.org/10.1109/TPAMI.2014.2366127>>.

SHAH, M. **Fundamentals of Computer Vision**. Orlando, FL, Dezembro 1997.

SHLENS, J. A tutorial on principal component analysis. **arXiv preprint arXiv:1404.1100**, Abril 2014.

SINGH, G.; CHHABRA, I. Human face recognition through moment descriptors. In: **Engineering and Computational Sciences (RAECS), 2014 Recent Advances in**. Chandigarh, India: IEEE, 2014. p. 1–6.

SIROVICH, L.; KIRBY, M. Low-dimensional procedure for the characterization of human faces. **JOSA A**, Optical Society of America, v. 4, n. 3, p. 519–524, 1987.

SOBRINHO, L. P. **Reconhecimento Facial 2D para Sistemas de Autenticação em Dispositivos Móveis**. 2010. Monografia, Universidade Regional de Blumenau.

TURK, M.; PENTLAND, A. Eigenfaces for recognition. **Journal of cognitive neuroscience**, MIT Press, v. 3, n. 1, p. 71–86, 1991.

TURK, M.; PENTLAND, A. P. et al. Face recognition using eigenfaces. In: IEEE. **Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on**. Cambridge, MA: IEEE, 1991. p. 586–591.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: IEEE. **Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on**. Kauai, Hawaii, 2001. v. 1, p. 511–518.

ZHAO, W. et al. Face recognition: A literature survey. **ACM computing surveys (CSUR)**, ACM, v. 35, n. 4, p. 399–458, 2003.

APÊNDICE A – Pseudocódigo de avaliação do sistema

Pseudocódigo 1 Avaliação de desempenho do sistema a partir de um Banco de Imagens com indivíduos “conhecidos” e “desconhecidos” pelo treinamento

```

1: função AVALIAR DESEMPENHO
2:   VP = 0                                ▷ inicializando as variáveis
3:   VN = 0
4:   FN = 0
5:   FP1 = 0
6:   FP2 = 0
7:   para f ← 1 até QTD_AMOSTRAS faça
8:     Obter imagem de teste
9:     Detectar e Segmentar a face
10:    Fazer equalização de histograma
11:    Subtrair a face média
12:    Projetar face de teste no espaço de faces
13:    mais_proximo = 1
14:    para c ← 2 até QTD_CLASSES faça
15:       $\epsilon[c]$  = Calcular DIFS entre a classe c e a face de prova
16:      se  $\epsilon[c] < \epsilon[\text{mais\_proximo}]$  então
17:        mais_proximo = c ▷ Calculando qual é a classe mais
18:        “próxima”
19:      fim se
20:    fim para
21:    se  $\epsilon[\text{mais\_proximo}] > \text{THRESHOLD}$  então
22:      se f pertencer a uma classe desconhecida então
23:        VN = VN + 1
24:      senão
25:        FN = FN + 1
26:      fim se
27:    senão
28:      se mais_proximo for a classe correta então
29:        VP = VP + 1
30:      senão
31:        se f pertencer a uma classe desconhecida então
32:          FP2 = FP2 + 1
33:        senão
34:          FP1 = FP1 + 1
35:        fim se
36:      fim se
37:    fim para
38:    SENSIBILIDADE = VP/(VP+FN+FP1)
39:    ESPECIFICIDADE = VN/(VN+FP2)
40: fim função

```
