

UNIVERSIDADE CANDIDO MENDES – UCAM
PROGRAMA DE PÓS-GRADUAÇÃO EM PESQUISA OPERACIONAL E
INTELIGÊNCIA COMPUTACIONAL CURSO DE MESTRADO EM PESQUISA
OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL

FLÁVIO IZO

**PROTÓTIPO DE UM SISTEMA WEB PARA
RECONHECIMENTO FACIAL UTILIZANDO A
METODOLOGIA EIGENFACES**

CAMPOS DOS GOYTACAZES, RJ
2015

UNIVERSIDADE CANDIDO MENDES – CAMPOS
PROGRAMA DE PÓS-GRADUAÇÃO EM PESQUISA OPERACIONAL E
INTELIGÊNCIA COMPUTACIONAL CURSO DE MESTRADO EM PESQUISA
OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL

FLÁVIO IZO

**PROTÓTIPO DE UM SISTEMA WEB PARA
RECONHECIMENTO FACIAL UTILIZANDO A
METODOLOGIA EIGENFACES**

Dissertação apresentada ao Programa de Pós-Graduação em Pesquisa Operacional e Inteligência Computacional, da Universidade Candido Mendes – Campos/RJ, para obtenção do grau de MESTRE EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL.

Orientador: Prof. Ítalo de Oliveira Matias, D.Sc.

CAMPOS DOS GOYTACAZES, RJ

Julho de 2015

FLÁVIO IZO

**PROTÓTIPO DE UM SISTEMA WEB PARA
RECONHECIMENTO FACIAL UTILIZANDO A
METODOLOGIA EIGENFACES**

Dissertação apresentada ao Programa de Pós-Graduação em Pesquisa Operacional e Inteligência Computacional, da Universidade Candido Mendes – Campos/RJ, para obtenção do grau de MESTRE EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL.

Aprovada em 22 de Agosto de 2015.

BANCA EXAMINADORA

Prof. Ítalo de Oliveira Matias, D.Sc. – orientador
UCAM-Campos

Prof. Dalessandro Soares Vianna, D.Sc.
UCAM-Campos

Prof. Marcilene de Fátima Dianin Vianna, D.Sc.
UFF-Rio das Ostras

CAMPOS DOS GOYTACAZES, RJ

2015

À minha esposa e minha filha que,
com braços abertos e sorriso no
rosto, recarregavam as energias
necessárias à realização deste
projeto.

AGRADECIMENTOS

A Deus, em primeiro lugar, por tudo.

A minha linda esposa, Suellen, pelo amor, paciência, carinho e incentivo.

A minha querida filha Maria Antonia, que nasceu durante o estudo do mestrado e a cada sorriso iluminava meu caminho e servia de incentivo para que o terminasse o quanto antes.

Aos meus familiares, principalmente meus pais Maria e Antonio, os quais foram fundamentais para a conclusão do mestrado.

Aos amigos que foram companheiros de jornada.

Aos professores da Universidade Candido Mendes pela atenção, amizade e dedicação.

E, principalmente, ao meu orientador e amigo professor D.Sc. Ítalo de Oliveira Matias pela dedicação, compreensão e norteamto das minhas pesquisas.

“De tudo ficam três coisas:
A certeza de que estamos sempre começando...
A certeza de que é preciso continuar...
A certeza de que seremos interrompidos antes de
terminar...
Façamos da interrupção um caminho novo...
Da queda um passo da dança...
Do medo, uma escada...
Do sonho, uma ponte...
Da procura... um encontro”.

Fernando Sabino

RESUMO

PROJETO E CONSTRUÇÃO DE UM SISTEMA WEB PARA RECONHECIMENTO FACIAL UTILIZANDO A METODOLOGIA EIGENFACES

O reconhecimento facial é uma área da visão computacional que está sempre presente em estudos de inteligência artificial. Este projeto tem como objetivo descrever um método de reconhecimento facial como uma alternativa para um sistema web. O objeto em estudo é o método eigenfaces, desenvolvido por Turk e Pentland e que é um processo aperfeiçoado do reconhecimento de padrões denominado “Análise dos Componentes Principais”, ou PCA. A eigenfaces possibilita a classificação de imagens à partir do cálculo de distância vetorial entre a imagem prova que está sendo testada e as imagens da base de treinamento que estão em um espaço de faces. Através de um valor limiar (*threshold*) é possível perceber se a imagem testada é de alguma pessoa presente na base de treinamento (face reconhecida) ou se a face não está presente (face não reconhecida). Além da descrição do método eigenfaces, este projeto apresenta algumas simulações de reconhecimento de faces e expõe os resultados obtidos. Em trabalhos futuros, pretende-se aplicar o sistema em um projeto piloto e utilizar outras técnicas de reconhecimento facial para fins de comparação.

Palavras chave: Eigenfaces; Classificação de Imagens; PCA; Detecção Facial; Visão Computacional

ABSTRACT

DESIGN AND CONSTRUCTION OF A SYSTEM FOR WEB FACIAL RECOGNITION USING THE METHODOLOGY EIGENFACES

Facial recognition is an area of computer vision that is always present in artificial intelligence studies. This project aims to describe a method of facial recognition as an alternative to a web system. The object under study is the eigenfaces method, developed by Turk and Pentland, and that is an improved method of pattern recognition called "Principal Component Analysis", or PCA. The eigenfaces enables the classification of the images from the distance vector calculus between the test image being tested and images of the training base who are on a space faces. Through a threshold value (threshold) you can see if the image is tested in any person present at the training base (recognized face) or if the face is not present (not recognized face). Besides the description of the eigenfaces method, this project presents some recognition simulations faces and exposes the results. In future work, we intend to apply the system in a pilot project and use other facial recognition techniques for comparative purposes.

Keywords: Eigenfaces; Image Classification; PCA; Facial detection; Computer Vision

LISTA DE FIGURAS

FIGURA 1 - Arquitetura básica de um sistema de reconhecimento facial	13
FIGURA 2 – À esquerda exemplo de dados iniciais. À direita os dados após a PCA	20
FIGURA 3 – Duas áreas da biometria.	25
FIGURA 4 – Esquema básico de reconhecimento facial.....	26
FIGURA 5 – Comparação de fotos de Galton.....	27
FIGURA 6 – Esquema de detecção e reconhecimento do protótipo do sistema ..	32
FIGURA 7 – Exemplos de imagens da base ORL.	34
FIGURA 8 – Exemplos de imagens da base AUT..	35
FIGURA 9 – Características de <i>Haar</i> mais usadas em detecção de faces.	36
FIGURA 10 – Associação de padrões com partes de uma face.....	37
FIGURA 11 – Detecção de face, utilizando os olhos como padrão.	37
FIGURA 12 – Transformação de matriz de imagem em vetor coluna.....	40
FIGURA 13 – Exemplos de eigenfaces.....	42
FIGURA 14 – Projeção de imagens com base nos <i>eigenfaces</i>	45
FIGURA 15 – Espaço de faces simplificado.....	46
FIGURA 16 – Protótipo da interface do módulo de autenticação.	49
FIGURA 17 – Protótipo da interface do módulo de cadastro.....	50
FIGURA 18 – Exemplo de faces da base AUT.....	51
FIGURA 19 – Transformação de matriz de imagem em vetor coluna.....	52
FIGURA 20 – Imagem Média gerada após o passo B.	53
FIGURA 21 – Exemplo de imagens geradas após a subtração da média.	54
FIGURA 22 – Exemplos de eigenfaces.....	55

LISTA DE QUADROS E TABELAS

Tabela 1: Conjunto de dados iniciais. Fonte: Smith (2002).....	18
Tabela 2: Média de X e Y ao lado esquerdo e o conjunto de dados iniciais subtraído da média, ao lado direito, denominado aqui de matriz A . Fonte: Smith (2002)	18
Tabela 3: Dois autovalores juntamente com os dois autovetores construídos à partir da matriz de covariância C . Fonte: Smith (2002)	19
Tabela 4: Descrição das bases de faces utilizadas. Fonte: ORL (1994) e o Autor.	35
Tabela 5 – Resultados da simulação da aplicação do método eigenfaces na base TOT.	59
Tabela 6 – Resultados da simulação com base em 4 testes e de 1 a 6 imagens de treinamentos.	61
Tabela 7 – Resultados da simulação da aplicação do método eigenfaces na base TOT, ordenados pelos melhores resultados.	61
Tabela 8 – Resultados da simulação da aplicação do método eigenfaces na base AUT.....	63
QUADRO 1: DISTRIBUIÇÃO DE REGISTROS POR VEÍCULO DE PUBLICAÇÃO.....	28
QUADRO 2: DISTRIBUIÇÃO DE REGISTROS POR PAÍSES.....	29
QUADRO 3: DISTRIBUIÇÃO DE REGISTROS POR VEÍCULO DE PUBLICAÇÃO.....	31
GRÁFICO 1: DISTRIBUIÇÃO DAS PUBLICAÇÕES POR ANO	30
GRÁFICO 2 - SIMULAÇÃO COM BASE EM 4 TESTES E DE 1 A 6 IMAGENS DE TREINAMENTOS	61

LISTA DE ABREVIATURAS E SIGLAS

AUT: Base de fotos do autor

BSD: *Berkeley Software Distribution*

IA: Inteligência Artificial

ICA: *Independent Component Analysis*

JPEG (JPG): *Joint Photographic Experts Group*

NFL: *Nearest Feature Lines*

OPENCV: *Open Computer Vision Library*

PCA: *Principal Components Analysis*

PIL: *Python Image Library*

PGM: *Portable Gray Map*

ORL: Base de fotos da AT & T

SUMÁRIO

1. INTRODUÇÃO	12
1.1 CONTEXTUALIZAÇÃO	12
1.2 OBJETIVOS	14
1.2.1 Geral	14
1.2.2 Específicos	14
1.3 ESTRUTURA DO TRABALHO	15
2. REVISÃO DA LITERATURA	16
2.1 <i>PRINCIPAL COMPONENTS ANALYSIS</i> (PCA)	16
2.2 <i>EIGENFACES</i>	21
2.3 RECONHECIMENTO FACIAL	22
2.4 ESTUDO BIBLIOMÉTRICO	28
3. MATERIAIS E MÉTODOS	32
3.1 AMBIENTE DE DESENVOLVIMENTO	33
3.2 BANCO DE FOTOS (FACES)	34
3.3 DETECÇÃO FACIAL	36
3.4 RECONHECIMENTO FACIAL: <i>EIGENFACES</i>	38
3.4.1 Como calcular o eigenfaces?	39
3.5 CLASSIFICAÇÃO COM DISTÂNCIA EUCLIDIANA	47
4. SISTEMA DE RECONHECIMENTO FACIAL PROPOSTO	48
4.1 DESCRIÇÃO GERAL	48
4.2 MÓDULO CADASTRO DE PESSOAS	50
4.3 MÓDULO DE AUTENTICAÇÃO ATRAVÉS DE RECONHECIMENTO FACIAL	51
4.3.1 Fase de Treinamento:	52
4.3.2 Fase de Classificação	55
5. RESULTADOS E DISCUSSÃO	59
6. CONSIDERAÇÕES FINAIS	64
REFERÊNCIAS	65

1. INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

A inteligência artificial (IA) tem tido crescente utilização entre os projetos de pesquisa que testam a simulação da mente humana. Muitos cientistas querem aperfeiçoar a maneira como os sistemas funcionam, e chegar próximo do raciocínio humano. Inserida na IA, está a Visão Computacional que tem como objetivo principal o estudo de um sistema de visão artificial que simule o sistema visual humano. Shapiro & Stockman (2001) destacam que a visão computacional é imprescindível para, no futuro, se consiga desenvolver um ser humano com vida artificial.

Kim et al. (2013) analisam que sistemas com visão computacional podem ser utilizados em uma grande variedade de situações: identificação criminal, sistemas de segurança, interação do homem com a máquina, controles de acesso, validação de cartões de crédito, entre outros.

Neste contexto, os mesmos autores citam como importante mecanismo para a execução da visão computacional, a biometria, que analisa características biológicas do ser humano para identificá-lo. Assim, parte-se do princípio de que cada ser humano possui um conjunto de características que o torna diferente dos demais. Fazem parte dessas características a impressão digital, a íris, a geometria da mão e dos dedos, a retina e o formato do rosto.

Nesta dissertação utiliza-se como característica biométrica a face, pois essa foi considerada como parâmetro de entrada do módulo de visão computacional. A análise biométrica do formato do rosto consiste em 2 processos: detecção e reconhecimento

facial. A detecção facial está fundamentada em detectar se existe e em qual parte da imagem está a face. Assim, deve-se extraí-la da imagem e separá-la. É um dos passos mais importantes do reconhecimento facial (CORCORAN e IANCU, 2011). Já o reconhecimento facial, identifica quem é a pessoa a partir da face detectada, ou informa que a pessoa não existe na base de dados.

A representação do funcionamento do sistema de reconhecimento de faces pode ser visualizado na Figura 1, na qual pode-se observar a entrada da imagem (*image*), que sofre o processo de detecção facial (*face detection*), e em seguida é feita a normalização desta mesma imagem (*pre-processing*). A próxima etapa (*feature extraction*) serve para extrair as informações da face de entrada. Por fim, é feito o reconhecimento facial, onde a base de dados contendo as imagens é treinada (*training*) e em seguida sofre a avaliação (*evaluation*) para descobrir se a imagem de entrada existe na base de dados.

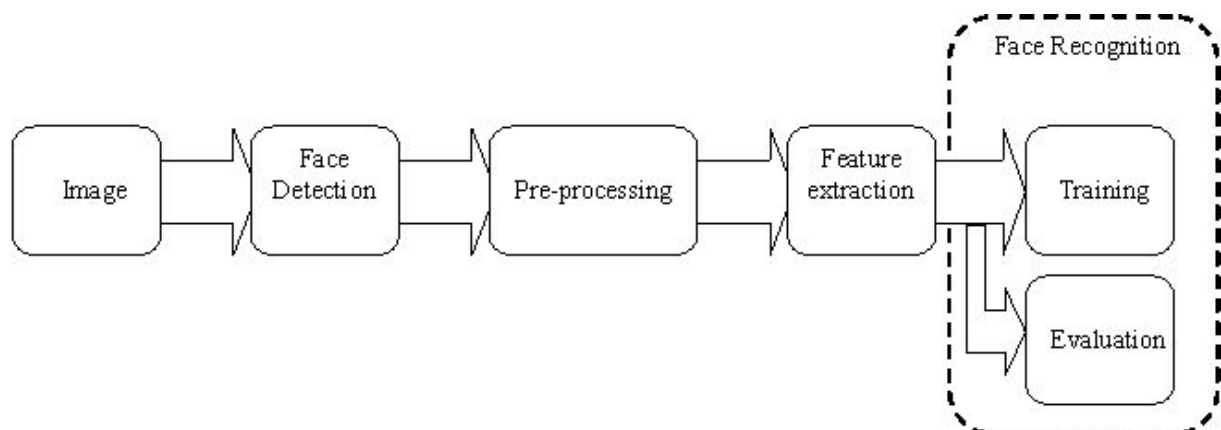


FIGURA 1 - Arquitetura básica de um sistema de reconhecimento facial. Fonte: CORCORAN e IANCU (2011)

Dentre as técnicas para reconhecimento facial, pode-se destacar as *eigenfaces*, *eigenfaces* probabilísticas, *fisherfaces*, Nearest Feature Lines (NFL) e Independent Component Analysis (ICA). Neste trabalho, foi escolhido como objeto de estudo as *eigenfaces*, apresentada inicialmente por Turk e Pentland (1991) que foram motivados pela técnica de Sirovich e Kirby (1987) e Kirby e Sirovich (1990) onde foi feita a representação de imagens faciais utilizando PCA.

Dessa forma, essa pesquisa tem como objetivo apresentar uma proposta de sistema web de reconhecimento facial utilizando *eigenfaces*, tendo como base as métricas de Turk e Pentland (1991), que são referências de desempenho para novas técnicas até os dias atuais. O sistema deve receber como entrada a imagem capturada

através de uma webcam e utilizar o algoritmo de Turk e Pentland para determinar se o rosto obtido está presente na base de dados, caso esteja, o programa permite que a pessoa acesse a área restrita do sistema. Caso contrário, o acesso não é permitido.

1.2 OBJETIVOS

1.2.1 Geral

O objetivo geral deste trabalho é apresentar o método *eigenfaces* como meio para reconhecimento de faces e gerar um protótipo de sistema web para reconhecimento facial.

1.2.2 Específicos

Esta pesquisa tem como objetivo específicos:

1. Realizar pesquisa bibliográfica em materiais que utilizam o *eigenfaces* como recurso para reconhecimento facial;
2. Gerar um protótipo de sistema web para reconhecimento facial;
3. Testar a funcionalidade deste método através do desenvolvimento de um protótipo de sistema web utilizando a linguagem *Python*;
4. Avaliar através de estatísticas, a taxa de acertos do sistema web no processo de reconhecimento facial utilizando o *eigenfaces*.

1.3 ESTRUTURA DO TRABALHO

O trabalho foi organizado em seis capítulos.

O primeiro é a introdução, onde foi apresentado a contextualização do tema, a formulação do problema em análise e os objetivos da pesquisa (geral e específicos).

No capítulo 2 fez-se a revisão da literatura acerca de *Principal Component Analysis* (PCA), *Eigenfaces* e *Reconhecimento Facial*.

O capítulo 3 apresenta a descrição das tecnologias utilizadas e descrição dos dados (faces) empregados no processo de reconhecimento facial.

No capítulo 4 é descrito o processo de desenvolvimento do protótipo, apresentando detalhadamente a proposta de um sistema web para reconhecimento facial utilizando *Eigenfaces*.

O capítulo 5 apresenta os resultados obtidos através da simulação do sistema, abordando a taxa de acerto como principal ponto a ser discutido.

No capítulo 6 é apresentada as considerações finais e sugestões de pesquisas futuras.

2. REVISÃO DA LITERATURA

Neste capítulo é apresentada a fundamentação teórica necessária ao desenvolvimento deste estudo. Deste modo, são abordados os principais conceitos relacionados à PCA, Eigenfaces e Reconhecimento Facial, descrevendo os principais conceitos de cada um dos tópicos. Por fim, é feita uma análise bibliométrica destacando as principais estatísticas encontradas a respeito do uso do método eigenfaces no processo de reconhecimento facial.

2.1 *PRINCIPAL COMPONENTS ANALYSIS (PCA)*

Os dados são fundamentais para a realização de muitas pesquisas, no entanto, o excesso desses dados pode atrapalhar e até mesmo dificultar a sua análise. Sabendo-se que o excesso de dados é um problema, muito pesquisadores adotam técnicas de descarte de parte desses dados. Porém, é preciso ter cuidado no descarte, pois às vezes um dado muito importante é deixado de lado e essa atitude pode influenciar no resultado final.

Mendonça (2005) cita que são diversas as pesquisas em variados domínios de aplicação que encontram um problema em comum, o excesso de dados para serem analisados. Por essa razão é comum encontrar pesquisas nas quais são utilizados algum processo de reduzir o número de informações.

Uma das técnicas de descarte é a Análise dos Componentes Principais (PCA, em inglês *Principal Component Analysis*).

A Análise dos componentes principais (PCA, em inglês *principal component*

analysis) é um método da álgebra linear que facilita o descarte de dados de maneira coerente e precisa. Uma de suas principais características é no processo de detecção de padrões.

De acordo com Nor'aini, Raveendran e Selvanathan (2005), a PCA tem mostrado um desempenho promissor com dados integrais, porém o tempo de computação durante o treinamento deve sempre ser levado em conta.

Jolliffe (2002) corrobora dizendo que a Análise de Componentes Principais (PCA) é um dos mais antigos e conhecidos métodos de análise multivariada.

Campos (2000) cita que a PCA, também é conhecida por transformada de *Hotelling* e por expansão de *Karhunen-Loève*.

Segundo Smith (2002) Smith e Shlens (2003) a *Principal Component Analysis*, análise de componentes principais ou expansão de *Karhunen-Loève*, talvez seja uma das maiores contribuições da álgebra linear e estatística aplicada.

Gonzalez (2000) afirma que quando visto pelo ponto de vista de transformada, o seu resultado é uma mudança de base com uma projeção em um novo espaço onde cada característica esteja livre de redundância e esteja expresso em ordem de variância ou contribuição ao conjunto de dados.

“Muitos algoritmos baseados em PCA têm sido aplicados em problemas de reconhecimento facial. A ideia principal da PCA é encontrar os vetores que melhor representam a distribuição da face dentro do espaço de imagens”. (ABDELWAHAB, 2012).

Agarwal (2010) afirma que quando se tem vetores de alta dimensão, deve-se adotar uma técnica, para reduzir a quantidade de características de uma imagem.

O PCA reconstrói um conjunto pequeno de componentes que resumem os dados originais, reduzindo a dimensionalidade dos mesmos, preservando as características mais significantes (características principais), afirma Kshirsagar (2011).

De acordo com SMITH (2002), PCA é uma poderosa ferramenta para análise de dados e deve seguir os seguintes passos:

- a) Obter os dados ou as amostras de vetores.

São os dados obtidos para a resolução do problema em questão. Neste exemplo, será utilizado o exemplo em 2-dimensões para facilitar a plotagem, seguindo a metodologia de Campos (2000). Uma amostra desses dados pode ser visualizada na tabela 1.

X	Y
2,5	2,4
0,5	0,7
2,2	2,9
1,9	2,2
3,1	2,0
2,3	2,7
2	1,6
1	1,1
1,5	1,6
1,1	0,9

Tabela 1: Conjunto de dados iniciais. Fonte: Smith (2002)

b) Calcular o valor médio e subtrair o valor médio de cada um dos itens.

O cálculo do valor médio é a média aritmética simples. Após a criação do vetor média, é feita a subtração de cada item pelo valor médio calculado. Tanto a média quanto os valores subtraídos podem ser observados na tabela 2.

X	Y	X	Y
1,81	1,81	0,69	0,59
		-1,31	-1,11
		0,39	1,09
		0,09	0,39
		1,29	0,19
		0,49	0,89
		0,19	-0,21
		-0,81	-0,71
		-0,31	-0,21
		-0,71	-0,91

Tabela 2: Média de X e Y ao lado esquerdo e o conjunto de dados iniciais subtraído da média, ao lado direito, denominado aqui de matriz A. Fonte: Smith (2002)

- c) Calcular a matriz de covariância.

O cálculo da matriz de variância obedece os dados ajustados da tabela 2. Para o cálculo da covariância é demonstrada a fórmula abaixo, onde N é a dimensão da matriz e A é a matriz demonstrada na tabela 2, e A^T é a transposta da matriz A .

$$C = \frac{1}{N-1} A^T \cdot A \quad (1)$$

- d) Calcular os autovetores e autovalores da matriz de covariância.

Só é possível calcular autovetores e autovalores em matriz quadradas, o que se observa neste procedimento. É muito importante o cálculo correto desses valores, haja vista que são eles que nos mostrarão as informações mais importantes acerca dos nossos dados.

Seguindo o exemplo, os autovalores e autovetores são observados abaixo, na tabela 3:

Autovalores	Auto vetores	
	X	Y
0,0490833989	-0,735178656	-0,677873399
1,28402771	0,677873399	-0,735178656

Tabela 3: Dois autovalores juntamente com os dois autovetores construídos à partir da matriz de covariância C . Fonte: Smith (2002)

- e) Escolher os componentes e formar um vetor de características

Nesta etapa é onde se observa a compreensão de dados e redução de dimensão. Pode-se verificar na seção anterior que os autovalores são bastante diferentes. Na realidade o autovetor com maior autovalor associado é o principal componente do conjunto de dados.

Em geral, quando os autovetores são encontrados a partir da matriz de covariância, o próximo passo é ordená-los pelos autovalores, do mais alto para o mais baixo. Isto fará com que os componentes fiquem em ordem de importância. Assim. Será possível ignorar os componentes de menor importância.

f) Derivar um novo conjunto de dados

Uma vez que foram escolhidos os componentes principais (autovetores), deve-se multiplicar a transposta desse vetor pelo conjunto original de dados. Esses serão os dados finais indicados pelo método PCA.

A partir do último passo, obtém-se novos dados, porém esses dados manterão o mesmo contexto dos dados iniciais.

Na Figura 2 pode-se observar que na imagem da esquerda está a plotagem dos dados originais (que geraram autovetores) e na imagem da direita a plotagem após a aplicação da PCA (utilizando um autovetor). Os dados, após a transformação utilizando o mais significativo autovetor, foram restaurados e mantiveram a mesma significância.

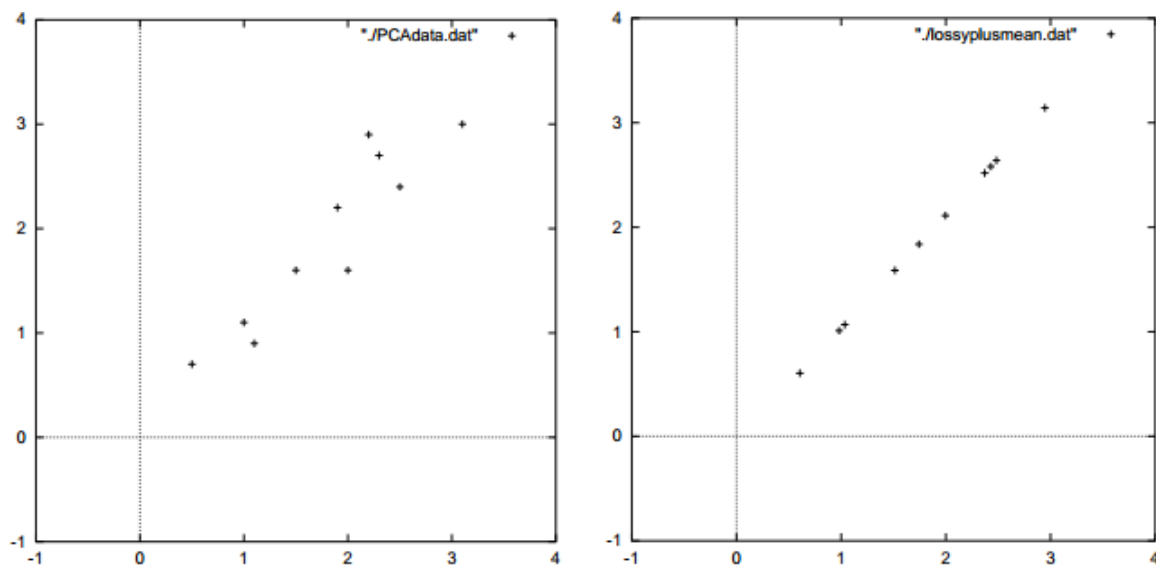


FIGURA 2 – À esquerda exemplo de dados iniciais. À direita os dados após a PCA. Fonte: SMITH (2002)

Como pode-se perceber, se vetores são de dimensão N^2 , serão criados N^2 autovetores. Porém, na prática, pode e deve-se descartar os autovetores com menores autovalores associados (menos significativos). Mesmo com esse descarte, a reconstrução do objeto será quase idêntica à imagem original.

2.2 EIGENFACES

Segundo Yang, Kriegman e Ahuja (2002), *eigenfaces* é um método Baseado na Aparência, ou seja, que representa as técnicas que não necessitam de conhecimento prévio sobre a característica a ser reconhecida. Normalmente, para que o objetivo seja realizado com sucesso, necessita-se de várias imagens e, a assim, aprendem somente o que é necessário para conseguir realizar o reconhecimento da característica de interesse.

Turk e Pentland (1991a) citam que, de modo geral, mesmo as imagens ocupando um espaço de alta dimensão, elas não estão descritas de modo aleatório, permitindo assim, que possam ser reescritas em um espaço de dimensão menor.

Para isso, Yang, Kriegman e Ahuja (2002) corrobora dizendo que, para isso ser possível, o *eigenfaces* busca os principais componentes em um espaço facial, ou seja, os autovetores que descrevem o rosto de uma pessoa. Isso é feito através da aproximação dos autovetores da matriz autocorrelação da imagem.

“Eigenfaces são um conjunto de autovetores utilizados em visão computacional para resolver problemas de reconhecimento facial” (LATA, 2009).

Kim (1996) afirma que autovetores associados com os maiores autovalores refletem na maior variação, e os associados com os menores autovalores representam a menor variação. Autovalores diminuem exponencialmente, de modo que cerca de 90% da variação total está contida nos primeiros 5% a 10% dos autovetores.

Smith (2002) lembra que só é possível achar autovetores com matriz quadrada e não são todas as matrizes quadradas que possuem autovetores. Em relação à quantidade de autovetores, o autor cita que uma matriz $n \times n$ possui n autovetores.

Turk e Pentland (1991a) definem que as *eigenfaces* são definidas como um espaço N-dimensional de características das faces composto por autovetores (*eigenvectors*), os quais são vetores que contêm as características mais relevantes da face, ou seja, os autovalores (*eigenvalues*).

Segundo Lata (2009), esse método se refere a uma abordagem baseada em aparência voltado para o reconhecimento de face que procura captar a variação em uma coleção de imagens de faces e usar essa informação para codificar e comparar imagens de rostos individuais de uma maneira holística.

A ideia de utilizar eigenfaces no reconhecimento de faces surgiu inicialmente com a técnica desenvolvida por Sirovich e Kirby (1987) que desejavam representar imagens de faces de forma eficiente utilizando PCA.

Lata (2009) completa ainda que a *eigenfaces* é usada, comumente, para:

- a. Extrair as informações relevantes facial, que pode ou não estar diretamente relacionada à intuição humana de características faciais, tais como os olhos, nariz e lábios. Uma maneira de fazer isso é capturar, estatisticamente, a variação entre as faces.
- b. Represente imagens de faces de maneira eficiente. Para reduzir a complexidade computacional e espaço, de cada imagem da face é representada utilizando um pequeno número de dimensões.

Perlibakas (2004) cita que a forma mais comum de calcular a distância entre vetores é através da distância Euclidiana, mas outras formas de medição também podem ser utilizadas, como a distância de Manhattan. O autor cita ainda, que se A e B são dois vetores de tamanho D, as fórmulas seriam assim:

1) Distância Euclidiana:

$$d(A, B) = \sqrt{\sum_{i=1}^D (a_i - b_i)^2} = ||A - B|| \quad (2)$$

2) Distância de Manhattan:

$$d(A, B) = \sum_{i=1}^D |a_i - b_i| \quad (3)$$

2.3 RECONHECIMENTO FACIAL

Pereira (2007) alega que o cérebro humano é capaz de identificar diversos tipos de padrões, como formas, cores, tamanhos sem grandes problemas. Além de ser apto também à identificação de padrões mais complexos podendo diferenciar sem grandes dificuldades a face de uma pessoa dentre várias.

Segundo Bicalho (2013), o reconhecimento facial é a capacidade de identificar um indivíduo com base nas características biométricas de sua face. Apesar desse processo ser feito de maneira rápida e intuitiva pelos seres humanos, o autor informa que a automatização desse processo vem sendo alvo de muitos estudos nas últimas décadas devido a sua vasta área de aplicação, por exemplo: segurança das vias e verificação de documentos de identidade.

Outros autores, como Ekman (1993) defendem que o objetivo principal do reconhecimento facial é identificar, porém, pode ir além, percebendo o estado emocional de uma pessoa apenas observando suas características (expressões) faciais.

Assim, para que todo esse reconhecimento facial ocorra é preciso que alguns cenários sejam estabelecidos. Normalmente são dois: verificação e identificação.

LI e Jain (2011) e Bicalho (2013) afirmam que, normalmente, um sistema de reconhecimento facial atua com dois cenários: verificação (ou autenticação) e identificação. No primeiro cenário verifica se há similaridade entre duas imagens de faces. Já o segundo, a imagem de entrada é comparada com várias imagens da base de faces para verificar a semelhança com alguma face, através de uma medição, e assim identificar o indivíduo.

De acordo com Pentland e Choudhury (2000) tem sido alto o investimento na área de reconhecimento facial, sendo que várias empresas, agências governamentais, cassinos e similares já fazem uso desse tipo de monitoramento.

Yang, Kriegman e Ahuja (2002) classificam as técnicas de reconhecimento facial em imagens em quatro categorias:

- Método baseado em conhecimento: Esses algoritmos são utilizados basicamente para localização de faces e baseiam-se no conhecimento das características da face humana para estabelecer critérios para sua localização.
- Abordagens de características invariantes: Esses algoritmos têm como objetivo encontrar características estruturais, mesmo em condições de cor, iluminação e posições adversas. Também é muito utilizado para localização de faces.

- Métodos de modelos de harmonização: Utilizam modelos de descrição de faces. O processo de reconhecimento é realizado através de comparação das imagens de faces. É utilizada tanto para reconhecimento quanto para localização de faces.
- Métodos baseados em aparência: Ao contrário do modelo de harmonização, os modelos são aprendidos a partir de um conjunto de imagens de treinamento que deve capturar a variabilidade representante da aparência facial. Após aprender, o algoritmo é projetado principalmente para a identificação facial.

Figueiredo (2011) cita que a classe de métodos de reconhecimento de faces mais explorada é a de métodos baseados em aparência, pois se assemelha ao processo de reconhecimento feito pelos seres humanos.

“Um dos melhores métodos de reconhecimento facial é o eigenfaces”. (Gao e Leung, 2002).

“As características mais exploradas pelos detectores de bordas são a boca, o nariz, os olhos e as sobrancelhas”. (Yang, Kriegman e Ahuja, 2002).

Turk e Pentland (1991b) afirmam que embora o ser humano utilize acessórios que causem diferentes mudanças faciais como óculos e tapa-olho, e que ocorram diferentes expressões faciais causadas pelo tempo de vida, ou até mudanças de corte de cabelo e barba, o ser humano, ainda assim, consegue realizar o reconhecimento facial de forma eficaz.

Agarwal (2002) cita que o reconhecimento facial tornou-se uma questão importante em muitas aplicações, tais como sistemas de segurança, verificação do cartão de crédito, identificação criminal etc. Mesmo se houver a capacidade de detectar apenas rostos, ao invés de reconhecê-los, esse processo pode ser importante.

Agarwal (2002), embora seja notável que as pessoas são boas em reconhecimento de faces, não é um processo trivial como as faces são codificados ou decodificados por um cérebro humano. Reconhecimento de rosto humano tem sido estudado há mais de vinte anos.

Li (2011) propôs um sistema de reconhecimento facial o qual consiste basicamente das seguintes etapas: (a) aquisição das imagens; (b) pré-processamento; (c) extração de características; (d) classificação e verificação.

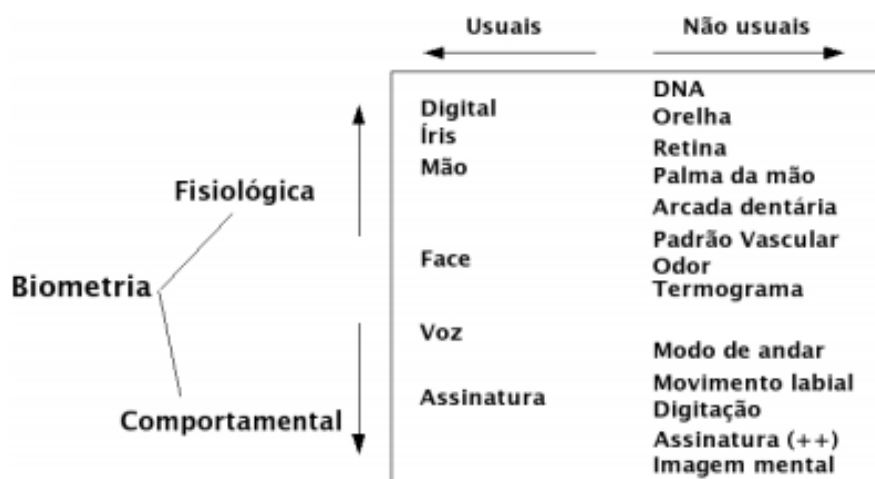


FIGURA 3 – Duas áreas da biometria. Fonte: Costa, Obelheiro e Fraga (2006).

Costa, Obelheiro e Fraga (2006) argumentam que ao longo do tempo, forma desenvolvidas várias tecnologias biométricas. Elas podem ser classificadas em dois grupos: fisiológicas (ou estáticas) e comportamentais (ou dinâmicas). Fisiológicas são traços que mudam pouco (ou nada) ao longo da vida do ser humano. Já as características comportamentais são aprendidas ou desenvolvidas ao longo da utilização constante, e podem variar fortemente ao longo do tempo. Tais divisões podem ser vistas na figura 3.

Grue (2004), discorre que a biometria está relacionada à grande quantidade de dados que precisam ser manipulados e analisados estaticamente, entretanto, por possuir representações simbólicas não numéricas, torna-se possível analisa-los através de um aplicativo de computador.

Andrade Neto (1997) acredita que um sistema de reconhecimento facial eficaz deve ser capaz de identificar uma face e suas características através de uma determinada imagem, do mesmo modo que o ser humano identifica uma pessoa simplesmente olhando para uma determinada fotografia, ou mesmo em uma transmissão de vídeo.

Alves (2008) descreve um esquema, conforme figura 4, onde é possível perceber os passos para o reconhecimento facial, sendo o primeiro passo a entrada da imagem teste, passando pela extração das características do indivíduo até chegar à imagem reconhecida.

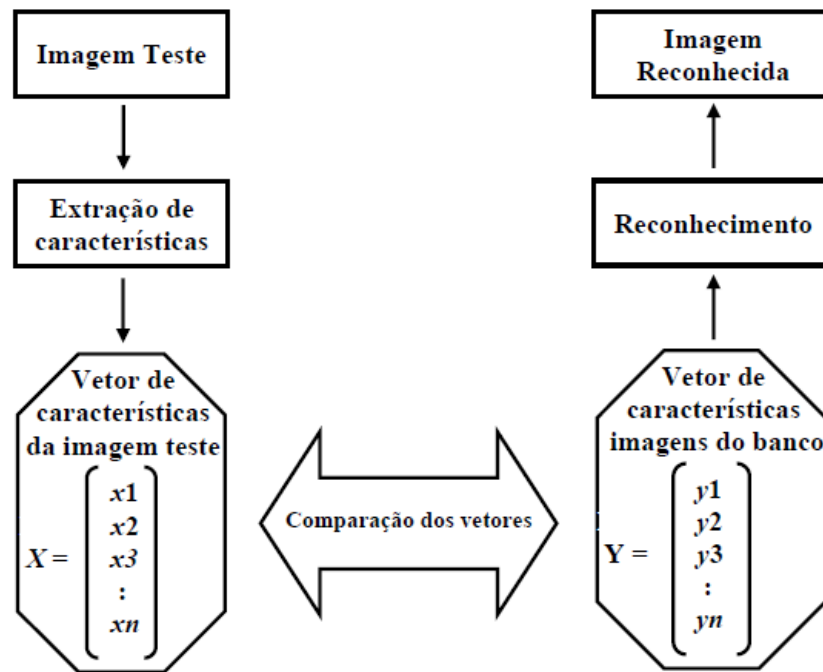


FIGURA 4 – Esquema básico de reconhecimento facial. Fonte: Alves (2008).

Segundo Da Silva e Santa Rosa (2004), o reconhecimento facial surgiu com o cientista inglês Sir Francis Galton, em 1878, com a apresentação de um artigo onde descreveu ser possível combinar e sobrepor imagens de faces e reproduzir as características típicas, eliminando as variações existentes. Galton imaginou uma série de aplicações úteis para a sua *Composite Portraiture* (Combinação de Fotos), conforme pode-se perceber na figura 5.

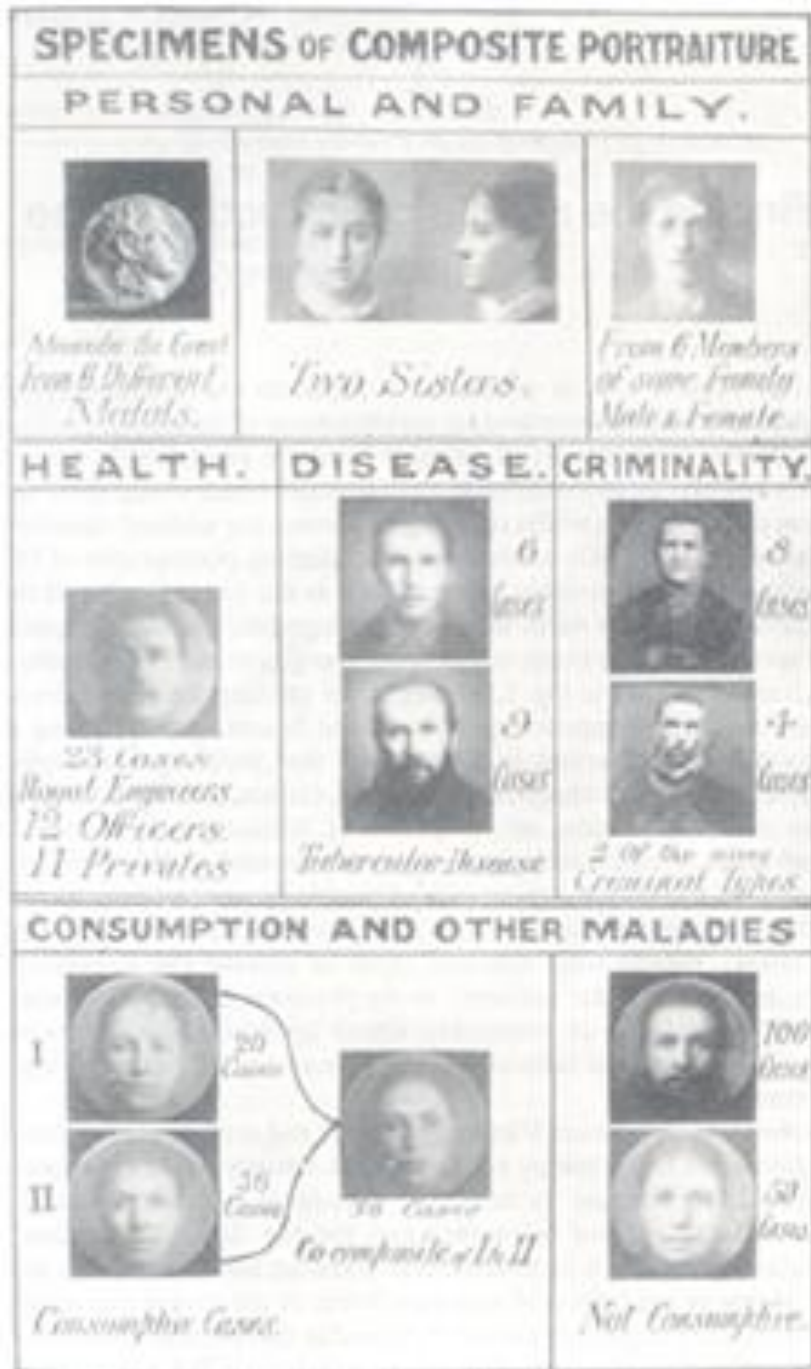


FIGURA 5 – Comparação de fotos de Galton. Fonte: Galton (apud Da Silva e Santa Rosa, 2004).

2.4 ESTUDO BIBLIOMÉTRICO

Esta pesquisa tem como objetivo mapear o estudo da arte no tema reconhecimento facial utilizando eigenfaces. Quanto à metodologia, foi feita uma análise das publicações na base de dados *Web Of Science* e também na base *Scopus* utilizando as palavras chave “*eigenfaces*” e “*Facial Recognition*”, selecionando materiais publicados até Julho de 2014.

O único idioma encontrado na confecção das publicações foi o inglês. Os resultados apresentados contribuem no sentido de direcionar os autores quanto às áreas mais carentes de trabalhos científicos aplicados a reconhecimento facial e eigenfaces, os países e idiomas com maior número de publicações, contribuindo de forma direcionada para o enriquecimento de trabalhos futuros.

A revisão bibliográfica foi feita tendo como base o método definido como *Webblomining*, desenvolvido por Costa (2010). O objetivo é pesquisar acerca de artigos que tratem do tema “reconhecimento facial” utilizando o método “*eigenfaces*”.

Como resultado da busca utilizando a frase “*eigenfaces*” e “*Facial Recognition*” (com aspas) foram retornados 36 registros na base *Web Of Science* e 27 na base *Scopus*, cuja distribuição, por tipo de documento está apresentada no quadro 01.

QUADRO 1: DISTRIBUIÇÃO DE REGISTROS POR VEÍCULO DE PUBLICAÇÃO

Tipo de Publicação	Quantidade de registros	
	<i>Web Of Science</i>	<i>Scopus</i>
Article	18	7
Proceedings paper	18	18
Review	-	1
Conference review	-	1
Total	36	27

FONTE: Elaboração própria (com dados da base de dados *Web of Science* e *Scopus*)

Este resultado indica que, até o momento da pesquisa, poucos são os materiais indexados nas bases que falem acerca do tema proposto e a maior parte são *proceedings paper*.

A partir deste resultado, foi aplicado um filtro como forma de selecionar apenas os registros associados a “*article*” e “*review*”. Esta ação levou aos resultados indicados no quadro 02, que demonstra os principais países onde as obras foram publicadas.

QUADRO 2: DISTRIBUIÇÃO DE REGISTROS POR PAÍSES

País	Quantidade de registros Das bases <i>Web Of Science</i> e <i>Scopus</i>
China	7
Estados Unidos	4
Canadá	2
Polônia	2
Coréia do Sul	2
Argentina	1
Inglaterra	1
Japão	1
Arábia Saudita	1
Singapura	1
África do Sul	1
Espanha	1
Turquia	1
Itália	1
Jordânia	1
Total	27

FONTE: Elaboração própria (com dados da base de dados *Web of Science* e *Scopus*)

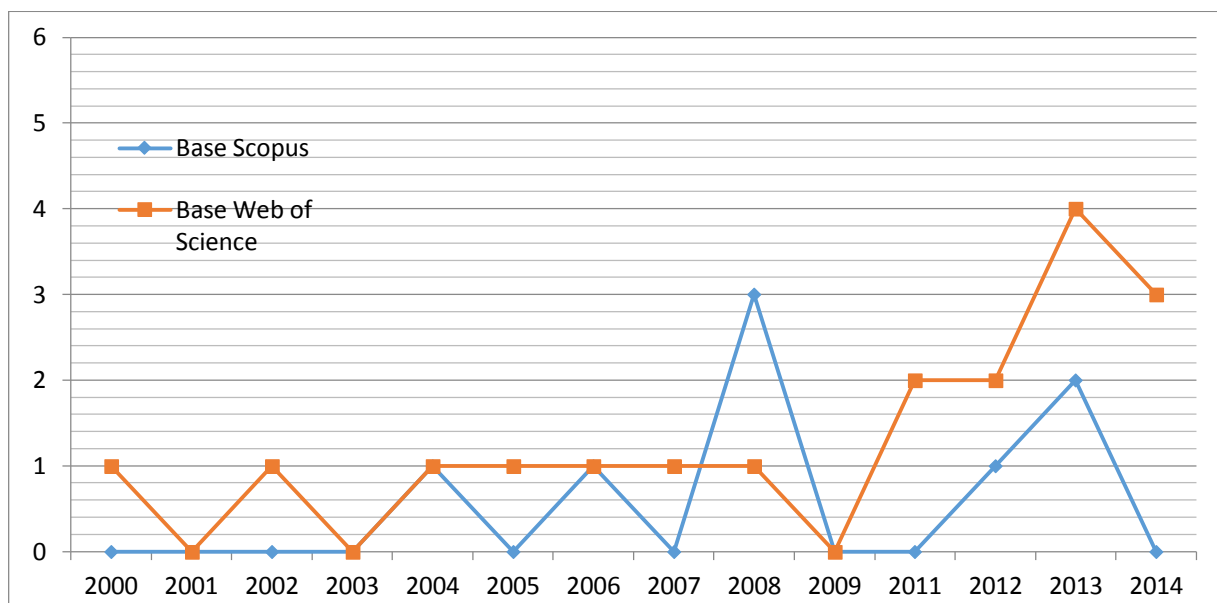
A China foi o país que mais publicou, com 7 publicações, seguida dos Estados Unidos. Canadá, Polônia e Coréia vêm logo atrás, com duas publicações cada. O restante dos países só teve 1 publicação cada. O destaque da China pode ser visto

como reflexo do crescimento na sua economia que também vem acendendo bastante nos últimos anos.

Apesar dos dados do quadro 1 serem importantes, inicialmente preferiu-se não restringir as publicações por países, até porque não se obteve dados suficientes que demonstrassem a soberania de um país em relação aos demais. Quanto ao idioma, todos os artigos foram escritos em inglês.

Em relação ao levantamento cronológico das produções, o gráfico 01 apresenta os dados referentes à distribuição de registros quanto ao ano de publicação.

GRÁFICO 1: DISTRIBUIÇÃO DAS PUBLICAÇÕES POR ANO



FONTE: Elaboração própria (com dados da base de dados ISI e no Scopus)

Através dos dados acima pode-se observar que a produção científica no tema pesquisado tem mantido a média de 01 publicação por ano ao longo do tempo, em uma escala cronológica. Analisando os dados referentes ao ano de publicação, observa-se que:

- A base *Web of Science* foram observados dois ciclos de produção mais acentuados: 2011-2012 e 2013;
- A base Scopus teve três ciclos: 2011 e 2013.
- Na base *Web of Science*, desde 2011 há uma crescente evolução no número de publicações, mesmo tendo uma queda em 2014 e 2010 que não houve nenhuma publicação na área;

- O artigo mais antigo indexado na base *Web of Science* refere-se ao ano de 2000, já na base Scopus é de 2004;

Após a análise cronológica, foi elaborada uma análise dos veículos de publicação com maior número de registros. Descobriu-se que existem dois veículos de publicação similares em ambas as bases analisadas. Também deu-se destaque para os periódicos que tinham 2 ou 3 publicações nas bases, haja vista que foi o maior número de registros. O quadro 03 apresenta os dados referentes à distribuição de registros quanto a esses títulos de periódicos. É importante frisar que foram desconsiderados 02 periódicos da base Scopus e 12 da base *Web of Science*, pois esses apresentaram somente 1 registro de publicação, não tinha similaridade entre as bases e também não estavam diretamente ligados ao tema em estudo.

QUADRO 3: DISTRIBUIÇÃO DE REGISTROS POR VEÍCULO DE PUBLICAÇÃO

Veículo de publicação	Quantidade de registros	
	<i>Web of Science</i>	Scopus
<i>Advances in Modelling and Analysis B</i>	-	3
<i>Neurocomputing</i>	2	-
<i>Journal Of Electronic Imaging</i>	2	-
<i>IEEE Latin America Transactions</i>	1	1
<i>SIAM Journal on Imaging Sciences</i>	1	1

FONTE: Elaboração própria (com dados da base de dados *Web of Science* e Scopus)

A partir dos dados observados foi feita a análise e separação dos artigos que poderiam ter informações relevantes acerca do assunto abordado nesta pesquisa. À partir de então, buscou-se analisar outras fontes, como no *Scielo* e no Google Acadêmico. Na primeira foi encontrado somente 1 artigo, enquanto que na segunda foram obtidos 3.490 registros, sendo que muitos registros apareceram repetidos. Dessa forma, houve um filtro pelos registros de maior relevância, mais novos, e com mais citações.

Não houve nenhum registro de uma aplicação web que foi feita utilizando o método *eigenfaces*. Sendo assim, este trabalho pretende avaliar e demonstrar o desempenho do método *eigenfaces* quando utilizado em uma aplicação web, podendo servir como base para futuras consultas.

3. MATERIAIS E MÉTODOS

O objetivo deste projeto é gerar um protótipo de sistema para reconhecimento facial utilizando o método *Eigenfaces*. Dessa forma, propõe-se o esquema demonstrado na figura 6 para a elaboração do sistema.

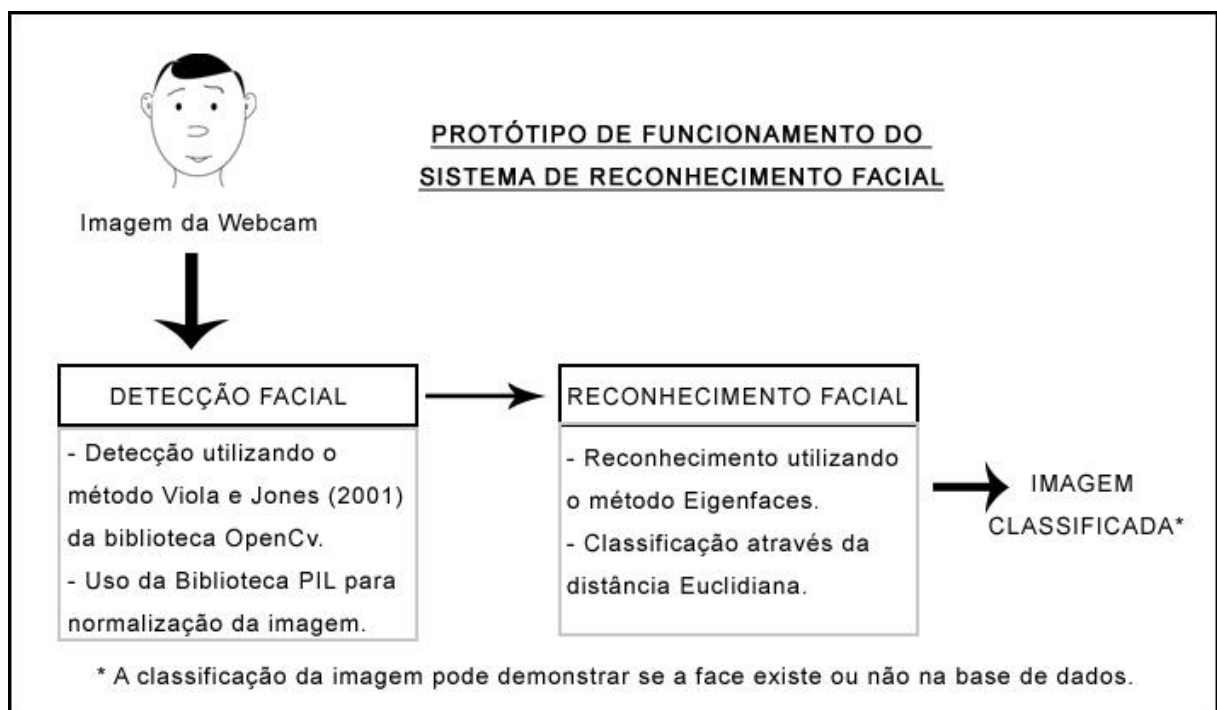


FIGURA 6 – Esquema de detecção e reconhecimento do protótipo do sistema . Fonte: O Autor.

A seguir, foram relacionadas algumas informações das tecnologias utilizadas no desenvolvimento deste ambiente.

Conforme a figura, uma imagem deve servir de entrada através da *webcam*, na qual o usuário necessita se posicionar na frente do periférico e está irá fazer a detecção facial do indivíduo utilizando o método proposto Viola e Jones (2001) que está presente na biblioteca OpenCv (*Open Computer Vision Library*). Assim, a imagem

é normalizada através da biblioteca PIL (*Python Image Library*). Assim, após a detecção da face, a mesma é analisada com base no método *eigenfaces* e classificada perante a base de face existente no programa.

3.1 AMBIENTE DE DESENVOLVIMENTO

A linguagem escolhida para o desenvolvimento deste projeto foi Python. Python é uma linguagem livre, que se caracteriza pela sintaxe simples, legibilidade do código e grande quantidade de módulos para os mais diversos fins. Além dela, foram utilizadas outras tecnologias:

Web2py Framework: é um framework livre e gratuito para desenvolvimento Web escrito em Python, que tem como um de seus principais objetivos proporcionar segurança e agilidade no desenvolvimento de aplicações web. Esse foi o framework que permitiu organizar a estrutura do projeto.

PIL: *Python Image Library* é uma biblioteca que suporta muitos formatos de imagens. Além disso, provê alto poder de processamento de imagens e gráficos. Permitiu fazer as edições nas imagens de entrada do processo de detecção facial.

OpenCV: A *Open Computer Vision Library* (OpenCv) é uma biblioteca multiplataforma que auxilia no desenvolvimento de aplicações na área de visão computacional (processamento de imagens e vídeos, detecção de objetos entre outros). Desenvolvido inicialmente pela empresa Intel®, tornou-se muito utilizado no meio acadêmico para experiências científicas, assim, atualmente possui licença BSD e pode também ser utilizado comercialmente.

NumPy: é um pacote para linguagem Python muito importante para cálculos científicos da computação. Foi utilizado para dar agilidade na identificação das *eigenfaces*.

Balsamiq Mockups: é um software proprietário capaz que modelar protótipos de interfaces de diversos projetos. Foi utilizada para a elaboração da interface web deste protótipo.

Em relação à configuração do computador utilizado neste projeto, os testes foram realizados em um notebook com processador Intel Core I7 4500U 1.8 GHz e 8 GB de RAM.

3.2 BANCO DE FOTOS (FACES)

O algoritmo apresentado por Turk e Pentland (1991a) necessita que todas as imagens da base estejam com as faces centralizadas e com o mesmo tamanho. Para tanto, as simulações para extração e representação das características de faces analisadas neste trabalho foram aplicadas no banco de imagens de domínio público bem conhecido na área de reconhecimento facial: ORL (1994). Entre as características dessa base, está a variação de iluminação e rotação das faces, o que torna a pesquisa mais próxima da realidade humana.

A base ORL contém imagens de quarenta pessoas, cada uma com dez imagens diferentes. As imagens foram preparadas em um processo rigoroso de controle pelo laboratório *AT&T Laboratories Cambridge* entre 1992 e 1994, controlando luminosidade e retirando o plano de fundo. As fotos possuem momentos diferentes, onde a mesma pessoa aparece com cabelo preso ou cabelo solto; de óculos ou sem óculos; boca aberta ou fechada; olho aberto ou fechado, com sorriso ou sério etc. As imagens estão também com pequenas rotações e inclinações.

O tamanho de cada imagem é 92 de largura por 112 de altura (92x112) em escala de cinza e o formato é PGM. Na Figura 7 pode-se perceber um exemplo de imagens utilizadas.



FIGURA 7 – Exemplos de imagens da base ORL. Fonte: ORL (1994).

Além da base ORL, necessitou de outra base para servir de dados para os testes à partir da webcam. Assim, foram escolhidas cinco pessoas de contato pessoal e profissional do autor e foram fotografadas 10 imagens contendo as mais variadas expressões faciais e inclinações desse grupo de pessoas, ilustrada na figura 8. A essa base se deu o nome de AUT (para representar imagens pertencentes ao autor).



FIGURA 8 – Exemplos de imagens da base AUT. Fonte: O autor.

Por fim, foram escolhidas aleatoriamente 35 pessoas da base ORL e agrupadas com as imagens da base AUT. Essas formaram a base TOT, que contém 40 pessoas com 10 fotos cada, totalizando 400 imagens de faces.

De forma resumida, a tabela 4 representa todas as imagens utilizadas nesta pesquisa.

	Base ORL	Base AUT	Base TOT
Total de imagens	400	50	400
Quantidade de Pessoas	40	5	40 (35 da base ORL e 5 da base AUT)
Imagens por pessoa		10	
Resolução		92x112	
Cor		Tons de cinza	

Tabela 4: Descrição das bases de faces utilizadas. Fonte: ORL (1994) e o Autor.

3.3 DETECÇÃO FACIAL

A detecção facial é uma importante etapa para o sucesso do sistema de reconhecimento facial. Seu objetivo principal é localizar uma face em meio a uma imagem. Existem diversos algoritmos que permitem efetuar a detecção facial, entre eles pode-se citar o de Viola e Jones (2001). Esse algoritmo é muito utilizado em pesquisas sobre faces e tem como característica principal encontrar em uma imagem características que codificam alguma informação do objeto (face, por exemplo) a ser detectada.

Segundo Araújo (2010), o algoritmo de Viola e Jones (2001) é capaz de detectar faces com precisão, alta taxa de acerto, baixa taxa de falsos positivos e baixo custo computacional.

Os padrões de detecção são baseados nas “características de *Haar*”, conforme pode-se perceber na figura 9, que codificam informações sobre a existência de contrastes orientados entre regiões da imagem e que possam representar o objeto a ser detectado.

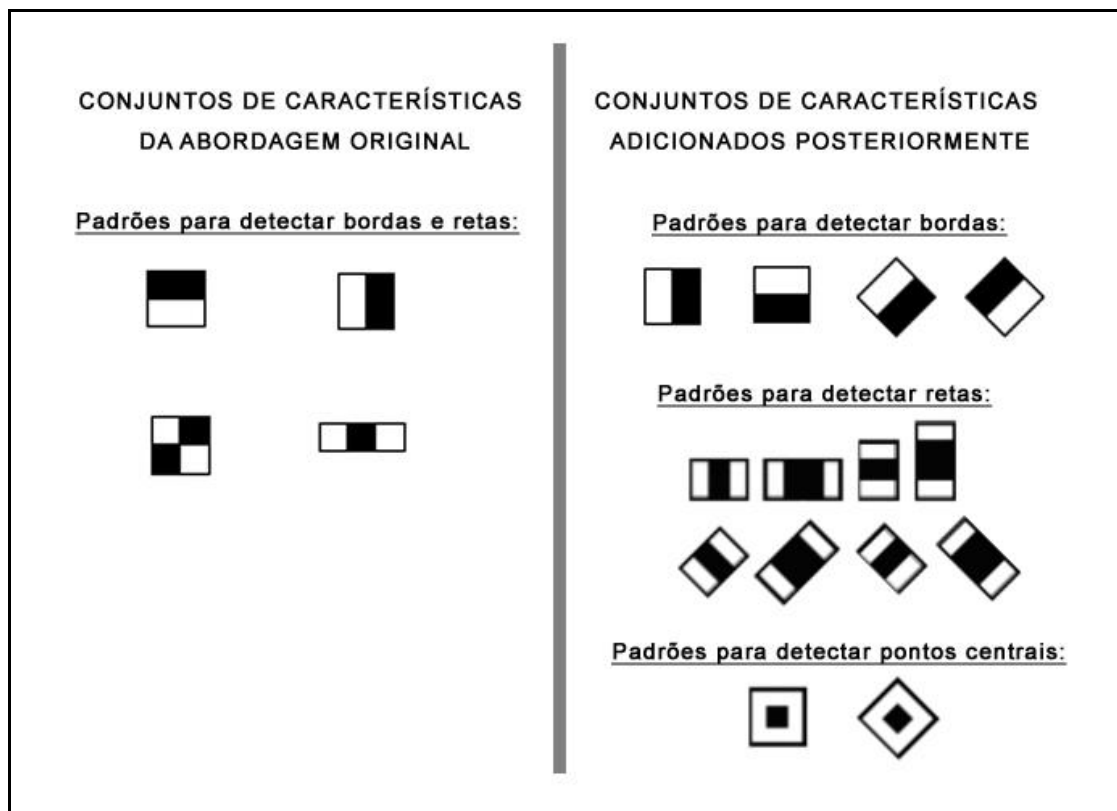


FIGURA 9 – Características de *Haar* mais usadas em detecção de faces. Fonte: VIOLA e JONES (2001)

A figura 10 mostra a utilização de algumas características de *haar* para detectar partes de uma face (olhos e nariz).

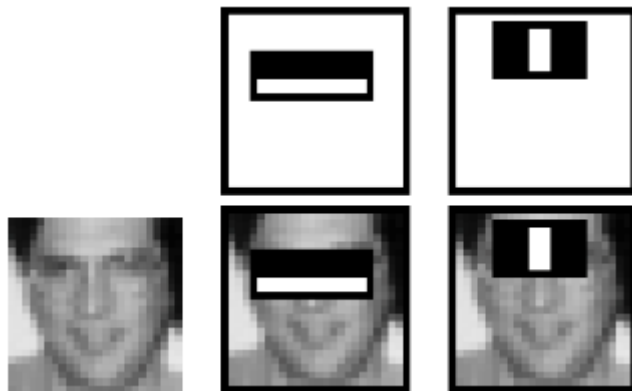


FIGURA 10 – Associação de padrões com partes de uma face. Fonte: VIOLA e JONES (2001)

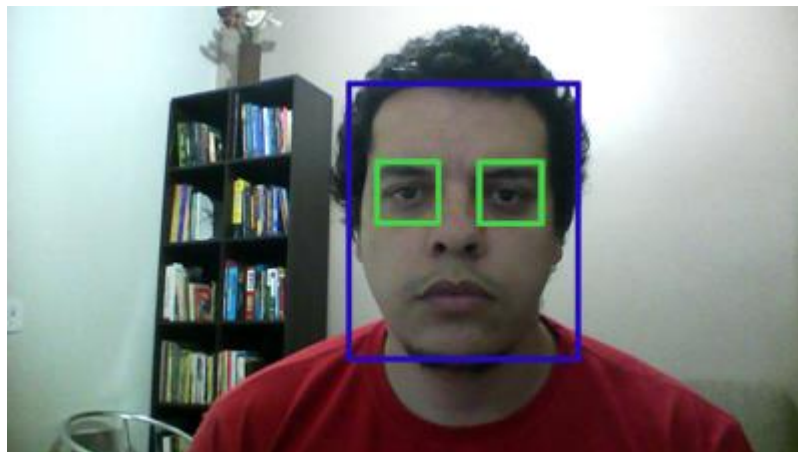


FIGURA 11 – Detecção de face, utilizando os olhos como padrão. Fonte: VIOLA e JONES (2001)

A figura 11 demonstra um exemplo do funcionamento das características de haar para detecção de uma face. A imagem seguiu o padrão do OpenCv, onde são colocados quadro detectando a face e os olhos. O exemplo foi promovido em tempo real, utilizando a própria webcam do notebook.

Posteriormente à fase de detecção, separa-se esta imagem (*crop*) para que possa ser utilizada no processo de reconhecimento facial. Não é uma etapa fácil haja vista que, normalmente, há vários elementos na foto que acabam dificultando o procedimento eficaz de detecção.

Optou-se por utilizar o Opencv que possui biblioteca para a linguagem Python. O Opencv utiliza o algoritmo de Viola e Jones (2001) através do método *Haar*, que como já abordado, analisa contrastes na imagem que podem representar um rosto.

3.4 RECONHECIMENTO FACIAL: *EIGENFACES*

O reconhecimento facial consiste em identificar uma pessoa. Porém, apesar de parecer fácil para os seres humanos, não é um processo trivial para o computador.

Esta é a etapa onde o rosto que serviu de entrada no sistema é processado e colocado em um “espaço de faces” para que seja possível compará-lo com o conjunto de imagens existentes na base. Para auxiliar esse processo, existem técnicas que possibilitam diminuir o tempo de processamento da comparação entre as imagens, analisando somente as características principais. Entre essas técnicas, pode-se citar o PCA, que será analisado à seguir. Existem também métodos que tornam o algoritmo mais eficiente, pois otimizam o cálculo dos autovetores e autovalores da imagem. Este projeto dará ênfase ao método *eigenfaces*.

Para este trabalho, optou-se pelo método de reconhecimento de faces denominado Eigenfaces.

Segundo Yang, Kriegman e Ahuja (2002), a técnica *eigenfaces* é baseada em aparência e busca os principais componentes de uma distribuição facial, ou seja, os auto-vetores (*eigenpictures*) de uma matriz de covariância de um conjunto de imagens. Foi desenvolvida por Turk e Pentland (1991a) com base nas técnicas de PCA.

Os *Eigenfaces* utilizam todos os passos da PCA que foram informados no item 2.1, no entanto há uma otimização para reduzir a matriz de covariância, e consequentemente o processamento para a geração de autovetores e autovalores serão mais rápidos.

Tendo como base a comparação de aparências, a seguir, serão considerados os passos para utilização do eigenfaces baseado em Turk e Pentland (1991a):

- a) Inicialização: Adquirir um conjunto de imagens de faces, treiná-lo e calcular o *eigenfaces*, definindo, assim, o espaço de faces.
- b) Assim, deve-se calcular um conjunto de pesos com base na imagem da face de entrada e os *M eigenfaces* gerados na etapa anterior, projetando a imagem de entrada em cada um dos *eigenfaces*.
- c) Determinar se a imagem de entrada é uma face. Para isso, deve-se verificar se a imagem está suficientemente próxima de outra no "espaço de faces."
- d) Caso seja uma face, deve-se classificá-la como imagem conhecida ou desconhecida da base de treinamento, utilizando para isso o seu peso.
- e) (Etapa opcional) Se uma mesma imagem desconhecida é analisada várias vezes, sugere-se que o sistema calcule as características de peso da imagem e a incorpore na base de faces do sistema. Isso faz com que o sistema aprenda a reconhecer as faces.

3.4.1 Como calcular o *eigenfaces*?

A ideia de Turk e Pentland (1991a) é demonstrar que uma simples imagem facial pode ser considerada matriz bidimensional de N por N ou um vetor de dimensão N^2 . Por exemplo, se tiver uma imagem de 256×256 poder-se-á descrever um vetor de 65.536 posições ou um ponto no espaço de 65.536-dimensional. Um conjunto de imagens então é mapeado neste grande espaço de pontos.

Se for considerado que as imagens de faces possuem as mesmas configurações (tamanho e faces centralizadas, por exemplo) elas não serão distribuídas de maneira aleatória nes grande espaço. Dessa forma, o objetivo é utilizar o PCA para reduzir o espaço de faces para uma estrutura menor, sem que sejam perdidas informações importantes. A esse espaço foi dado o nome "espaço de faces".

Abaixo foram relacionados os passos do método *eigenfaces* para treinamento do conjuntos de M imagens:

Passo 01: Considerando um conjunto de imagens (M) existentes na base de treinamento, deve-se treiná-las. Assim, o primeiro passo é armazená-las em um vetor coluna, conforme a figura 12. O vetor coluna que contém cada imagem recebeu

originalmente o nome de Γ_1 para a face 1, Γ_2 para a face 2 e assim até completar as M faces. Sendo assim, a transformação ocorrida na figura 12 deverá acontecer em todas as imagens do conjunto de faces.

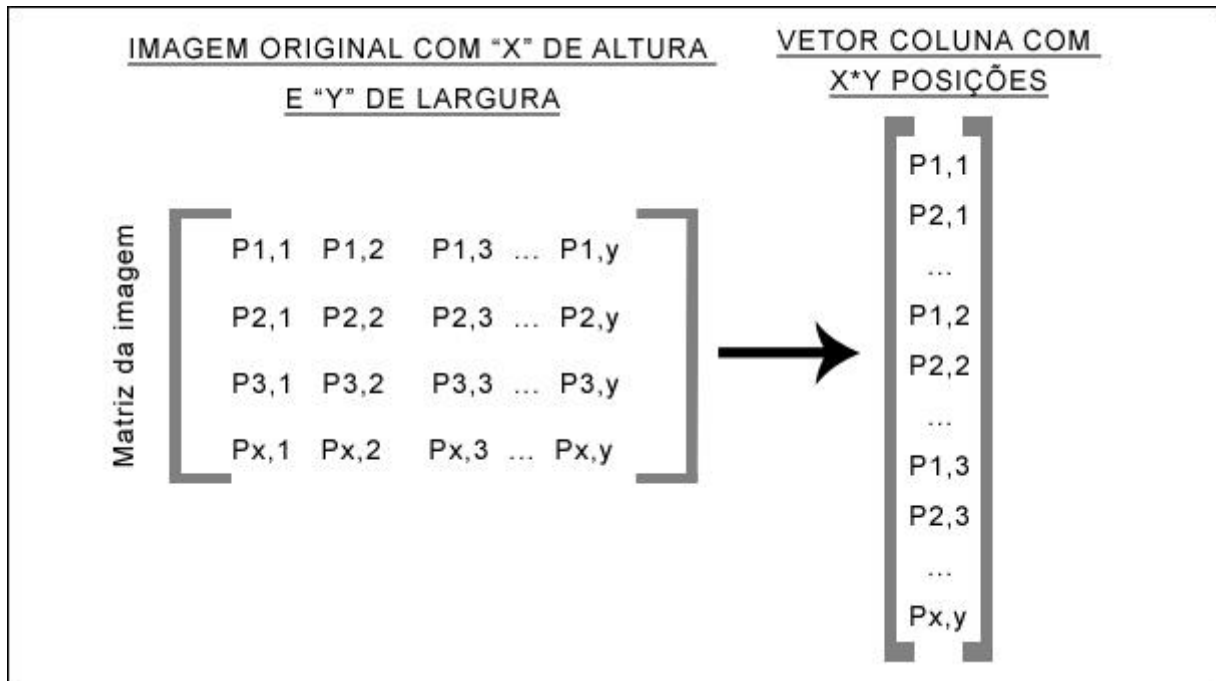


FIGURA 12 – Transformação de matriz de imagem em vetor coluna. Fonte: Elaboração própria com base em Turk e Pentland (1991a)

A partir deste ponto, sempre que se falar de N, estar-se-á referenciando à multiplicação de X por Y, ou seja, largura * altura da imagem.

Passo 02: Deve-se calcular o vetor coluna médio, conforme a fórmula (4).

$$\Psi = \frac{1}{M} \sum_{N=1}^M \Gamma_N \quad (4)$$

Onde:

- vetor coluna médio Ψ terá N linhas e 1 coluna.
- M significa a quantidade de imagens do conjunto M.

Passo 03: Subtrair o vetor coluna média de cada uma das M imagens de faces.

$$\phi_i = \Gamma_i - \Psi \quad (5)$$

Onde:

- cada vetor coluna Φ_i terá N linhas por 1 coluna.

Passo 04: Esse conjunto de vetores é então sujeito ao PCA, o qual busca pelos autovetores (*eigenvectors*) \mathbf{U}_n e seus autovalores (*eigenvalues*) λ_k associados que melhor descrevem a distribuição dos dados. Dessa forma, os \mathbf{U}_k são o *eigenvectors* e os λ_k são os seus respectivos *eigenvalues* da matriz de covariância.

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (6)$$

Onde:

- a matriz A é formada pelos vetores coluna Φ . Assim, $A = [\Phi_1, \Phi_2 \dots \Phi_m]$, e terá N linhas por M colunas.

A matriz C será gerada através da multiplicação de A por A^T , logo, terá será uma matriz com N linhas e N colunas. A partir dessa matriz de covariância serão descobertos N *eigenvectors* e N *eigenvalues*.

Porém, Turk e Pentland (1991b) afirmam que esse valor pode parecer alto, então deve-se verificar se o número de imagens é menor que o espaço de faces, ou seja, se $M < N$. Em caso afirmativo, é possível fazer uma combinação linear com as imagens de face de Φ_i .

$$A^T A V_i = \mu_i V_i \quad (7)$$

Multiplicando ambos os lados por A, tem-se:

$$A A^T A V_i = \mu_i A V_i \quad (8)$$

Dessa forma, sabendo-se que AV_i são os *eigenvectors* da matriz de covariância $C = AA^T$

Assim, seguindo essa análise, deve-se construir a matriz $L = \phi_m^T \phi_n$, encontrar os M *eigenvectors* V_i e determinar através de combinação linear as *eigenfaces* U_i :

$$U_l = \sum_{k=1}^M V_{lk} \Phi_k \quad (9)$$

Onde:

- l é igual ao conjunto $\{1, 2, 3, 4 \dots M\}$

Dessa forma, feito a análise acima, os cálculos serão bastante reduzidos. Basicamente, a matriz reduziu de N (número de pixels da imagem) para M (número de imagens). Em um exemplo prático, se tiver 100 imagens no tamanho de 100×80 , os cálculos serão reduzidos de 8.000 (N) para 100 (M).

Após essa etapa obter-se-á as M *eigenfaces* que calculadas através do treinamento das imagens de face da base. Agora é necessário iniciar a etapa de classificação da imagens de entrada.

Nesta próxima etapa, com as *eigenfaces* criadas, pode-se dizer que o processo é o reconhecimento de padrões. Os *eigenfaces* estão distribuídos em um subespaço M' -dimensional em relação ao espaço de faces original de N . M' são os *eigenvectors* (autovetores) da da matriz L , que foram escolhidos com base no maiores *eigenvalues* (autovalores). O número de *eigenfaces* utilizados é escolhido heurísticamente com base nos *eigenvalues*. A figura 13 demonstra algumas *eigenfaces* geradas à partir do modelo de Turk e Pentland (1991a).



FIGURA 13 – Exemplos de *eigenfaces*. Fonte: Elaboração própria (teoria de Turk e Pentland, 1991a)

Assim, abaixo foram relacionados os passos do método *eigenfaces* para classificação de uma imagem de entrada:

Passo 01: Primeiramente deve-se projetar cada uma das faces de Φ no espaço

de faces, através da fórmula:

$$\omega_k = U_k^T \phi_i \quad (10)$$

Onde:

- i é igual ao conjunto $\{1, 2, 3, 4 \dots M\}$, ou seja, representa cada imagem de treinamento.
- k são os valores de 1 até M' *eigenfaces*.

Após essa etapa, serão gerados M vetores $\Omega_i = \{ \omega_1, \omega_2, \omega_3 \dots \omega_{m'} \}$, onde $i = \{ 1, 2, 3 \dots 50 \}$ que corresponde ao conjunto de pesos de cada imagem projetada no espaço de faces.

Passo 02: Considerando a entrada de uma imagem de tamanho X por Y , deve-se armazená-las em um vetor coluna, conforme a figura 12. O vetor coluna que contém a imagem será denominado Γ_p . Sendo assim, ocorrerá a mesma transformação apresentada na figura 12. O vetor coluna Γ_p terá N linhas por 1 coluna.

Passo 03: Subtrair o vetor coluna média da imagem de entrada.

$$\phi_p = \Gamma_p - \Psi \quad (11)$$

Onde:

- o vetor coluna Φ_p terá N linhas por 1 coluna.

Passo 04: Deve-se projetar a imagem de entrada correspondente à face que se quer identificar. Assim, repete-se o passo 1, porém utilizando a imagem de entrada. Veja a fórmula:

$$\omega_k = U_k^T \phi_p \quad (12)$$

Onde:

- k são os valores de 1 até M' *eigenfaces*.

Após essa etapa, será gerado um vetor $\Omega_p = \{ \omega_1, \omega_2, \omega_3 \dots \omega_{m'} \}$.

Passo 05: Deve-se calcular a distância entre as os vetores de pesos. A menor distância e que esteja abaixo de um valor limiar (*threshold*) Θ prédefinido será considerado a face conhecida. Conforme visto anteriormente, no item 2.3, Perlibakas (2004) cita que a forma mais comum de calcular a distância entre vetores é através da distância Euclidiana, mas outras formas de medição também podem ser utilizadas, como a distância de *Manhattan*.

No formato original do método *eigenfaces*, Turk e Pentland (1991a) sugerem a distância Euclidiana, dessa forma o cálculo deve ser estabelecido da seguinte forma:

$$\epsilon_k = ||(\Omega_p - \Omega_k)|| \quad (13)$$

Onde:

- k são os valores de 1 até M vetores da base de faces.

A face é classificada (identificada) como a pessoa “ k ” que possuir a menor distância euclidiana ϵ_k e que esteja abaixo do valor limiar (*threshold*) Θ . Caso contrário, a face de entrada será classificada como desconhecida.

O *eigenfaces* ainda pode contribuir para verificar se uma imagem representa a face uma pessoa ou representa um outro objeto qualquer. Isso é possível porque quando se elabora o *eigenfaces*, forma-se o padrão que as faces devem possuir dentro das imagens. Dessa forma, se uma imagem foge desse padrão, ela não representa uma face.

Uma imagem de face não muda radicalmente quando projetada no espaço de faces, enquanto uma imagem que não possui uma face irá aparecer completamente diferente. Essa afirmação pode ser comprovada na figura 14.

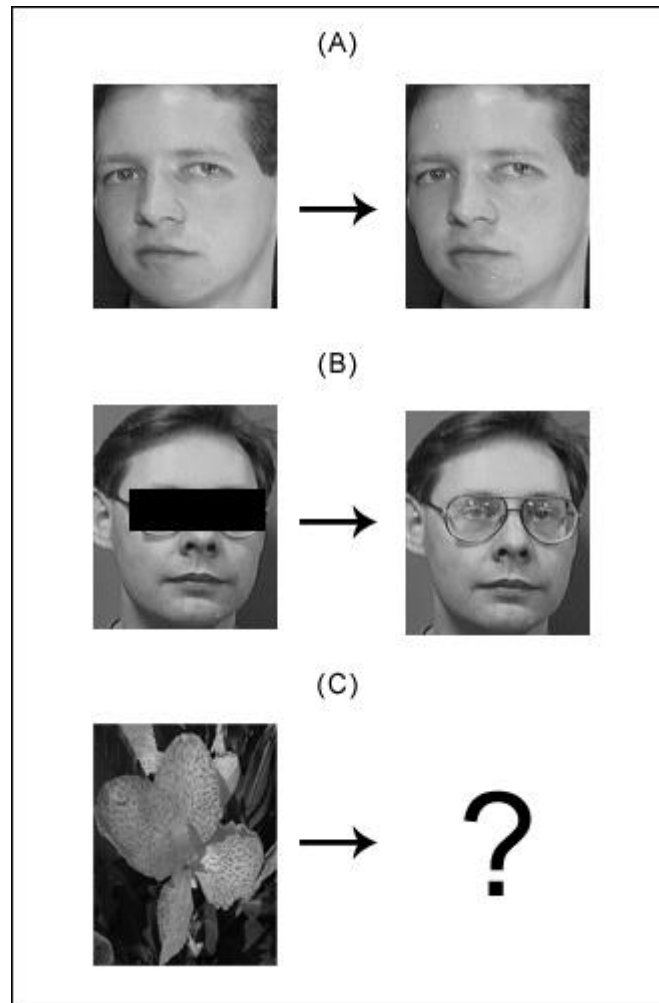


FIGURA 14 – Projeção de imagens com base nos *eigenfaces*. Fonte: Elaboração própria com base na teoria de Turk e Pentland (1991a)

Na figura 14, os autores apresentam um exemplo simples de como funciona a projeção de faces em um espaço de faces e como se apresentam sua classificação. Mesmo com a redução de características, a imagem pode ser reconstruída sem grandes alterações, exceto a imagem C que não representa uma face.

A figura 15 ilustra uma versão simplificada do espaço de faces com quatro projeções de imagens de faces. No exemplo, existem duas *eigenfaces* (u_1 e u_2) e três exemplos de indivíduos da base de treinamento (Ω_1 , Ω_2 e Ω_3).

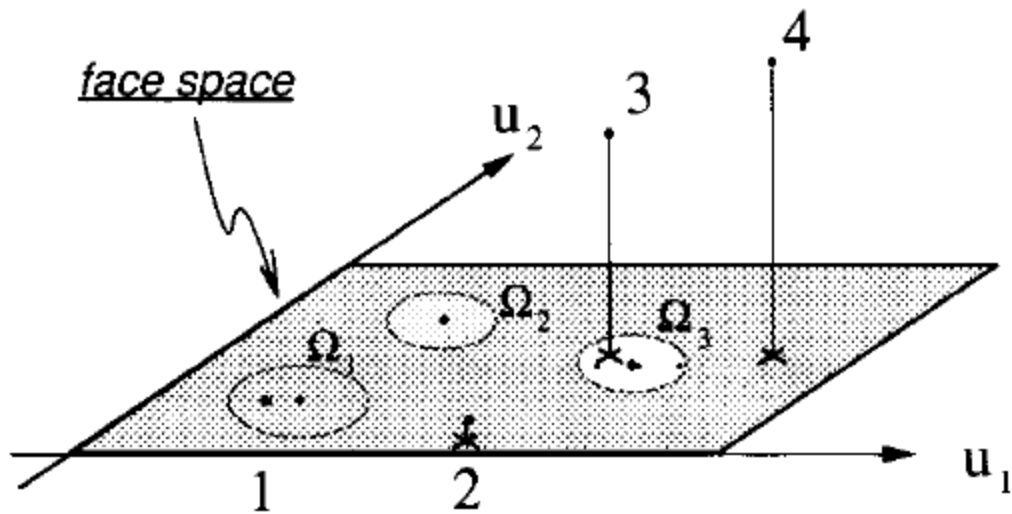


FIGURA 15 – Espaço de faces simplificado. Fonte: Turk e Pentland (1991a)

Nessa abordagem são apresentadas 4 possibilidades de situações para uma imagem de entrada:

- (1) Perto de um espaço de faces e próximo de uma classe de face. Aqui um indivíduo é reconhecido e identificado.
- (2) Perto de um espaço de face, porém distante de uma classe face. Um desconhecido indivíduo está presente.
- (3) Distante de um espaço de face e perto de uma classe de face. Não é uma imagem de face. Aqui percebe-se como um falso positivo pode ser reconhecido nos sistemas, porém, o método *eigenfaces* pode detectar esse falso reconhecimento devido a distância entre a imagem e o subespaço da imagem esperada.
- (4) Distante de um espaço de face e distante de uma classe de face. Não é uma imagem de face.

Com base nessa ideologia, na figura 14, pode-se perceber exemplos de situações geradas pelas quatro possibilidades acima. A e B são exemplos do caso (1), enquanto o C representa o caso 4.

Um ponto muito importante a ser considerado é que a *eigenfaces* necessita de poucas imagens de cada indivíduo para criação de uma base de treinamento.

3.5 CLASSIFICAÇÃO COM DISTÂNCIA EUCLIDIANA

Após efetuar os procedimentos para reconhecer uma face, chega-se ao seguinte ponto: existe um vetor coluna com os pixels da imagem a ser reconhecida e um conjunto de imagens testes que foram preparadas e colocadas em uma matriz de vetor coluna, conforme pode ser visto na figura 12.

Para realizar o cálculo da distância entre os vetores, optou-se pelo método da distância Euclidiana, haja vista que é um método muito utilizado no meio científico, conforme afirma Perlibakas (2004). Assim, sabendo-se que se A e B são dois vetores de tamanho D, a fórmula da distância Euclidiana pode ser definida assim:

$$d(A, B) = \sqrt{\sum_{i=1}^D (a_i - b_i)^2} = ||A - B|| \quad (14)$$

4. SISTEMA DE RECONHECIMENTO FACIAL PROPOSTO

4.1 DESCRIÇÃO GERAL

O modelo proposto é um sistema web que utiliza o *eigenfaces* para reconhecimento facial. Esse sistema consiste em uma área de autenticação na qual o usuário, previamente cadastrado, será detectado através da webcam.

Logo em seguida, após ter sua face detectada, ele será redirecionado automaticamente para a próxima tela, que fará o reconhecimento facial e mostrará a foto do usuário que está cadastrada na base de treinamento. Dessa forma, o mesmo deve digitar sua senha para ser redirecionado para a área restrita do sistema.

Caso o usuário não tenha sua face reconhecida pelo sistema, ele será redirecionado para uma tela solicitando que o mesmo repita o processo, pois sua face não foi reconhecida. A utilização da senha, neste exemplo foi somente evitar falsos-positivos no reconhecimento facial. A ilustração da interface desse módulo pode ser vista na figura 16.

O outro módulo do sistema será o cadastro de pessoas. Esse é um módulo administrativo que possibilita ao administrador o cadastro dos dados pessoais e das fotos que servirão de treinamento para a base.

A escolha para a utilização do método *eigenfaces* foi devido à sua eficiência e rápido poder de processamento. Assim, resolveu-se testá-lo em plataforma WEB para confirmar a viabilidade desse processo.

Outra vantagem a ser citada é que não é necessário criar o *eigenfaces* toda

vez que for efetuar o reconhecimento facial. A ideia é executar a fase treinamentos uma única vez e armazená-la para consultas futuras, e a partir daí só executar a parte de reconhecimento. Quando surgir o cadastro de uma nova pessoa, basta atualizar a base de treinamento. Dessa forma, pode-se dizer que o método eigenfaces também é um método de aprendizagem.

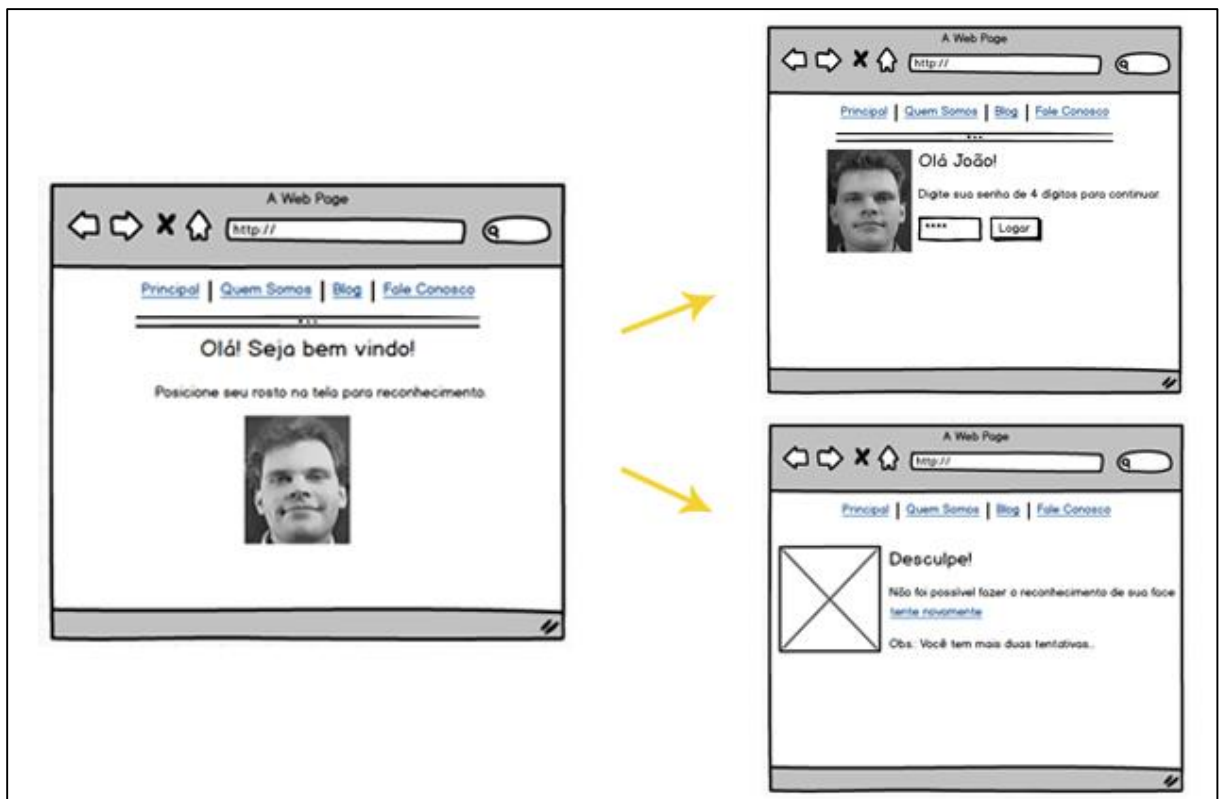


FIGURA 16 – Protótipo da interface do módulo de autenticação. Fonte: Elaboração própria através do software Balsamiq Mockups

4.2 MÓDULO CADASTRO DE PESSOAS

Área Administrativa -> Cadastro de Usuário!

...

Preencha todas as informações abaixo:

Nome:

Senha:

Nome:	Tamanho:	Remover
DSC00097.jpg	76kb	X
DSC00098.jpg	65kb	X
DSC00099.jpg	86kb	X

FIGURA 17 – Protótipo da interface do módulo de cadastro. Fonte: Elaboração própria através do software Balsamiq Mockups

O módulo de cadastro é a parte de alimentação do sistema. Nela será possível cadastrar todos os usuários que poderão, futuramente, se autenticar e acessar a área restrita. O protótipo da interface deste módulo pode ser visto na figura 17.

Neste módulo, as informações básicas que deverão ser cadastradas são:

Nome: Nome completo do usuário

Senha: Senha de 4 dígitos para evitar falsos positivos.

Fotos: 10 imagens nas mais diversas posições, inclinações e expressões.

O item foto é considerado o principal item do projeto, pois é este que estará alimentando a base de treinamento para a confecção dos eigenfaces e futura comparação para a análise do reconhecimento facial. As imagens serão inseridas

através de um campo de *upload*. Considera-se importante que o administrador já cadastre as fotos seguindo o padrão:

- Imagem em formato JPEG (ou JPG): *Joint Photographic Experts Group* é um padrão de imagem fotográfica.
- Imagem em tons de cinza, conhecido como *Gray Scale* ou escala de cinzas.
- A face da pessoa deve estar centralizada na imagem.
- O local onde a foto for produzida deve estar bem iluminado e sem sombras, para evitar que sejam produzidos ruídos na imagem.

Um exemplo dessas imagens pode ser visto na figura 18.



FIGURA 18 – Exemplo de faces da base AUT. Fonte: Elaboração própria

4.3 MÓDULO DE AUTENTICAÇÃO ATRAVÉS DE RECONHECIMENTO FACIAL

Tendo como base a comparação de aparências, a seguir, serão considerados os passos para utilização do *eigenfaces* baseado em Turk e Pentland (1991), já citados no item 3.4 desta pesquisa.

Para esta pesquisa, vários foram os testes efetuados e estes serão explicados no capítulo 5, em resultados e discussão.

Os dados utilizados para explicar esta pesquisa servem apenas para referência de aplicado do método *eigenfaces*:

- Foram escolhidas 5 pessoas com 10 imagens de faces de cada. Totalizando, assim, 50 imagens. Todas da base AUT.
- Cada imagem possui tamanho 92 de largura por 112 de altura.

4.3.1 Fase de Treinamento:

- A. Obter as imagens de treinamento e gerar um vetor coluna para cada imagem.

Nessa etapa é necessário pegar cada imagem de face da base de treinamento e concatená-la para que se tenha um vetor coluna para cada imagem. Tal concatenação pode ser observada na figura 19.

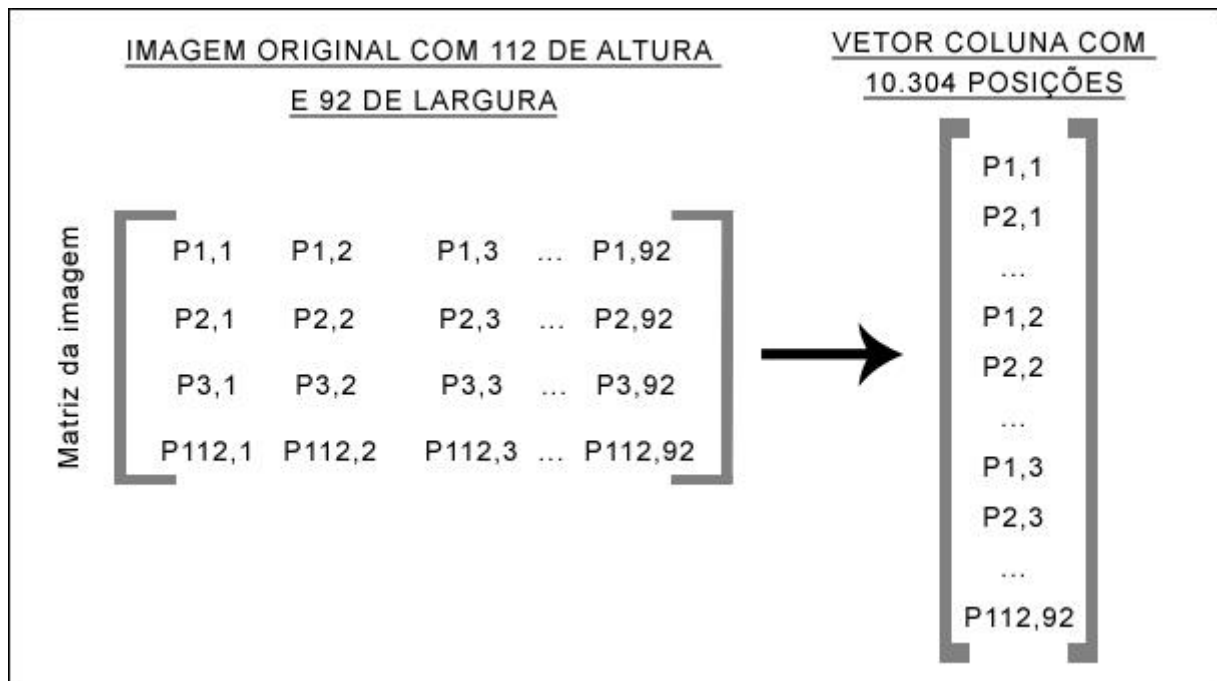


FIGURA 19 – Transformação de matriz de imagem em vetor coluna. Fonte: Elaboração própria com base em Turk e Pentland (1991a)

Considerando que se tenha 50 imagens de 92 x 112 (largura x altura), ter-se-á 50 vetores Γ de 10.304 linhas ($\Gamma_{i,[10.304]}$).

- B. Calcular o vetor coluna médio.

O vetor coluna médio será a soma de todas as colunas de cada linha e dividir pela quantidade de colunas (média aritmética). Assim, o vetor médio

Ψ terá tamanho 10.304×1 ($\Psi_{[10.304 \times 1]}$)

$$\Psi = \frac{1}{50} \sum_{N=1}^{50} \Gamma_N \quad (15)$$

Onde:

- vetor coluna médio Ψ terá 10.304 linhas e 1 coluna.
- N significa a quantidade de linhas de cada vetor Γ , ou seja, 10.304.



FIGURA 20 – Imagem Média gerada após o passo B. Fonte: Elaboração própria com base em Turk e Pentland (1991a)

C. Subtrair o vetor coluna média de cada uma das 50 imagens de faces.

$$\phi_i = \Gamma_i - \Psi \quad (16)$$

Onde:

- cada vetor coluna Φ_i terá 10.304 linhas por 1 coluna.
- I representa cada uma das 50 imagens da base de treinamento.

Logo em seguida, deve-se subtrair o vetor coluna médio de cada uma das colunas do vetor Γ gerados no passo A. Desta forma, serão gerados 50 novos vetores colunas Φ contendo cada um 10.304 linhas. Cada vetor coluna representa a respectiva imagem subtraída da média. Um exemplo da imagens geradas à partir do passo C pode ser visto na figura 21.



FIGURA 21 – Exemplo de imagens geradas após a subtração da média. Fonte: Elaboração própria com base em Turk e Pentland (1991a)

D. Calcular a matriz de covariância.

O cálculo da matriz de covariância deveria ser $C = A \cdot A^T$ ($10.304 \times 50 \times 50 \times 10.304$), porém é inviável computacionalmente gerar autovetores e autovalores com quantidade tão grande de dados, haja vista que seria gerada uma matriz C de tamanho 10.304×10.304 .

É nesse ponto que entra a técnica de Turk e Pentland, onde é feita uma inversão de cálculos da matriz de covariância, que será chamada de L:

$$L_{[50 \times 50]} = A_{[50 \times 10.304]}^T \cdot A_{[10.304 \times 50]} \quad (16)$$

Onde:

- a matriz A é formada pelos vetores coluna Φ . Assim, $A = [\Phi_1, \Phi_2 \dots \Phi_{50}]$, e terá N linhas por M colunas.

Agora a matriz de covariância terá tamanho 50×50 , o que possibilitará a geração de autovetores e autovalores de maneira mais rápida.

E. Calcular os *eigenfaces* com base nos autovetores e autovalores da matriz de covariância.

Tendo uma matriz L, os primeiros M' autovetores de C serão expressos à partir da combinação linear dos autovetores de L (chamados de V) e as imagens contidas em A (WEI, 1998, apud NOR'AINI, 2005).

Nesta etapa, os *eigenfaces* estão distribuídos em um subespaço M'-dimensional em relação ao espaço de faces original de 10.304. M' são os *eigenvectors* (autovetores) da da matriz L, que foram escolhidos com base no maiores *eigenvalues* (autovalores). O número de *eigenfaces* utilizados é escolhido heurísticamente com base nos *eigenvalues*. Dessa forma, foram selecionados $M' = M - 1$, ou seja, $M' = 49$ *eigenfaces*.

$$U_{[10.304 \times 49]} = A_{[10.304 \times 50]} \cdot V_{[50 \times 49]} \quad (17)$$

O símbolo U é o que chamamos de *eigenfaces*. É importante frizar

que, neste exemplo, serão gerados 49 eigenfaces. Na figura 22 pode-se observar alguns exemplos de eigenfaces, onde as características principais de cada uma das imagens do banco de faces são realçadas.



FIGURA 22 – Exemplos de eigenfaces. Fonte: Elaboração própria (teoria de Turk e Pentland, 1991)

4.3.2 Fase de Classificação

A. Projetar as imagens de treinamento no “espaço de faces”

A projeção de cada face Φ no espaço de faces é fundamental para o processo de reconhecimento de faces. Os valores obtidos também são conhecidos como “peso”. Esta projeção permite verificar quanto próximas são duas faces. A fórmula está representada abaixo:

$$\omega_k = U_k^T \phi i \quad (18)$$

Onde:

- i é igual ao conjunto $\{1, 2, 3, 4 \dots 50\}$, ou seja, representa cada imagem de treinamento.
- k são os valores de 1 até 49 (M') *eigenfaces*.

Após essa etapa, serão gerados 50 (M) vetores $\Omega_i = \{ \omega_1, \omega_2, \omega_3 \dots \omega_{49} \}$, onde $i = \{ 1, 2, 3 \dots 50 \}$ que corresponde ao conjunto de pesos de cada imagem projetada no espaço de faces.

Cada vetor coluna Ω representa a projeção de uma imagem. Ela servirá de base para o reconhecimento facial. Outro ponto importante a ser observado é que o tamanho (49×1) é bem menor do que o tamanho original da imagens (10304×1). Esse é o objetivo principal do *eigenfaces*.

- B. Obter a imagem de entrada e gerar um vetor coluna para esta imagem.

Conforme já citado, a imagem prova deve ter o mesmo tamanho das imagens de treinamento. Assim, caso seja necessário efetuar a normalização da imagem prova, esse é o momento de a fazer. Desta forma, a imagem prova será colocada em um vetor coluna denominado de Γ de dimensão 10.304×1 , conforme feito anteriormente.

- C. Subtrair o vetor coluna média da imagem de entrada.

$$\phi_p = \Gamma_p - \Psi \quad (19)$$

Onde:

- vetor coluna Φ_p terá 10.304 linhas por 1 coluna.

- D. Projetar a imagem prova, correspondente à face que se quer identificar, no “espaço de faces”.

A projeção da imagem no espaço de faces servirá para comparar com as projeções feitas com as imagens de treinamento. O valor obtido também é conhecido como “peso”. A fórmula está representada abaixo:

$$\omega_k = U_k^T \phi_p \quad (20)$$

Onde:

- k são os valores de 1 até 49 (M') *eigenfaces*.

Após essa etapa, será gerado um vetor $\Omega_p = \{ \omega_1, \omega_2, \omega_3 \dots \omega_{49} \}$.

- E. Comparar a distância entre a projeção da imagem prova com as imagens de treinamento

O próximo passo é comparar a distância entre o peso da imagem prova com cada um dos pesos das *eigenfaces*. Para o cálculo da distância entre as projeções, existem várias fórmulas (distância euclidiana, distância de manhattan etc.). Neste trabalho considerou-se a distância euclidiana, haja vista que foi a mais utilizada nas pesquisas avaliadas e trouxe resultados

satisfatórios. A fórmula da distância euclidiana encontra-se abaixo:

$$\epsilon_k = ||(\Omega_p - \Omega_k)|| \quad (21)$$

Onde:

- k são os valores de 1 até 50 (M) vetores da base de faces.

F. Calcular a distância entre a projeção da imagem prova e a imagem prova reconstruída

Também é importante calcular a distância entre a projeção da imagem prova com a imagem prova reconstruída. Assim, calcula-se as duas fórmulas abaixo. A primeira reconstrói a imagem e a segunda calcula a distância entre a imagem prova original e sua reconstrução:

$$\Gamma_r = (U * \Omega_p) + \Psi \quad (22)$$

$$\epsilon_p = ||\Gamma_p - \Gamma_r|| \quad (23)$$

A distância ϵ_p será importante para verificar se a imagem de entrada representa uma face ou não.

G. Calcular o valor limiar (*threshold*)

O valor *threshold* é muito importante para servir de limiar entre um face ser reconhecida ou não reconhecida. Se a face não está na base de treinamento, ela deve informar tal situação. Apesar de importante, não é tão fácil determinar uma fórmula única para descobrir o *threshold*. Gupta et. al (2010) observou que uma maneira eficaz é calcular 80% do maior valor entre as menores distâncias de cada uma das imagens da base treinamento em relação às outras imagens.

Assim, nesta pesquisa, calculaou-se a menor distância de cada uma das imagens da base treinamento em relação às outras imagens e armazenou-se em um vetor denominado “*rate*”. Em seguida, o *threshold* foi calculado como sendo 80% do maior valor armazenado no vetor *rate*. Esse valor foi

armazenado em Θ .

$$\Theta = 0.8 * \max(rate) \quad (24)$$

H. Definir como face reconhecida, não reconhecida ou não sendo uma face. Perlibakas (2004) alude que após calcular o *threshold*, deve-se efetuar os testes finais para a resolução da fase de reconhecimento e sugere o algoritmo:

Se ($\varepsilon_p \geq \Theta$) então

Imprima (“A imagem não representa uma face humana”)

Senão

Se ($\min(\varepsilon_i) \geq \Theta$) então

Imprima (“A imagem representa uma face humana”)

Imprima (“Porém, a face não foi reconhecida na base de dados”)

Senão

Imprima (“A imagem representa uma face humana”)

Imprima (“A face foi identificada no índice i do vetor R”)

fimse

fimse

5. RESULTADOS E DISCUSSÃO

Na etapa de testes desta proposta de sistema, tentou-se efetuar simulações que pudessem comprovar a possibilidade de utilização deste tipo de sistema e método para reconhecimento facial.

Para que os resultados da fase de testes não fossem comprometidos, as imagens de testes não eram as mesmas imagens de treinamento. Dessa forma, sabe-se que foram utilizadas 400 imagens ao todo, sendo 10 por pessoa. Assim, as imagens que não estão na etapa de treinamento, foram direcionadas para a etapa de testes. Na tabela 5 estão apresentados os resultados dos testes efetuados.

Tabela 5 – Resultados da simulação da aplicação do método eigenfaces na base TOT.

FASE DE TREINAMENTO		FASE DE AVALIAÇÃO		RESULTADOS		
Imagens por pessoa	Total de imagens	Imagens por pessoa	Total de imagens	Acertos	Erros	% de Acertos
1	40	9	360	201	159	55,83333333
1	40	8	320	179	141	55,9375
1	40	7	280	158	122	56,42857143
1	40	6	240	140	100	58,33333333
1	40	5	200	121	79	60,5
1	40	4	160	97	63	60,625
1	40	3	120	78	42	65
1	40	2	80	57	23	71,25
1	40	1	40	28	12	70
2	80	8	320	222	98	69,375
2	80	7	280	195	85	69,64285714
2	80	6	240	169	71	70,41666667
2	80	5	200	144	56	72
2	80	4	160	115	45	71,875
2	80	3	120	90	30	75
2	80	2	80	62	18	77,5

2	80	1	40	34	6	85
3	120	7	280	208	72	74,28571429
3	120	6	240	178	62	74,16666667
3	120	5	200	147	53	73,5
3	120	4	160	120	40	75
3	120	3	120	90	30	75
3	120	2	80	62	18	77,5
3	120	1	40	32	8	80
4	160	6	240	186	54	77,5
4	160	5	200	155	45	77,5
4	160	4	160	126	34	78,75
4	160	3	120	95	25	79,16666667
4	160	2	80	63	17	78,75
4	160	1	40	36	4	90
5	200	5	200	158	42	79
5	200	4	160	127	33	79,375
5	200	3	120	96	24	80
5	200	2	80	64	16	80
5	200	1	40	32	8	80
6	240	4	160	145	15	90,625
6	240	3	120	108	12	90
6	240	2	80	73	7	91,25
6	240	1	40	38	2	95
7	280	3	120	111	9	92,5
7	280	2	80	74	6	92,5
7	280	1	40	37	3	92,5
8	320	2	80	75	5	93,75
8	320	1	40	37	3	92,5
9	360	1	40	38	2	95

Fonte: Elaboração própria a partir do método eigenfaces (Turk e Pentland, 1991).

Conforme visto na Tabela 5, a porcentagem de acerto foi maior à partir do momento que aumentou-se o número de imagens (por pessoa) da base de treinamento, sendo que com 6 imagens por pessoa obteve-se no mínimo 90,62% de acerto, o que foi considerado como uma quantidade de imagens satisfatórias para o estudo em questão.

Se houver uma separação, analisando somente testes contendo 4 imagens pode-se observar que o aumento na porcentagem da taxa de acertos é gradativo, sendo considerável quando utilizado 6 imagens para treinamento, atingindo 90,62% de acertos. Com 5 imagens de treinamento essa taxa foi de 79,37% . Essa análise pode ser observada na tabela 6. Esses mesmos dados são demonstrados,

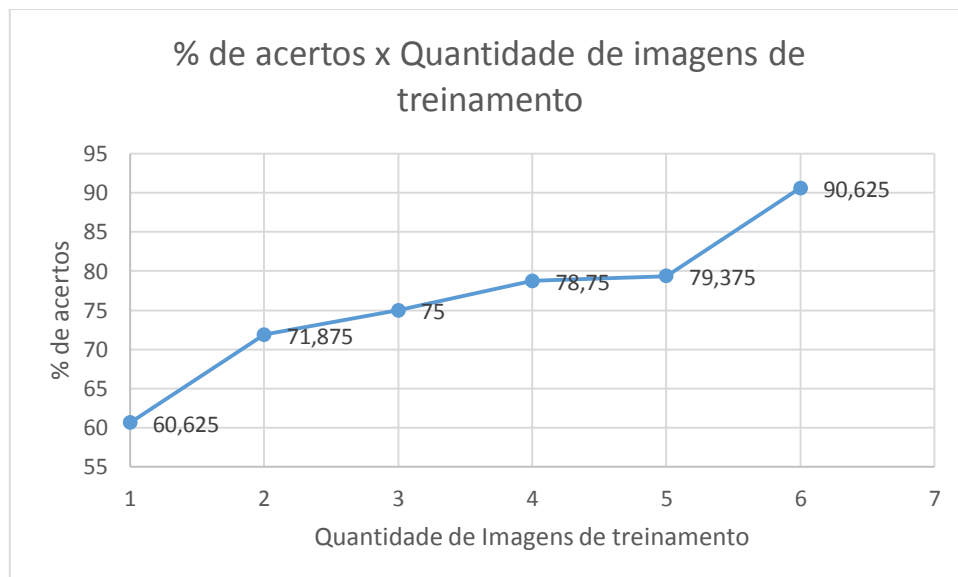
graficamente, no gráfico 2.

Tabela 6 – Resultados da simulação com base em 4 testes e de 1 a 6 imagens de treinamentos.

FASE DE TREINAMENTO		FASE DE AVALIAÇÃO		RESULTADOS		
Imagens por pessoa	Total de imagens	Imagens por pessoa	Total de imagens	Acertos	Erros	% de Acertos
1	40	4	160	97	63	60,625
2	80	4	160	115	45	71,875
3	120	4	160	120	40	75
4	160	4	160	126	34	78,75
5	200	4	160	127	33	79,375
6	240	4	160	145	15	90,625

Fonte: Elaboração própria a partir do método eigenfaces (Turk e Pentland, 1991).

GRÁFICO 2 - SIMULAÇÃO COM BASE EM 4 TESTES E DE 1 A 6 IMAGENS DE TREINAMENTOS



Fonte: Elaboração própria a partir do método eigenfaces (Turk e Pentland, 1991).

Na Tabela 7 foi feito uma ordenação com base nos melhores resultados, considerando para tal, aqueles que conseguiram melhor porcentagem de acertos.

Tabela 7 – Resultados da simulação da aplicação do método eigenfaces na base TOT, ordenados pelos melhores resultados.

FASE DE TREINAMENTO		FASE DE AVALIAÇÃO		RESULTADOS		
Imagens por pessoa	Total de imagens	Imagens por pessoa	Total de imagens	Acertos	Erros	% de Acertos
6	240	1	40	38	2	95
9	360	1	40	38	2	95
8	320	2	80	75	5	93,75

7	280	3	120	111	9	92,5
7	280	2	80	74	6	92,5
7	280	1	40	37	3	92,5
8	320	1	40	37	3	92,5
6	240	2	80	73	7	91,25
6	240	4	160	145	15	90,625
4	160	1	40	36	4	90
6	240	3	120	108	12	90
2	80	1	40	34	6	85
3	120	1	40	32	8	80
5	200	3	120	96	24	80
5	200	2	80	64	16	80
5	200	1	40	32	8	80
5	200	4	160	127	33	79,375
4	160	3	120	95	25	79,16666667
5	200	5	200	158	42	79
4	160	4	160	126	34	78,75
4	160	2	80	63	17	78,75
2	80	2	80	62	18	77,5
3	120	2	80	62	18	77,5
4	160	6	240	186	54	77,5
4	160	5	200	155	45	77,5
2	80	3	120	90	30	75
3	120	4	160	120	40	75
3	120	3	120	90	30	75
3	120	7	280	208	72	74,28571429
3	120	6	240	178	62	74,16666667
3	120	5	200	147	53	73,5
2	80	5	200	144	56	72
2	80	4	160	115	45	71,875
1	40	2	80	57	23	71,25
2	80	6	240	169	71	70,41666667
1	40	1	40	28	12	70
2	80	7	280	195	85	69,64285714
2	80	8	320	222	98	69,375
1	40	3	120	78	42	65
1	40	4	160	97	63	60,625
1	40	5	200	121	79	60,5
1	40	6	240	140	100	58,33333333
1	40	7	280	158	122	56,42857143
1	40	8	320	179	141	55,9375
1	40	9	360	201	159	55,83333333

Fonte: Elaboração própria a partir do método eigenfaces (Turk e Pentland, 1991).

Na Tabela 8, foram separadas as imagens da base AUT. Nesta simulação foi utilizada a webcam do próprio computador. Mesmo sem ter tido um tratamento laboratorial específico, o tratamento e normalização aplicados às imagens foi suficiente para atender ao objetivo proposto. Outrossim, todos os testes foram feitos em ambiente com boa iluminação.

Uma análise em comum entre as bases ORL e AUT é que 2 ou 3 imagens por pessoa, apesar dos resultados serem considerados bons, não trouxeram números tão satisfatórios para utilização em sistema real, ficando próximos de 60% e 80% da taxa de acerto, respectivamente.

Tabela 8 – Resultados da simulação da aplicação do método eigenfaces na base AUT

Testes	Treinamento e Testes		Resultados		
	Quantidade de imagens de treinamento	Quantidade de tentativas	Acertos	Erros	Taxa de acerto
1°	10 (2 por pessoa)	10 (2 por pessoa)	6	4	60%
2°	15 (3 por pessoa)	20 (4 por pessoa)	16	4	80%
3°	20 (4 por pessoa)	30 (6 por pessoa)	27	3	90%
4°	25 (5 por pessoa)	40 (8 por pessoa)	37	3	92,50%

Fonte: Elaboração própria a partir do método eigenfaces (Turk e Pentland, 1991).

6. CONSIDERAÇÕES FINAIS

A proposta desta dissertação é apresentar o método eigenfaces como técnica para o reconhecimento facial, especificamente em um sistema web. À partir dos dados obtidos com a simulação, foi possível perceber que é possível utilizá-lo para este fim.

Durante o desenvolvimento do trabalho, constatou-se a que o método estudado possui algumas exigências para seu funcionamento perfeito, tais como ambiente com boa iluminação e face da pessoa estar centralizada na imagem. Entretanto, mesmo com estas regras, o algoritmo de Turk e Pentland alcançou bons resultados no reconhecimento de faces, tanto com imagens salvas em arquivos quanto em imagens capturadas através da webcam.

Outro ponto positivo observado deste método foi que mesmo quando se faz pequenas rotações e inclinações com a face ou variação nas expressões faciais e oclusões de características, os resultados foram satisfatórios. Conclui-se também que não é necessário um grande número de fotos de face por pessoa na base de treinamento, haja vista que 6 imagens já foram suficientes para obter resultados acima de 90% de eficiência.

Propondo-se a utilização do método eigenfaces para reconhecimento facial em sistemas web pretende-se contribuir para futuras pesquisas onde se pretenda efetuar a identificação (autenticação) de pessoas para acesso a informações confidenciais.

REFERÊNCIAS

ABDELWAHAB, Moataz M.; ALY, Salah A.; YOUSRY, Islam. Efficient Web-based Facial Recognition System Employing 2DHOG. **arXiv preprint arXiv:1202.2449**, 2012.

AGARWAL, Mayank et al. Face recognition using principle component analysis, eigenface and neural network. In: **Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on**. IEEE, 2010. p. 310-314.

ALVES, Walison Joel Barberá. **Identificação de Pessoas através de Algoritmo Genético aplicado em medidas das Proporções Áureas da Face Humana**. 2008. Tese de Doutorado. Universidade de São Paulo.

ANDRADE NETO, E. L. **Sistemas de Identificação Pessoal Utilizando Técnicas de Reconhecimento e Verificação Facial Automáticas**. 1997. Dissertação de Mestrado em Engenharia Elétrica. Universidade de Campinas. São Paulo, 1997. Disponível em: <http://libdigi.unicamp.br/document>.

ARAUJO, Gabriel Matos. **Algoritmo para reconhecimento de características faciais baseado em filtros de correlação**. 2010. Tese de Doutorado. Universidade Federal do Rio de Janeiro.

BICALHO, Gustavo Carneiro. Desenvolvimento do módulo de reconhecimento facial para a ferramenta em análise forense FI2. 2013.

CAMPOS, T. E. Técnicas de Seleção de Atributos e de Classificação para Reconhecimento de Faces. Dissertação de Mestrado, Departamento de Ciência da Computação, IME - USP. 2000. Disponível em: <<http://www.vision.ime.usp.br/~teo/publications/qualificacao/quali.html>>

CORCORAN, P. M.; IANCU, C. Automatic Face Recognition System for Hidden Markov Model Techniques, New Approaches to Characterization and Recognition of Faces, 2011. ISBN: 978-953-307-515-0, InTech, DOI: 10.5772/17694. Disponível em <<http://www.intechopen.com/books/new-approaches-to-characterization-and-recognition-of-faces/automatic-face-recognition-system-for-hidden-markov-model-techniques>> Acesso em Mar. 2014.

COSTA, Luciano R.; OBELHEIRO, Rafael R.; FRAGA, Joni S. Introdução à Biometria. **Livro texto dos Minicursos do VI Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg2006)**. SBC: Porto Alegre, v. 1, p. 103-151, 2006.

COSTA, Helder Gomes. Modelo para webibliomining: proposta e caso de aplicação. **Revista da FAE**, v. 13, n. 1, p. 115-126, 2010.

EKMAN, Paul. Facial expression and emotion. **American psychologist**, v. 48, n. 4, p. 384, 1993.

FIGUEREDO, M. B. **Reconhecimento de faces aplicado ao problema de pessoas desaparecidas- Estudo de caso do Eigenface**. 2011. Dissertação (Mestrado)-SENAI CIMATEC, Salvador.

GAO, Yongsheng; LEUNG, Maylor KH. Face recognition using line edge map. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. 24, n. 6, p. 764-779, 2002.

GONZALEZ, Rafael C.; WOODS, Richard E. **Processamento de imagens digitais**. Edgard Blucher, 2000.

GUPTA, Sheifali et al. A new optimized approach to face recognition using eigenfaces. **Global Journal of Computer Science and Technology**, v. 10, n. 1, 2010.

GRUE A. R. *Facial Recognition: Limited Application in Safety and Security*, 2004. Disponível em <http://ocw.mit.edu/courses/science-technology-and-society/sts-035-the-history-of-computing-spring-2004/assignments/anthony_final.pdf>

JOLLIFFE, Ian. **Principal component analysis**. John Wiley & Sons, Ltd, 2002.

KIM, Kyungnam. Face recognition using principle component analysis. In: **International Conference on Computer Vision and Pattern Recognition**. 1996. p. 586-591.

KIM, D.; LEE, S.; SOHN M. *Face Recognition via Local Directional Pattern*. *International Journal of Security and Its Applications*. Vol. 7, No. 2, March, 2013.

KIRBY, M; SIROVICH, L. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990.

KSHIRSAGAR, V. P.; BAVISKAR, M. R.; GAIKWAD, M. E. Face recognition using Eigenfaces. In: **Computer Research and Development (ICCRD), 2011 3rd International Conference on**. IEEE, 2011. p. 302-306.

LATA, Y. Vijaya et al. Facial recognition using eigenfaces by PCA. 2009.

LI, Stan Z.; JAIN, Anil K. Handbook of Face Recognition. 2011. Disponível em <https://books.google.com.br/books?id=KSXwPmoqGWYC>

MENDONÇA, Marcelo Aparecido. **Classificação de gasolinas comerciais através de métodos estatísticos multivariáveis**. 2005. Dissertação de Mestrado. Universidade de São Paulo.

NOR'AINI, A. J.; RAVEENDRAN, P.; SELVANATHAN, N. A comparative analysis of feature extraction methods for face recognition system. In: **Sensors and the International Conference on new Techniques in Pharmaceutical and Biomedical Research, 2005 Asian Conference on**. IEEE, 2005. p. 176-181.

OpenCV. Open Source Computer Vision. Disponível em: <<http://opencv.org>> Acesso em: Mar. 2014.

ORL. The ORL Database of Faces, Cambridge University - Engineering Department, 1994. Disponível em <<http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>>. Acesso em Fev. 2014

PENTLAND, Alex; CHOUDHURY, Tanzeem. Face recognition for smart environments. **Computer**, v. 33, n. 2, p. 50-55, 2000.

PEREIRA, D. S. A. REZENDE, J. A. R. SILVA, J. C. SILVA, E. "Métodos de processamento e tratamento de imagens para reconhecimento facial." 2007.

PERLIBAKAS, Vytautas. Distance measures for PCA-based face recognition. **Pattern Recognition Letters**, v. 25, n. 6, p. 711-724, 2004.

SHAPIRO, L. G.; STOCKMAN, G. C. *Computer Vision*, Prentice Hall, 2001. 608p.

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

DA SILVA, Paulo Quintiliano; SANTA ROSA, Antônio Nuno de Castro. Reconhecimento facial aplicado a perícia criminal. **ICCyber'2004**, p. 176.

SIROVICH, Lawrence; KIRBY, Michael. Low-dimensional procedure for the characterization of human faces. **JOSA A**, v. 4, n. 3, p. 519-524, 1987.

SMITH, L. I. A Tutorial on Principal Components Analyzing – Discussion and singular value decomposition, 2002. Disponível em http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf Acesso em: Mar. 2014.

TURK, Matthew A.; PENTLAND, Alex P. Face recognition using eigenfaces. In: **Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on**. IEEE, 1991a. p. 586-591.

TURK, Matthew; PENTLAND, Alex. Eigenfaces for recognition. **Journal of cognitive neuroscience**, v. 3, n. 1, p. 71-86, 1991b.

Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE, 2001.

YANG, M; KRIEGMAN, D. J; AHUJA, N. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.