



**UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO  
UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA  
COMPUTAÇÃO**



**ALEX LIMA SILVA**

**REDUÇÃO DE CARACTERÍSTICAS PARA  
CLASSIFICAÇÃO DE IMAGENS DE FACES**

**MOSSORÓ - RN  
2016**

**ALEX LIMA SILVA**

**REDUÇÃO DE CARACTERÍSTICAS PARA  
CLASSIFICAÇÃO DE IMAGENS DE FACES**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação - associação ampla entre a Universidade do Estado do Rio Grande do Norte e a Universidade Federal Rural do Semi-Árido, para a obtenção do título de Mestre em Ciência da Computação.

Orientador: Profº Dr. Marcos Evandro Cintra

**MOSSORÓ - RN  
2016**

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do(a) autor(a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996, e Direitos Autorais Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data da defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu(a) respectivo(a) autor(a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

Dados Internacionais de Catalogação na Publicação (CIP)  
BIBLIOTECA CENTRAL ORLANDO TEIXEIRA - CAMPUS MOSSORÓ  
Setor de Informação e Referência

S586r Silva, Alex Lima.

Redução de características para classificação de imagens de faces /  
Alex Lima Silva. - Mossoró, 2016.  
98f: il.

Orientador: Marcos Evandro Cintra

Dissertação (MESTRADO EM CIÊNCIA DA COMPUTAÇÃO) -  
Universidade Federal Rural do Semi-Árido. Pró-Reitoria de Pesquisa e  
Pós-Graduação

1. Ciência da computação. 2. Biometria facial - reconhecimento. 3.  
Inteligência artificial. I. Título

RN/UFERSA/BOT/001

CDD 004

ALEX LIMA SILVA

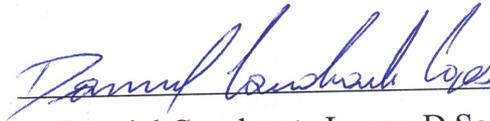
REDUÇÃO DE CARACTERÍSTICAS PARA CLASSIFICAÇÃO DE IMAGENS DE FACES

APROVADA EM: 10 / 12 / 2015

BANCA EXAMINADORA:



Marcos Evandro Cintra, D.Sc.  
Orientador



Danniell Cavalcante Lopes, D.Sc.  
Examinador Interno (UFERSA)



Ana Trindade Winck, D.Sc.  
Examinadora Externa (UFSM)

## Agradecimentos

À minha família, principalmente aos meus pais pelo suporte ao longo de toda minha vida acadêmica.

Aos todos os meus amigos, especialmente a Ronnison Reges Vidal, Nadyégila M. Alves Costa e Vladimir Ferreira L. da Silva pelo apoio e incentivo.

Aos colegas de mestrado e graduação pelo companheirismo ao longo de tantos anos de luta.

Ao Programa de Pós-Graduação em Ciência da Computação UERN/UFERSA, em especial ao professor Marcos Evandro Cintra pelos ensinamentos.

Agradecimento à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio financeiro.

E a todos que direta ou indiretamente fizeram parte da minha formação.

"Seja a mudança que deseja ver no mundo."

*—Mahatma Gandhi*

## Resumo

O reconhecimento de faces é uma das formas de biometrias mais utilizadas na atualidade. As redes sociais, os aplicativos para *smartphones* e as câmeras de vigilância eletrônica utilizam as técnicas de reconhecimento facial como meio de proporcionar entretenimento e segurança às pessoas. No entanto, apesar dos avanços recentes, as técnicas utilizadas atualmente não apresentam um desempenho equivalente ao dos seres humanos. Fatores como iluminação, pose e expressões faciais influenciam no desempenho dos métodos de reconhecimento facial, o que motiva o estudo e aprimoramento de novas técnicas que apresentem resultados melhores que as atuais. Em busca de avanços que ultrapassem essas limitações, foi realizado um estudo comparativo de métodos de reconhecimento facial com o propósito de conhecer as técnicas aplicadas atualmente e propor melhorias que representem um avanço na área. Nesta dissertação de mestrado é apresentado o desenvolvimento um método de reconhecimento facial baseado na extração de características realizada pelo método híbrido PCA/LDA. A classificação das características faciais extraídas pelo método proposto foi realizada por métricas de similaridade e classificadores. A análise dos resultados obtidos mostram, por meio de comparações com métodos encontrados na literatura, que o método proposto apresenta avanços no reconhecimento de padrões faciais.

**Palavras-chave:** Reconhecimento de faces, Reconhecimento de padrões, Extração de características, Inteligência artificial.

## Abstract

Face recognition of faces is one of the most used forms of biometrics today. Social networks, smartphones applications and electronic surveillance cameras use facial recognition techniques as a means of providing entertainment and security to people. However, despite recent advances, the techniques currently used do not have a performance equivalent to that of humans. Factors such as lighting, pose and facial expressions influence the performance of face recognition methods, and, thus, motivate the study and improvement of new techniques. Aiming at tackling such limitations of existing proposals, we conducted a comparative study of facial recognition methods in order to evaluate the current techniques found in the literature. This master thesis describes a face recognition method that uses a feature extraction technique based on a PCA/LDA hybrid method. The classification of facial features extracted by the proposed method was performed using similarity metrics and different classifiers. Our method was compared with methods found in the literature. The obtained results show that our method outperforms the other methods.

**Key-words:** Face Recognition, Pattern Recognition, Feature Extraction, Artificial Intelligence.

## Lista de Figuras

Figura 1 – Traços biométricos . . . . .	19
Figura 2 – Registro biométrico . . . . .	21
Figura 3 – Verificação biométrica . . . . .	21
Figura 4 – Identificação biométrica . . . . .	21
Figura 5 – Espaço de dados e espaço de características da PCA . . . . .	24
Figura 6 – Espaço de faces da PCA . . . . .	25
Figura 7 – Representação de uma imagem de face de 2 dimensões em 3 dimensões	27
Figura 8 – Face média . . . . .	27
Figura 9 – Subtração pela face média . . . . .	27
Figura 10 – Transformação de imagem em vetor . . . . .	28
Figura 11 – Eigenfaces . . . . .	30
Figura 12 – Espaço de dados e espaço de características da PCA . . . . .	32
Figura 13 – Espaço de faces da LDA . . . . .	33
Figura 14 – Face média por classe . . . . .	34
Figura 15 – Face média geral . . . . .	35
Figura 16 – Transformação de imagem em vetor . . . . .	35
Figura 17 – Métricas de similaridade . . . . .	39
Figura 18 – Exemplo de classificação usando o algoritmo $k$ -NN . . . . .	40
Figura 19 – Plano dos dados e Hiperplano gerado pelo SVM . . . . .	41
Figura 20 – Vetores de suporte . . . . .	41
Figura 21 – Neurônio artificial . . . . .	42
Figura 22 – Funções de ativação de redes neurais . . . . .	43
Figura 23 – Rede perceptron de múltiplas camadas . . . . .	44
Figura 24 – Modelo proposto para reconhecimento facial . . . . .	58
Figura 25 – Modelo proposto para extração de características . . . . .	59
Figura 26 – Modelo proposto para classificação de características . . . . .	60
Figura 27 – Variância acumulada das componentes da PCA no banco de imagens ORL . . . . .	63
Figura 28 – Variação acumulada das componentes da LDA no banco de imagens ORL . . . . .	64
Figura 29 – Classificação de características faciais por uma rede neural MLP . . . .	68

## Lista de tabelas

Tabela 1 – Desempenho dos métodos PCA, LDA, ICA-I e ICA-II . . . . .	51
Tabela 2 – Comparação de desempenho de métodos holísticos . . . . .	52
Tabela 3 – Comparação com o estado da arte no LFW <i>dataset</i> . . . . .	53
Tabela 4 – Comparação de desempenho de métodos geométricos . . . . .	54
Tabela 5 – Comparação de métodos híbridos . . . . .	55
Tabela 6 – Experimentos para avaliação de métricas de similaridade com o banco de imagens ORL . . . . .	65
Tabela 7 – Experimentos para avaliação do método $k$ -NN no banco de imagens ORL . . . . .	66
Tabela 8 – Experimento com máquina de vetores de suporte . . . . .	67
Tabela 9 – Resultado comparativo da distância Cosseno com rede neural MLP .	68
Tabela 10 – Experimentos com métricas e classificadores . . . . .	69
Tabela 11 – Experimentos com o banco de imagens ORL . . . . .	70
Tabela 12 – Experimentos com o banco de imagens JAFFE . . . . .	70

## Lista de quadros

1	Comparação de traços biométricos . . . . .	20
2	Aplicações de biometria . . . . .	22
3	Bancos de imagens de face . . . . .	46

## Lista de abreviaturas e siglas

PCA	<i>Principal Component Analysis</i>
LDA	<i>Linear Discriminant Analysis</i>
ICA	<i>Independent Component Analysis</i>
RNA	Rede Neural Artificial
MLP	<i>Multilayer Perceptron</i>
SVM	<i>Support Vector Machine</i>
SMO	<i>Sequential Minimal Optimization</i>

## Sumário

1	<b>INTRODUÇÃO</b>	14
1.1	Motivação	15
1.2	Problemática	15
1.3	Objetivo	17
1.4	Organização do trabalho	17
2	<b>REFERENCIAL TEÓRICO</b>	18
2.1	Biometria	18
2.1.1	Traços biométricos	18
2.1.2	Aplicações de biometria	22
2.2	Extração de características	23
2.2.1	Análise de Componentes Principais	23
2.2.1.1	Conceito	23
2.2.1.2	Descrição algébrica	26
2.2.2	Análise Discriminante Linear	31
2.2.2.1	Conceito	31
2.2.2.2	Descrição algébrica	34
2.3	Métricas de similaridade	38
2.3.1	Distância euclidiana	38
2.3.2	Distância Cosseno	38
2.3.3	Distância de Mahalanobis	38
2.3.4	Distância de Manhattan	39
2.4	Métodos de Classificação	40
2.4.1	<i>k</i> -Vizinhos Mais Próximos	40
2.4.2	Máquina de Vetor de Suporte	41
2.4.3	Redes Neurais Artificiais	42
2.5	Conclusão	44
3	<b>REVISÃO BIBLIOGRÁFICA</b>	45
3.1	Repositórios de imagens faciais	45
3.2	Métodos de Reconhecimento Facial	49
3.2.1	Métodos holísticos	50
3.2.2	Métodos geométricos	54
3.2.3	Métodos híbridos	55
3.3	Conclusão	56

---

4	PROPOSTA DE SOLUÇÃO . . . . .	57
4.1	Extração de características . . . . .	58
4.2	Reconhecimento de faces . . . . .	60
4.3	Validação da proposta . . . . .	61
5	EXPERIMENTOS . . . . .	62
5.1	Experimento para determinar a dimensionalidade dos vetores de características . . . . .	62
5.2	Experimentos para avaliação de métricas de similaridade . . . . .	65
5.3	Experimentos para avaliação do método $k$ -NN . . . . .	66
5.4	Experimentos com Máquina de Vetores de Suporte . . . . .	66
5.5	Experimento para avaliação de redes neurais . . . . .	67
5.6	Análise dos experimentos . . . . .	68
6	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS . . . . .	71
6.1	Trabalhos futuros . . . . .	71
6.2	Publicações . . . . .	72
	REFERÊNCIAS . . . . .	73
	 ANEXOS . . . . .	 78
	ANEXO A – CÓDIGOS . . . . .	79
A.1	Código para determinar a dimensionalidade dos vetores de ca- racterísticas . . . . .	79
A.1.1	Banco de imagens JAFFE . . . . .	79
A.1.2	Banco de imagens ORL . . . . .	84
A.2	Código para testes com métricas de similaridade . . . . .	89
A.2.1	Banco de imagens JAFFE . . . . .	89
A.2.2	Banco de imagens ORL . . . . .	94
A.3	Código para extração de características . . . . .	98
A.3.1	Banco de imagens JAFFE . . . . .	98
A.3.2	Banco de imagens ORL . . . . .	102

## 1 INTRODUÇÃO

O reconhecimento facial está a cada dia mais presente na vida das pessoas. As redes sociais, por exemplo utilizam métodos de reconhecimento facial para identificar pessoas de forma automática. Celulares e *tablets* possuem *softwares* que verificam a identidade do proprietário e empresas de vigilância eletrônica são capazes de identificar criminosos em meio a uma multidão devido aos métodos de reconhecimento de faces.

A vantagem do reconhecimento facial sobre outros métodos biométricos é o fato de não ser um método intrusivo, que pode ser realizado sem que o indivíduo interaja de forma direta com o sistema (ZENICOLA; SILVA, 2013). Além disso, diferente do reconhecimento por digitais ou por retina, não há necessidade de um hardware especial para captar a informação. Uma *webcam* ou câmera de celular, por exemplo, são equipamentos baratos e encontrados em qualquer loja de eletrônicos. Isso torna o reconhecimento facial um meio prático de realizar a identificação de um usuário.

No entanto, embora seja um dos métodos biométricos mais utilizados, ainda não existe uma abordagem com a mesma capacidade humana de reconhecer faces. O rosto humano possui uma estrutura básica presente em todas as pessoas. Porém, cada indivíduo possui um conjunto de características únicas, e estas características podem mudar tanto com o tempo quanto pela vontade de cada um, o que dificulta tal tarefa. Um humano é capaz de reconhecer uma pessoa familiar mesmo com muitos obstáculos como distância, sombras e até mesmo a visão parcial do rosto. O cérebro humano, com seu alto poder de processamento paralelo, é capaz de distinguir com muita eficiência os padrões faciais de indivíduos conhecidos.

Um computador, no entanto, precisa realizar inúmeros procedimentos para reconhecer um conjunto de padrões predeterminados e rotular uma face como conhecida ou desconhecida. Diferente de outros traços biométricos que permanecem os mesmos ao longo de anos ou décadas, o padrão facial é mutável e dinâmico. A aparência da face pode apresentar mudanças constantes causadas por expressões faciais, barba, maquiagem ou iluminação. Todos esses fatores influenciam no desempenho de métodos de reconhecimento de faces.

A principal contribuição deste trabalho é o desenvolvimento de um método de reconhecimento facial composto pelo método híbrido de extração de características PCA/LDA. Como contribuições secundárias: (i) elaboração de uma tabela de referência de repositórios de imagens de faces com suas principais características; (ii) estudo comparativo de métodos de reconhecimento facial com diferentes abordagens.

## 1.1 Motivação

Apesar dos avanços trazidos pelas técnicas recentes, o problema geral de reconhecimento de faces continua em aberto, já que a maioria dos sistemas atuais apenas alcança bons resultados quando certas condições de restrição são estabelecidas, tais como iluminação uniforme, expressão facial neutra ou rosto completamente visível. Assim, quando as imagens de face capturadas apresentam variações de posição, iluminação, expressão, escala, oclusão ou acessórios de vestuário, o desempenho é prejudicado (ZHAO et al., 2003).

Essas limitações têm estimulado pesquisas para desenvolver técnicas de reconhecimento de face mais robustas às possíveis variações na imagem. Em busca de oferecer meios para ultrapassar essas limitações, este trabalho apresenta o desenvolvimento de um método para classificação de imagens de face que busca construir uma representação precisa dos padrões faciais para aprimorar seu reconhecimento.

## 1.2 Problemática

Neste capítulo são descritos os fatores que ocasionam problemas relacionados a tarefa de reconhecimento facial automático. O problema de reconhecimento facial pode ser definido como: dada uma imagem de face de teste e um banco de imagens de faces, identificar ou verificar a identidade da pessoa na imagem de teste (BHATTACHARYYA et al., 2009). A face humana possui uma estrutura complexa e dinâmica, o que dificulta a modelagem de seus padrões. Sua aparência é afetada por diversos fatores tais como:

- **Ambiente** - O plano de fundo das imagens de face podem afetar o desempenho do método, principalmente métodos que seguem a abordagem holística, pois utilizam toda a imagem para realizar o reconhecimento.
- **Expressões faciais** - As expressões faciais alteram a estrutura padrão de uma face, reduzindo ou aumentando da boca boca, exibindo os dentes ou ocultando os olhos.
- **Iluminação** - A iluminação é um dos fatores que mais pode influenciar no reconhecimento de uma face. O ângulo da iluminação, sua intensidade, cor e dispersão (difusa ou focada) alteram a aparência da face na imagem, causando variações de desempenho.

- **Obstruções** - Alguns objetos, tais como óculos, cachecóis e chapéus, podem ocultar parcialmente ou completamente a face do indivíduo, dificultando seu reconhecimento.
- **Pose** - A pose de um indivíduo pode ocultar parte de seu rosto, desse modo evitando que seu padrão facial seja capturado na imagem.
- **Tempo** - As mudanças faciais ao longo o tempo podem interferir no reconhecimento facial. Algumas alterações podem ocorrer mesmo a curto prazo como crescimento da barba ou crescimento do cabelo.

Para gerar um modelo que descreva as características faciais que contemple todos esses fatores são necessárias imagens que apresentem essas variações. Quanto mais variações, mais imagens são necessárias para descrevê-las e isso leva a outro problema, a dimensionalidade.

O problema da dimensionalidade, conhecido como maldição da dimensionalidade, pode ser descrito por casos em que as instâncias do problema apresentam um número de atributos muito elevado. Muitos problemas tratados por métodos de aprendizado de máquina, como o reconhecimento de padrões em imagens, apresentam esse problema. Considerando que cada *pixel* da imagem é um atributo, uma imagem de 1024 por 1024 *pixels* teria mais de um milhão de atributos, tornando o problema intratável. Se cada atributo for visto como uma coordenada em um espaço  $d$ -dimensional, em que  $d$  é o número de atributos, o hipervolume que representa esse espaço cresce exponencialmente com a adição de novos atributos (FACELI et al., 2011).

A possível solução para este problema é utilizar técnicas para reduzir o número de dimensões, que aplicadas ao conjunto de dados brutos, podem refinar o número de atributos, melhorar o desempenho do modelo induzido e reduzir seu custo computacional. As técnicas de redução de dimensionalidade podem ser divididas em duas abordagens:

### **Agregação**

As técnicas de redução de dimensionalidade que seguem a abordagem de agregação aplicam uma transformação no conjunto de dados para gerar novos atributos formados pela combinação dos atributos originais. Análise de Componentes Principais, do inglês *Principal Component Analysis* (PCA) e Análise Discriminante Linear, do inglês *Linear Discriminant Analysis* (LDA), são exemplos de técnicas de redução de dimensionalidade baseadas em agregação.

### **Seleção de atributos**

As técnicas de seleção de atributos para redução de dimensionalidade realizam uma seleção no conjunto de atributos original, mantendo apenas os mais relevantes e descartando o restante.

As técnicas de extração de características aplicadas neste trabalho seguem a abordagem de agregação, no qual imagens  $n$ -dimensionais, onde  $n$  é o número de *pixels*, são mapeadas para um espaço de características  $m$ -dimensional, onde  $m$  é o número de atributos do vetor de características da imagem, tal que  $m < n$ .

### 1.3 Objetivo

O objetivo deste trabalho é o desenvolvimento e avaliação de um método de classificação de imagens de face que, por meio da aplicação de uma estratégia para a extração mínima de características para obter uma representação fiel aos padrões faciais de um indivíduo, apresente avanços no reconhecimento de padrões faciais.

### 1.4 Organização do trabalho

Este trabalho está organizado da seguinte forma:

No Capítulo 2 está o referencial teórico, no qual são descritos métodos de extração de características e reconhecimento de padrões utilizados na proposta, no Capítulo 3 é apresentado um estudo comparativo entre métodos de reconhecimento facial com base na revisão da literatura, no Capítulo 4 é descrito o método híbrido para classificação de imagens de faces proposto neste trabalho, no Capítulo 5 são descritos e analisados os experimentos realizados para validação deste trabalho, no Capítulo 6 são apresentadas sugestões de trabalhos futuros, as contribuições realizadas durante o desenvolvimento deste trabalho e as considerações finais.

## 2 REFERENCIAL TEÓRICO

Este capítulo descreve os principais conceitos relacionados com as técnicas usadas para o desenvolvimento deste trabalho. São descritos conceitos de biometria, técnicas de extração de características, métricas de similaridade e métodos de classificação.

### 2.1 Biometria

Biometria é a ciência que estabelece a identidade de um indivíduo baseada em seus atributos físicos, químicos ou comportamentais (JAIN; FLYNN; ROSS, 2007). A tecnologia biométrica é um conjunto de técnicas aplicadas para capturar, analisar e identificar pessoas por meio de seus traços biométricos. Traços biométricos são características únicas pertencentes a cada indivíduo que podem ser utilizadas para identificá-lo. A tecnologia biométrica oferece vantagens em relação a outros métodos tradicionais de identificação, tais como senhas, documentos e *tokens*, pois fornecem mais segurança quanto a autenticidade dos dados.

#### 2.1.1 Traços biométricos

Os traços biométricos podem ser atributos físicos (tais como face, íris, retina, geometria da mão, impressão da palma da mão e da digital, ouvido, arcada dentária, DNA) e comportamentais (tais como voz, assinatura e dinâmica de digitação) (JAIN; BOLLE; PANKANTI, 1998). A Figura 1 ilustra exemplos de traços biométricos.

Os traços apresentados na Figura 1 são escolhidos por serem atributos únicos em cada indivíduo, embora sejam características universais. Traços biométricos não podem ser perdidos ou esquecidos, são difíceis de serem copiados, compartilhados ou distribuídos. Os métodos de autenticação biométrica requerem que a pessoa autenticada esteja presente na hora e lugar da autenticação, evitando que pessoas não autorizadas possam ser autenticadas. Cada técnica biométrica analisa um conjunto de traços biométricos específicos, selecionados para cada tipo de aplicação. A seguir é descrita uma lista de atributos biométricos e os traços biométricos examinados por tais atributos.

Figura 1 – Traços biométricos



- **Arcada dentária:** Forma de dentes contínuos;
- **Assinatura:** Pressão, direção, tempo de aceleração e o comprimento dos traços;
- **DNA:** Código de DNA que pode ser extraído de células de sangue, pele e outras substâncias corporais;
- **Face:** Distância de características faciais específicas (olhos, nariz, boca);
- **Geometria da mão:** Estimativa de comprimento, largura, espessura, a forma e a área de superfície da mão;
- **Impressão digital:** Curvas Ridge com porção de elevada fricção, estruturas porosas, travessões e marcas;
- **Íris:** Textura da íris, como sardas, coronas, tiras, sulcos e criptas.
- **Ouvido:** Dimensão do ouvido visível.
- **Retina:** Padrão dos vasos na retina do olho, como os vasos sanguíneos na parte de trás do olho.
- **Ritmo de digitação:** Intervalo de digitação entre as teclas.
- **Voz:** Palavras e tons.

Os traços biométricos descritos possuem características particulares que definem o tipo de biometria e, portanto, o tipo de aplicação na qual ela é utilizada. Cada biometria possui características e campos de aplicação específicos. Alguns pesquisadores como [Srivastava \(2013\)](#), [Jain, Flynn e Ross \(2007\)](#) identificaram um conjunto de fatores que permitem comparar os diferentes tipos de biometria de acordo com as características dos traços biométricos utilizados, descritos a seguir:

- **Acessibilidade:** o traço biométrico é acessível à leitura;
- **Desempenho:** o traço possibilita um bom reconhecimento;
- **Evasão:** o traço biométrico pode ser fraudado;
- **Mensurabilidade:** o traço biométrico é mensurável;
- **Permanência:** o traço biométrico mantém suas principais características ao longo do tempo;
- **Unicidade:** Cada indivíduo possui traços biométricos com características únicas;
- **Universalidade:** o traço biométrico está presente em todos os indivíduos.

O Quadro 1 apresenta um comparativo de biometrias com base nas características dos traços biométricos utilizados.

Quadro 1 – Comparação de traços biométricos

Biometria	Acessibilidade	Desempenho	Evasão	Mensurabilidade	Permanência	Unicidade	Universalidade
Arcada dentária	Baixo	Médio	Alto	Baixo	Baixo	Baixo	Baixo
Assinatura	Alto	Médio	Alto	Médio	Baixo	Alto	Baixo
DNA	Alto	Alto	Baixo	Baixo	Alto	Alto	Alto
Face	Alto	Baixo	Alto	Médio	Médio	Médio	Alto
Geometria da mão	Médio	Médio	Médio	Alto	Baixo	Médio	Alto
Íris	Médio	Alto	Baixo	Médio	Alto	Alto	Alto
Impressão digital	Alto	Médio	Médio	Alto	Alto	Alto	Médio
Impressão da Palma da mão	Alto	Médio	Médio	Alto	Alto	Alto	Médio
Ouvido	Médio	Alto	Baixo	Médio	Médio	Médio	Baixo
Retina	Alto	Alto	Baixo	Baixo	Alto	Baixo	Alto
Voz	Baixo	Médio	Alto	Médio	Baixo	Alto	Baixo
Ritmo de digitação	Baixo	Baixo	Médio	Baixo	Baixo	Baixo	Baixo

Fonte: (SRIVASTAVA, 2013)

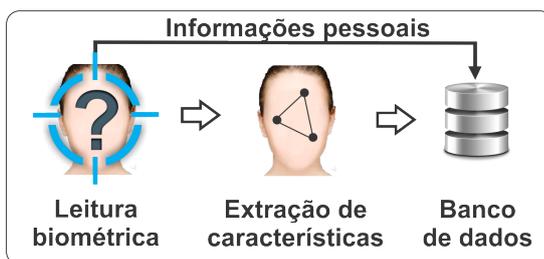
Nesse comparativo os atributos presentes nas biometrias são classificados em 3 níveis: Alto, Médio e Baixo. Por exemplo, a alta unicidade da íris indica que a mesma é um traço biométrico de alta distinção entre as pessoas.

A autenticação é o ato de estabelecer ou confirmar alguém, ou alguma coisa, como autêntico, isto é, que as alegações feitas por ou sobre a coisa é verdadeira (BHATTACHARYYA et al., 2009). Autenticação biométrica é o uso da biometria para reconhecimento, identificação ou verificação, de um traço biométrico de um indivíduo com o objetivo de autenticar sua identidade. Os traços biométricos podem também ser definidos como os atributos analisados pelas técnicas biométricas.

Nas Figuras 2, 3 e 4 são descritos os processos de *registro biométrico*, *verificação biométrica* e *identificação biométrica*.

A Figura 2 descreve o processo de cadastro de dados pessoais e biométricos de um indivíduo em um sistema de gerenciamento de identidade. O sistema realiza a leitura biométrica, extrai as características que devem ser utilizadas no reconhecimento e as armazena num banco de dados, juntamente com os dados pessoais associados ao indivíduo.

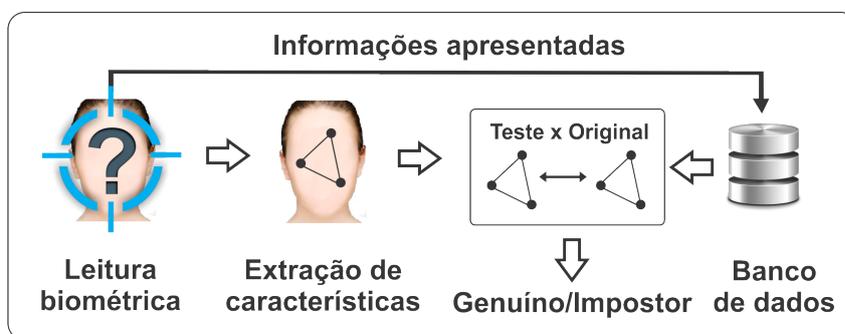
Figura 2 – Registro biométrico



Fonte: Autoria própria

As Figuras 3 e 4 descrevem o processo de reconhecimento biométrico, que pode ser realizado por meio de verificação ou identificação.

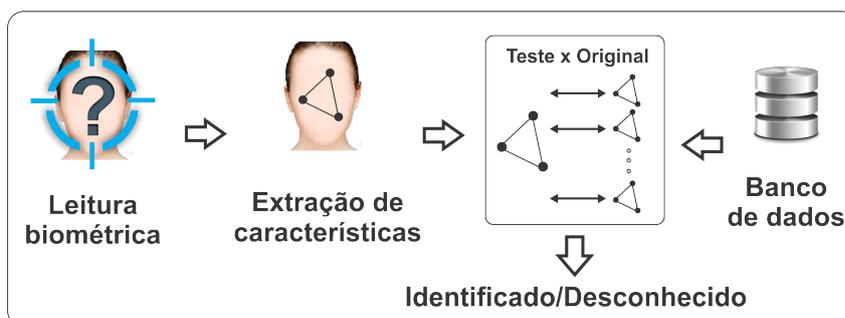
Figura 3 – Verificação biométrica



Fonte: Autoria própria

A Figura 3 descreve o processo de verificação biométrica, no qual o sistema lê os dados pessoais, tais como nome de usuário, senha, número de identidade, e biométricos apresentados. O sistema verifica se os dados biométricos apresentados contém o mesmo padrão que os dados biométricos previamente armazenados para o usuário com os mesmos dados pessoais. Caso a verificação seja positiva, o sistema reconhece o usuário como genuíno, caso contrário, o sistema o reconhece como um impostor.

Figura 4 – Identificação biométrica



Fonte: Autoria própria

A Figura 4 descreve o processo de identificação biométrica, no qual o sistema captura o traço biométrico do indivíduo e o compara com os demais traços biométricos registrados. Caso o padrão biométrico capturado seja identificado entre os padrões registrados, o indivíduo é identificado, caso contrário o indivíduo é declarado como desconhecido.

### 2.1.2 Aplicações de biometria

A biometria possui um amplo campo de aplicações nas mais diversas áreas. O Quadro 2 apresenta uma lista de aplicações de acordo com a área de aplicação.

Quadro 2 – Aplicações de biometria

Comercial	Governamental	Forense
Celulares Comércio eletrônico Controle de acesso Terminal bancário	Carteira de identidade Carteira de motorista Carteira de reservista Passaporte	Busca por desaparecidos Exame de paternidade Identificação de corpos Investigação criminal

Fonte: Autoria própria

Cada aplicação apresentada no Quadro 2 possui requisitos específicos. As aplicações forenses, por exemplo, utilizam atributos biométricos com alta unicidade para investigar crimes e esclarecer questões sobre paternidade. Já o trabalho de [Diniz et al. \(2013\)](#) utiliza o reconhecimento de expressões faciais de alunos para avaliar o seu aprendizado em um ambiente virtual. São esses requisitos que definem qual biometria deve ser utilizada. A análise comparativa da Tabela 1 e os requisitos de aplicações como as mostradas no Quadro 2 permite determinar qual biometria é mais adequada a uma aplicação específica.

O reconhecimento facial oferece diversas vantagens sobre outras biometrias. Há tecnologias biométricas que dependem de uma participação ativa do usuário para que o mesmo seja identificado, como por exemplo, para o reconhecimento de digital ou da palma da mão, é preciso que o usuário coloque a mão sobre o sensor para que seus dados sejam lidos. A identificação por retina requer que o usuário posicione o olho em frente à câmera para que a imagem seja capturada. O uso de reconhecimento facial dispensa a necessidade de o usuário realize uma ação para o reconhecimento acontecer. Isso oferece vantagens em aplicações como vigilância eletrônica.

Por outro lado, a aquisição dos dados para métodos biométricos geralmente apresenta problemas. Técnicas de reconhecimento de mãos e dedos, por exemplo, podem ocasionar falhas caso a epiderme do usuário esteja danificada no momento da leitura. Métodos de identificação por íris ou retina requerem um equipamento especializado e

são sensíveis ao movimento do corpo (JAFRI; ARABNIA, 2009). O reconhecimento por voz é suscetível a ruídos do ambiente. As assinaturas podem ser modificadas, esquecidas e falsificadas.

Já o reconhecimento facial dispensa equipamentos especializados, podendo ser realizado por meio de câmeras simples, permitindo o reconhecimento de mais de um indivíduo ao mesmo tempo com apenas um equipamento, o que não acontece com outras biometrias. Outra vantagem dos algoritmos de reconhecimento facial é que, aliados a técnicas de pré-processamento de imagens adequadas, eles podem compensar variações de escala, orientação, iluminação e expressões faciais.

## 2.2 Extração de características

A extração de características pode ser definida como a captura das informações mais relevantes de um conjunto de dados fornecido como entrada para realizar uma classificação (DEVIJVER; KITTLER, 1982). Para capturar essas informações, as técnicas de extração de características, a partir dos dados de entrada, realizam transformações para gerar um novo conjunto de dados, o qual representa as características do conjunto de dados original. Essas transformações buscam eliminar informações irrelevantes ou redundantes e representar os dados com o mínimo de características necessárias.

Esta seção apresenta técnicas de extração de características denominadas análise de componentes principais e análise discriminante linear utilizadas neste trabalho.

### 2.2.1 Análise de Componentes Principais

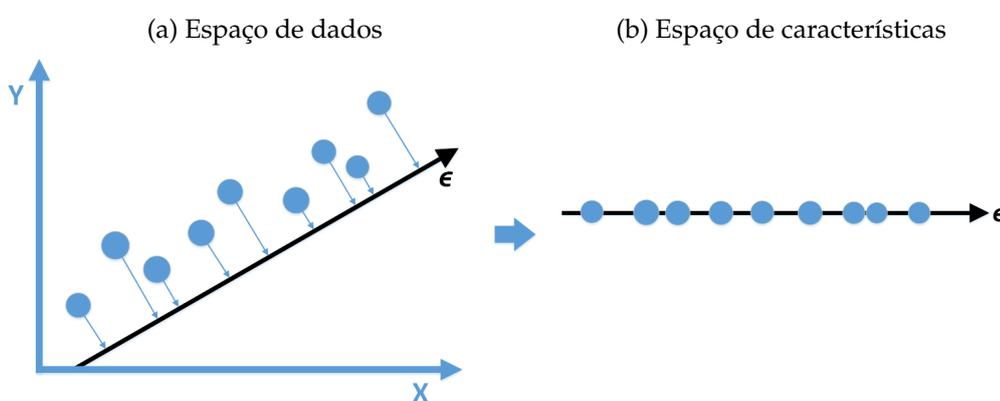
A seguir, a técnica análise de componentes principais é descrita de duas formas. A subseção 2.2.1.1 descreve de forma conceitual a análise de componentes principais aplicada ao reconhecimento de faces. A subseção 2.2.1.2 descreve os procedimentos algébricos utilizados na análise de componentes principais .

#### 2.2.1.1 Conceito

O reconhecimento de padrões em imagens é um exemplo de problema em que o número de atributos é muito elevado, onde faz-se necessário o uso de técnicas para

transformar o conjunto de dados original e um conjunto de dados com um número menor de atributos. Neste caso, a redução de dimensionalidade atua de forma a extrair características do conjunto de dados, transformando a imagem em um vetor de características. A Análise de Componentes Principais, do inglês *Principal Component Analysis* (PCA), é uma técnica de redução de dimensionalidade que, a partir de um conjunto de dados de entrada multidimensional, obtém um novo conjunto de dados de menor dimensionalidade que representa os dados originais. Esta técnica obtém a redução por meio de um mapeamento dos objetos representados em um espaço de dados de alta dimensionalidade para um espaço de características com menor dimensionalidade. A Figura 5 ilustra um exemplo de como a técnica PCA realiza a redução de dimensionalidade.

Figura 5 – Espaço de dados e espaço de características da PCA



Fonte: Autoria própria

Na Figura 5-a, cada ponto representa um objeto no espaço de dados, no qual cada eixo representa um atributo. Cada objeto possui dois valores, um para cada atributo, que definem seu lugar no espaço de dados. O objetivo da PCA é mapear esses pontos para um espaço com um número menor de dimensões, como ilustrado na Figura 5-b. A PCA é utilizada com o princípio de que a classificação de objetos que apresentem maior dispersão no espaço de dados é mais precisa do que a classificação de objetos agrupados em uma única região. Assim, a PCA gera um novo espaço de dados, chamado de espaço de características, a partir da construção do vetor  $\epsilon$ , que representa a direção na qual os objetos apresentam a maior dispersão no espaço de dados. Ao projetar os objetos do espaço de dados no espaço de características, os objetos passam a ser representados por vetores de características em um espaço de menor dimensionalidade.

Há casos em que os objetos possuem um número muito elevado de atributos, de modo que os dados podem estar dispersos em mais de uma direção no espaço de dados. Nesses casos, a PCA pode gerar um espaço de características em que cada eixo representa uma direção na qual os dados estão dispersos. Assim, cada vetor  $\epsilon$ , possui um valor associado que indica sua variância. A variância é uma medida estatística que

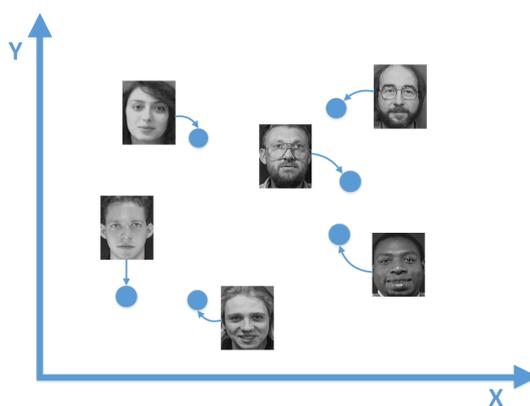
indica a dispersão de um conjunto de dados, ou seja, o quanto os objetos estão dispersos em um dado eixo.

Turk e Pentland (1991) aplicaram a PCA, como uma técnica de extração de características, em imagens de face, na qual consideraram cada *pixel* de uma imagem como um atributo, representado por uma dimensão no espaço de dados. A proposta de Turk e Pentland (1991) é compor um espaço de dados formado pelas imagens de face e, por meio da PCA, gerar um espaço de características que reduz a dimensionalidade das imagens, chamado de espaço de faces. Os vetores de características originários desse processo foram utilizados para realizar o reconhecimento de faces. A PCA aplicada ao reconhecimento facial é também conhecido como *eigenfaces*, devido ao nome dado aos vetores que geram o espaço de faces.

As *eigenfaces* podem ser visualizadas como imagens de características. As características presentes nas *eigenfaces* são variações de aparência presentes nas imagens de face, tais como variações de aparência causadas devido a mudanças de iluminação, pose e expressões faciais. Quando concentradas em uma *eigenface*, aparentam a imagem de uma face borrada ou distorcida.

Assim como as imagens no espaço de dados possuem um valor para cada atributo, os vetores de características possuem um valor para cada *eigenface* no espaço de faces. A Figura 6 exemplifica, de forma análoga, a projeção de 6 imagens no espaço de características.

Figura 6 – Espaço de faces da PCA



Fonte: Autoria própria

A Figura 6 ilustra um espaço de faces no qual as imagens de faces são representadas. Os pontos marcados dentro desse espaço representam as projeções das imagens de face. A posição de cada projeção é determinada pelos vetores de características extraídas de cada imagem. Assim, como as projeções das imagens de face estão mapeadas em pontos no espaço formado pelos eixos X e Y no exemplo da Figura 6, as imagens de face são projetadas no espaço das faces formado pelas *eigenfaces*.

Ao projetar uma imagem de face no espaço de faces, ela passa a ser representada por um vetor de características que indica suas coordenadas neste espaço. As projeções realizadas pelo PCA intensificam as variações entre as imagens de face, de modo que quanto mais diferentes elas forem, mais dispersas elas são projetadas no espaço de características. O reconhecimento facial é realizado projetando-se uma imagem de teste no espaço de faces e calculando sua similaridade com as demais projeções. A similaridade é calculada utilizando uma métrica, sendo a distância euclidiana uma das mais usuais.

O método *eigenfaces* é formado por 7 etapas:

1. Cálculo da face média
2. Subtração pela face média
3. Transformação das imagens em vetores
4. Construção da matriz de covariância
5. Cálculo das *eigenfaces*
6. Cálculo dos vetores de características
7. Cálculo de similaridade

Estas etapas são descritas a seguir.

#### 2.2.1.2 Descrição algébrica

##### 1. Cálculo da face média

Uma imagem é uma matriz de *pixels*, cujo valor de cada *pixel* indica sua cor ou intensidade. A Figura 7 mostra um exemplo de uma imagem de face em duas dimensões projetada em 3 dimensões, na qual a terceira dimensão é representada pelos valores dos *pixels* da imagem.

A face média, na Figura 8, é o resultado da soma *pixel a pixel* de todas as imagens do conjunto de treinamento dividido pelo número total de imagens.

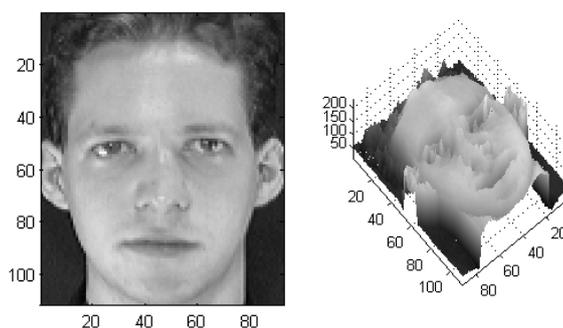
O cálculo da face média é descrito na Equação 2.1.

$$M = \frac{\sum_{i=1}^N I_i}{N} \quad (2.1)$$

Onde,

Figura 7 – Representação de uma imagem de face de 2 dimensões em 3 dimensões

(a) Face em 2 dimensões (b) Face em 3 dimensões



Fonte: Autoria própria

Figura 8 – Face média



Fonte: Autoria própria

- $M$  é a face média.
- $I_i$  é a  $i$ -ésima imagem de face.
- $N$  é o número de imagens do conjunto de treinamento.

## 2. Subtração pela face média

Ela é feita subtraindo-se, separadamente, cada uma das imagens do conjunto de treinamento pela face média.

Figura 9 – Subtração pela face média



Fonte: Autoria própria

O cálculo da subtração das imagens do conjunto de treinamento pela face média é

descrito na Equação 2.2.

$$\Phi_i = I_i - M \quad (2.2)$$

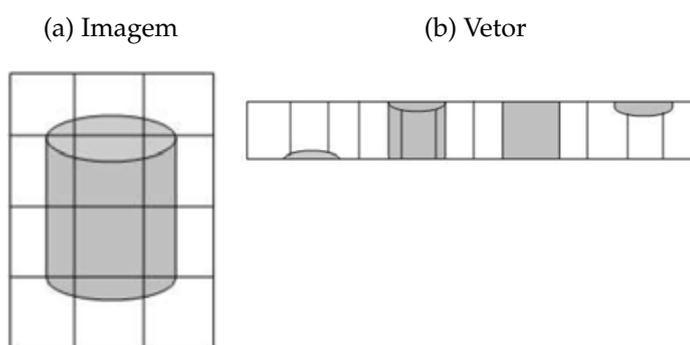
Onde,

- $\Phi_i$  é a  $i$ -ésima imagem resultante da subtração de uma imagem de treinamento pela face média.
- $I_i$  é a  $i$ -ésima imagem de face do conjunto de treinamento.
- $M$  é a face média.

### 3. Transformação das imagens em vetores

A conversão de uma imagem em um vetor consiste em concatenar as linhas da imagem, unindo o último *pixel* de cada linha com o primeiro *pixel* da linha seguinte, formando um vetor de *pixels*, como ilustrado na Figura 10.

Figura 10 – Transformação de imagem em vetor



Fonte: Autoria própria

A Figura 10 ilustra o exemplo de uma imagem de um cilindro é transformada em vetor pela concatenação de suas linhas. Este processo é realizado com as imagens de  $\Phi_i$ , as quais, convertidas em vetor formam as colunas da matriz  $\Phi$ .

### 4. Construção a matriz de covariância

Matriz de covariância é uma matriz quadrada que contém a relação de variância entre um conjunto de variáveis em relação ao seu valor médio e em relação umas as outras. A diagonal principal da matriz de covariância contém as variâncias das variáveis e os elementos de fora da diagonal principal contém as covariâncias entre os pares de variáveis.

A matriz de covariância  $\mu$  do método *eigenfaces* expressa o quanto as imagens de face variam em relação a face média e em relação as outras imagens de face. A diagonal principal contém a variância de cada imagem de face em relação a face

média. Os elementos de fora da diagonal principal contém a covariância entre as imagens. Os elementos da matriz de covariância são calculados pela Equação 2.3.

$$C = \begin{bmatrix} (\Phi_1)(\Phi_1) & (\Phi_1)(\Phi_2) & \dots & (\Phi_1)(\Phi_n) \\ (\Phi_2)(\Phi_1) & (\Phi_2)(\Phi_2) & \dots & (\Phi_2)(\Phi_n) \\ \dots & \dots & \dots & \dots \\ (\Phi_n)(\Phi_1) & (\Phi_n)(\Phi_2) & \dots & (\Phi_n)(\Phi_n) \end{bmatrix} \quad (2.3)$$

ou simplificando,

$$C = \begin{bmatrix} (I_1 - M)(I_1 - M) & (I_1 - M)(I_2 - M) & \dots & (I_1 - M)(I_n - M) \\ (I_2 - M)(I_1 - M) & (I_2 - M)(I_2 - M) & \dots & (I_2 - M)(I_n - M) \\ \dots & \dots & \dots & \dots \\ (I_n - M)(I_1 - M) & (I_n - M)(I_2 - M) & \dots & (I_n - M)(I_n - M) \end{bmatrix} \quad (2.4)$$

Onde,

- $\Phi_i$  é uma imagem de face subtraída pela face média,
- $I$  é uma imagem de face,
- $M$  é a face média.

## 5. Cálculo das *eigenfaces*

As *eigenfaces* do PCA e seus valores associados são, respectivamente, os autovetores e autovalores calculados a partir da matriz de covariância  $\mu$ . As *eigenfaces* são os vetores que definem os eixos do espaço de faces e seus valores associados indicam o quanto os dados estão dispersos ao longo desse eixo. As *eigenfaces* podem ser calculadas pela Equação 2.5.

$$E = \det(C - \lambda I) \quad (2.5)$$

Onde,

- $C$  é a matriz de covariância,
- $\lambda$  são os autovalores,
- $I$  é matriz identidade, isto é, uma matriz cujos valores da diagonal principal são iguais a 1 e os demais iguais a 0.

A Figura 11 ilustra um conjunto de exemplos de *eigenfaces* redimensionadas para o tamanho das imagens do conjunto de treinamento.

Figura 11 – Eigenfaces



Fonte: Autoria própria

## 6. Cálculo dos vetores de características

As imagens de face são representadas pela soma ponderada das *eigenfaces*. Os pesos dessa combinação formam o vetor de características da imagem de face. Para calcular o peso de cada *eigenface* na representação de uma imagem de face, multiplica-se a imagem de face, subtraída da média, por cada *eigenface*. Os valores resultantes formam o vetor de características da imagem. A construção do vetor de características das *eigenfaces* é descrito na Equação 2.6.

$$P_i = U * \Phi_i \quad (2.6)$$

simplificando,

$$P_i = U * (I_i - M) \quad (2.7)$$

Onde,

- $P_i$  é o peso correspondente a  $i$ -ésima *eigenface* para a imagem.
- $U$  é uma *eigenface*.
- $\Phi$  é a imagem de face subtraída da face média.
- $I_i$  é  $i$ -ésima imagem do conjunto de treinamento.
- $M$  é face média, calculada no passo 1.

$P$  é um vetor de  $k$  dimensões que indica a posição da imagem no espaço de faces.  $k$  é o número de *eigenfaces* utilizado para representação da imagem de face, ou seja, uma *eigenface* para cada dimensão do vetor de características. A extração do vetor de características pode ser vista como a projeção da imagem de face em um espaço de características.

## 7. Cálculo da similaridade

O reconhecimento pode ser realizado por meio de uma métrica de similaridade. Uma das métricas de similaridade mais comum é a distância euclidiana, descrita pela Equação 2.8:

$$D(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2.8)$$

Onde,

- $p_i$  é o vetor de características de uma imagem de teste.
- $q_i$  é o vetor de características de uma imagem de treinamento.

Para determinar a qual classe pertence uma imagem de face calcula-se a distância  $D$  entre o vetor de características que representa a imagem de teste e os demais vetores de características utilizados no treinamento. Uma face é classificada como sendo da classe  $k$  quando a imagem mais próxima pertence a  $k$  e sua distância é menor que um limite de similaridade, um limiar, determinado de maneira empírica. Este limite existe para determinar o quão uma imagem de teste da classe  $k$  deve parecer com as imagens de treinamento desta mesma classe, evitando que um impostor se passe por um indivíduo legítimo.

### 2.2.2 Análise Discriminante Linear

A seguir, a técnica análise discriminante linear é descrita duas formas. A subseção 2.2.2.1 descreve de forma conceitual a análise discriminante linear. A subseção 2.2.2.2 descreve os procedimentos algébricos utilizados na análise discriminante linear.

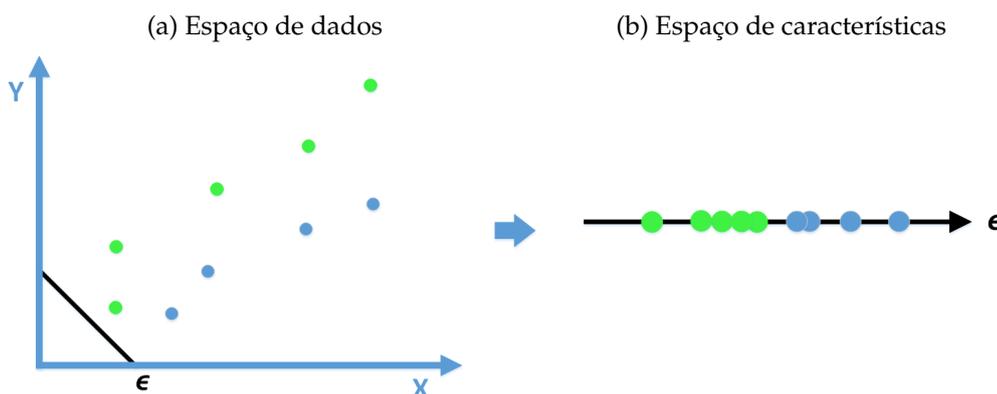
#### 2.2.2.1 Conceito

A técnica PCA é bastante utilizada devido a seu desempenho na redução de dimensionalidade dos dados. Porém, as características extraídas por esta técnica não tem como foco a separação de classes, uma vantagem apresentada pela técnica análise discriminante linear.

A Análise Discriminante Linear, do inglês *Linear Discriminant Analysis* (LDA), é uma técnica de redução de dimensionalidade utilizada para problemas de classificação

(HIDAYAT et al., 2011). A partir de um conjunto de dados multidimensionais rotulados, a LDA gera um novo conjunto de dados de menor dimensionalidade que representa as classes dos dados originais. Esta técnica realiza a redução por meio de um mapeamento dos objetos representados em um espaço de dados de alta dimensionalidade para um espaço de características de menor dimensionalidade. A Figura 12 ilustra um exemplo de como a técnica LDA realiza a redução de dimensionalidade.

Figura 12 – Espaço de dados e espaço de características da PCA



Fonte: Autoria própria

Na Figura 12-a, cada ponto representa um objeto no espaço de dados, no qual cada eixo representa um atributo. Cada objeto possui dois valores, um para cada atributo, que definem seu lugar no espaço de dados. O objetivo da LDA é mapear esses pontos para um espaço de características de menor dimensionalidade, chamado espaço de faces. Para realizar o mapeamento, a LDA gera o vetor  $\epsilon$  para construir um espaço de características que busca minimizar a dispersão intra-classe e maximizar a dispersão interclasse, isto é, aproximar os objetos de uma mesma classe e distanciar os objetos de classes distintas. Como ilustrado na Figura 12-b, os objetos do espaço de dados, ao serem mapeados para o espaço de características gerado pela LDA, são alocados próximos de outros objetos da mesma classe, facilitando o processo de classificação.

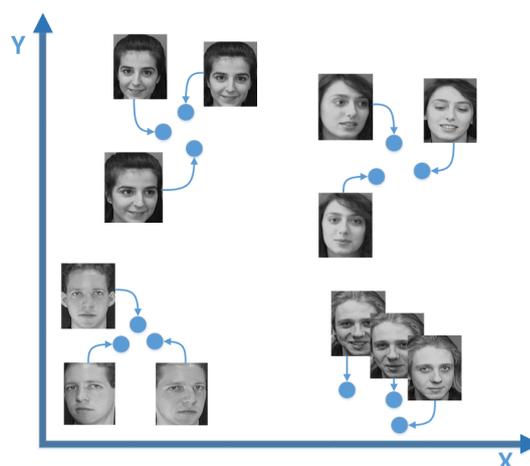
Bellhumer, Hespanha e Kriegman (1997) aplicaram a LDA, como uma técnica de extração de características, em imagens de face, para formar um espaço de dados no qual cada ponto representava uma imagem, sendo cada *pixel* um atributo da imagem, representado por uma dimensão nesse espaço. Por meio da LDA, foi gerado um espaço de características cuja distância entre as projeções de um mesmo indivíduo fosse menor que a distância entre projeções de indivíduos distintos. Desse modo, a representação obtida pela LDA possibilita um meio para a discriminação dos dados pela classe. A LDA aplicada ao reconhecimento de faces também é conhecida como *fisherfaces*, devido ao nome dado aos vetores que formam o espaço de faces.

De forma análoga as *eigenfaces*, as *fisherfaces* podem ser visualizadas como imagens de características. As características presentes nas *fisherfaces* são variações de aparência

presentes nas imagens de cada indivíduo, tais como variações de aparência causadas devido a mudanças de iluminação, pose e expressões faciais. Quando concentradas em uma *fisherface*, aparentam a imagem de uma face borrada ou distorcida.

Assim como as imagens no espaço de dados possuem um valor para cada atributo, os vetores e características possuem um valor para cada *fisherface*. A Figura 13 ilustra um exemplo de imagens projetadas no espaço de características gerado pela LDA.

Figura 13 – Espaço de faces da LDA



Fonte: Autoria própria

Na Figura 13, imagens de face de 4 indivíduos estão projetadas no espaço de características gerado pela LDA. A LDA busca realizar a projeção de uma imagem de face no espaço de características que maximize dispersão entre classes e minimize dispersão intra-classes, assim quanto mais diferente for uma pessoa da outra, mais distantes deverão estar suas projeções, e quanto mais parecidas forem as imagens de uma pessoa mais próximas deverão estar suas projeções.

Para realizar o reconhecimento, projeta-se a imagem de teste no espaço de faces e utiliza-se uma métrica de similaridade para calcular a distância entre a projeção da imagem de teste e as demais projeções. A classe da projeção mais próxima é atribuída a projeção de teste.

O *fisherfaces* é composto por 8 etapas:

1. Cálculo da face média por classe.
2. Cálculo da face média geral.
3. Transformação das imagens em vetores.
4. Construção da matriz de dispersão intra-classe.
5. Construção da matriz de dispersão inter-classe.

6. Cálculo das *fisherfaces*.
7. Cálculo dos vetores de características.
8. Cálculo da similaridade.

Estas etapas são descritas a seguir.

### 2.2.2.2 Descrição algébrica

#### 1. Cálculo da face média por classe

A face média por classe é o resultado da soma *pixel a pixel* de todas as imagens de uma mesma classe do conjunto de treinamento dividido pelo total de imagens da classe, como ilustrado na Figura 14.

Figura 14 – Face média por classe



Fonte: Autoria própria

O cálculo da face média da classe é descrito pela Equação 2.9.

$$M = \frac{\sum_{i=1}^N I_i}{N} \quad (2.9)$$

Onde,

- **M** é a face média.
- $I_i$  é a  $i$ -ésima imagem de face de uma classe.
- **N** é o número de imagens do conjunto de treinamento de uma classe.

#### 2. Cálculo da face média geral

A face média é o resultado da soma *pixel a pixel* de todas as imagens do conjunto de treinamento dividido pelo total de imagens, como ilustrado na Figura 15.

O cálculo da face média de todo o conjunto de treinamento é descrito pela Equação 2.10.

$$M = \frac{\sum_{i=1}^N I_i}{N} \quad (2.10)$$

Figura 15 – Face média geral



Fonte: Autoria própria

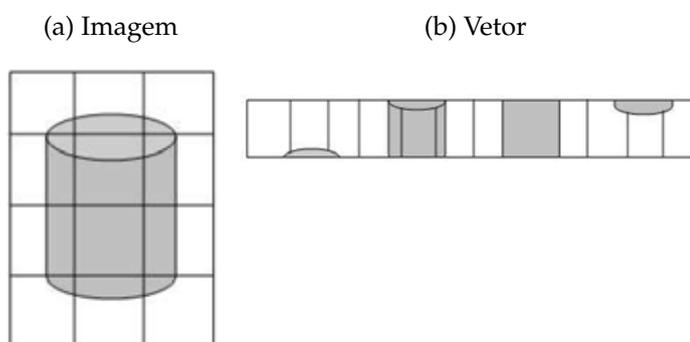
Onde,

- $M$  é a face média.
- $I_i$  é a  $i$ -ésima imagem de face.
- $N$  é o número de imagens do conjunto de treinamento.

### 3. Transformação das imagens em vetores

A transformação de uma imagem em um vetor consiste em concatenar as linhas da imagem, unindo o último *pixel* de cada linha com o primeiro *pixel* da linha seguinte, formando um vetor de *pixels*, como ilustrado na Figura 16.

Figura 16 – Transformação de imagem em vetor



Fonte: Autoria própria

A Figura 16 ilustra o exemplo de uma imagem de um cilindro é transformada em vetor pela concatenação de suas linhas. Este processo é realizado com as imagens do conjunto de treinamento.

### 4. Construção da matriz de dispersão intra-classe

A matriz de dispersão intra-classe  $S_w$  calcula o quanto os dados estão dispersos dentro da própria classe, isto é, o quanto as imagens de um mesmo indivíduo diferem umas das outras.

$$S_w = \sum_{i=1}^c (x_i - \mu_k)(x_i - \mu_k)^T \quad (2.11)$$

Onde,

- $x_i$  é a  $i$ -ésima imagem transformada em vetor pertencente a classe  $k$ .
- $\mu$  é a face média da classe  $k$ .

### 5. Construção da matriz de dispersão inter-classe

A matriz de dispersão inter-classe  $S_b$  calcula o quanto os dados estão dispersos entre as classes, isto é, o quanto as imagens de indivíduos distintos diferem umas das outras. O cálculo da matriz de dispersão é descrito pela Equação 2.12.

$$S_b = \sum_{i=1}^c (\mu_k - \mu)(\mu_k - \mu)^T \quad (2.12)$$

Onde,

- $\mu_k$  é a face média da  $k$ -ésima classe.
- $\mu$  é a face média de todas as imagens.

### 6. Cálculo das *fisherfaces*

As *fisherfaces* do LDA e seus valores associados são, respectivamente, os autovetores e autovalores da matriz  $W$  descrita na Equação 2.13.

$$W = (S_w^{-1} S_b) \quad (2.13)$$

Onde,

- $S_w$  é a matriz de dispersão intra-classe,
- $S_b$  é a matriz de dispersão inter-classe,

As *fisherfaces* podem ser calculadas pela Equação 2.14.

$$F = \det(W - \lambda I) \quad (2.14)$$

Onde,

- $W$  é a matriz de separação,
- $\lambda$  são os autovalores,
- $I$  é matriz identidade, isto é, uma matriz cujos valores da diagonal principal são iguais a 1 e os demais iguais a 0.

## 7. Cálculo dos vetores de características

As imagens de face são representadas pela soma ponderada das *fisherfaces*. Os pesos dessa combinação foram os vetores de características das imagens de face. Para calcular o peso de cada *fisherfaces* na representação de uma imagem de face, multiplica-se a imagem de face, subtraída da média, por cada uma *fisherfaces*. O cálculo do vetor de características é descrito pela Equação 2.15.

$$P_i = U_i * (I_i - M) \quad (2.15)$$

Onde,

- $P_i$  é o peso correspondente a  $i$ -ésima *fisherface*.
- $U_i$  é a  $i$ -ésima *fisherface*.
- $I_i$  é  $i$ -ésima imagem de face.
- $M$  é a face média, calculada no passo 1.

$P$  é um vetor de  $k$  dimensões que indica a posição daquela imagem no espaço de faces.  $k$  é o número de *fisherface* utilizado para construir o espaço de faces, isto é, uma *fisherface* para cada dimensão.

## 8. Cálculo de similaridade

O reconhecimento pode ser realizado por meio do uso de um classificador ou de uma métrica de similaridade. Uma das métricas de similaridade mais comum é a distância euclidiana, descrita pela Equação 2.16:

$$D(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2.16)$$

Onde,

- $p_i$  é o vetor de características de uma imagem de teste,
- $q_i$  é o vetor de características de uma imagem de treinamento.

A classe de uma imagem de teste é determinada calculando-se a distância  $D$  entre o vetor de características da imagem de teste e os demais vetores de características. Uma face é classificada como sendo da classe  $k$  quando a imagem mais próxima pertence a  $k$  e sua distância é menor que um limite de similaridade, um limiar, determinado de maneira empírica. Este limite existe para determinar o quão uma imagem de teste da classe  $k$  deve parecer com as imagens de treinamento desta mesma classe, evitando que um indivíduo se passe por impostor.

## 2.3 Métricas de similaridade

Métricas de similaridade são medidas que avaliam o quão similar são dois objetos calculando a distância entre eles em um espaço de entrada. Métricas diferentes podem ser utilizadas para avaliar a similaridade de diferentes formas. As métricas de similaridade mais comuns são: distância euclidiana, distância Cosseno, distância de Mahalanobis e distância de Manhattan, descritas a seguir.

### 2.3.1 Distância euclidiana

A distância euclidiana determina a similaridade entre dois objetos calculando a distância da reta entre os 2 pontos,  $p$  e  $q$ . Dados dois pontos,  $p$  e  $q$ , a distância euclidiana é dada pela Equação 2.17.

$$D(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2.17)$$

### 2.3.2 Distância Cosseno

A distância Cosseno calculo o cosseno do ângulo formado pelos pontos  $p$  e  $q$ . Dados dois pontos,  $p$  e  $q$ , a distância Cosseno é dada pela Equação 2.18.

$$D(p, q) = \frac{\sum_{i=1}^n (p_i - q_i)}{\sqrt{\sum_{i=1}^n (p_i)^2} \sqrt{\sum_{i=1}^n (q_i)^2}} \quad (2.18)$$

### 2.3.3 Distância de Mahalanobis

A distância de Mahalanobis é uma métrica que determina a similaridade entre objetos em relação a distribuição dos dados no espaço de entrada. A distância de

Mahalanobis pode ser considerada uma distância euclidiana normalizada (LINDEN, 2009). Dados dois pontos,  $p$  e  $q$ , a distância de Mahalanobis é dada pela Equação 2.19.

$$D(p, q) = \sqrt{\sum_{i=1}^n \frac{(p_i - q_i)^2}{\sigma_i^2}} \quad (2.19)$$

Onde,

- $\sigma_i$  é o desvio padrão da amostra.

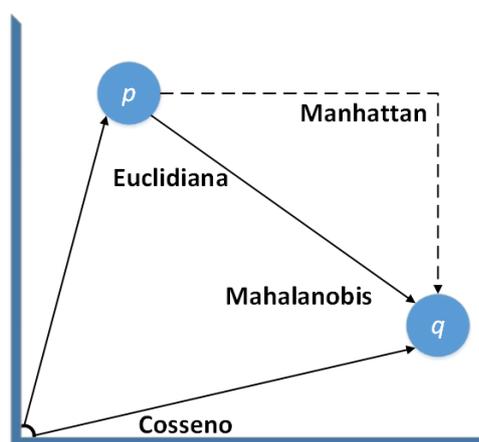
### 2.3.4 Distância de Manhattan

A distância de Manhattan determina a similaridade entre dois objetos calculando a diferença absoluta entre seus pontos. A distância de Manhattan é também conhecida como distância  $L_1$  ou *City-block*. Dados dois pontos,  $p$  e  $q$ , a distância de Manhattan é dada pela Equação 2.20.

$$D(p, q) = \sum_{i=1}^n |p_i - q_i| \quad (2.20)$$

A Figura 17 ilustra como as distâncias são calculadas.

Figura 17 – Métricas de similaridade



Fonte: Autoria própria

Na Figura 17, as distâncias entre os pontos  $p$  e  $q$  são representadas pelas retas no gráfico. A reta tracejada indica a distância de Manhattan, a reta contínua entre os pontos indica a distância euclidiana e a distância formada pelo ângulo entre os pontos é a distância cosseno.

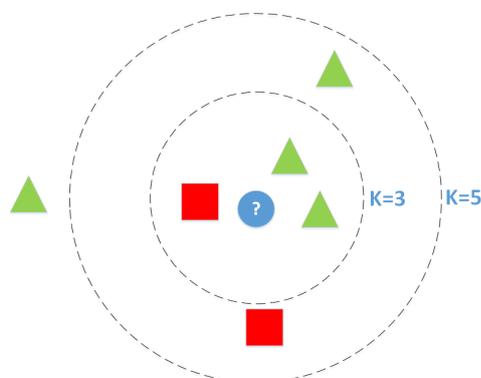
## 2.4 Métodos de Classificação

A seguir são descritos os métodos de classificação utilizados neste trabalho. São eles:  $k$ -NN, SVM e redes neurais artificiais.

### 2.4.1 $k$ -Vizinhos Mais Próximos

O algoritmo dos  $k$ -Vizinhos Mais Próximos, do inglês *k-Nearest Neighbour* ( $k$ -NN), baseia-se na hipótese de que dados similares tendem a se concentrar nas mesmas regiões no espaço de entrada, enquanto dados não similares tendem a se concentrar em regiões distintas. Neste algoritmo, cada objeto é representado por um ponto no espaço definido pelos valores de seus atributos. A similaridade entre dois objetos é definida por uma métrica que calcula a distância entre os dois pontos. O algoritmo  $k$ -NN classifica um novo objeto com base nos  $k$  exemplos do conjunto de treinamento mais próximos (FACELI et al., 2011).

Figura 18 – Exemplo de classificação usando o algoritmo  $k$ -NN



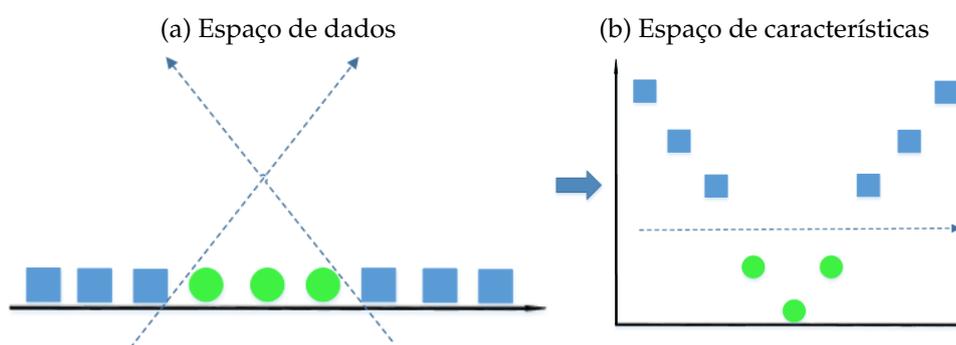
Fonte: Autoria própria

A Figura 18 ilustra o  $k$ -NN apresentando um conjunto de objetos distribuídos pelo espaço de entrada. Um objeto de teste (círculo) é inserido neste espaço para ser classificado pelo  $k$ -NN. As classes são indicadas pela cor e formato dos objetos. As circunferências ao redor do objeto de teste, indicam quais os  $k$  vizinhos mais próximos dele. O valor de  $k$  geralmente é determinado de forma empírica, portanto deve ser testado e avaliado de acordo com o desempenho do classificador. A classe que possuir o maior número de exemplos mais próximos, dentro do raio de  $k$  objetos, determina classe do objeto de teste.

### 2.4.2 Máquina de Vetor de Suporte

Máquina de Vetor de Suporte, do inglês *Support Vector Machine* (SVM), é um algoritmo de classificação com aprendizagem supervisionada que mapeia os dados de entrada para um hiperplano de características, no qual os dados são separados de forma linear. A Figura 19 ilustra um exemplo de como os dados podem ser classificados pelo SVM.

Figura 19 – Plano dos dados e Hiperplano gerado pelo SVM

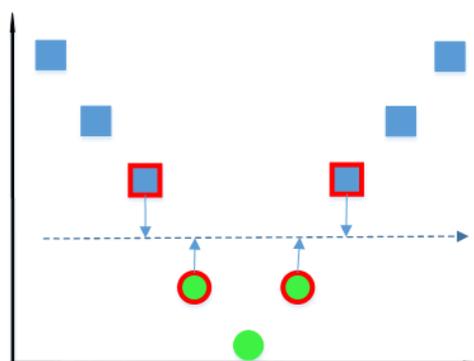


Fonte: Autoria própria

Na Figura 19, duas classes de dados são representadas pelas formas quadrado e círculo. Na Figura 19a, os dados estão representados no espaço de entrada, no qual não podem ser separados linearmente. Na Figura 19b, os dados estão mapeados no espaço de características multidimensional gerado pelo SVM.

Neste espaço de características, os dados são separados de forma linear por uma estrutura chamada superfície de decisão linear. A superfície de decisão linear é delimitada pelas exemplos mais próximas ao espaço entre as classes, chamados vetores de suporte. Os vetores de suporte são os vetores que demarcam a fronteira entre as classes no hiperplano de características, como ilustrado na Figura 20.

Figura 20 – Vetores de suporte



Fonte: Autoria própria

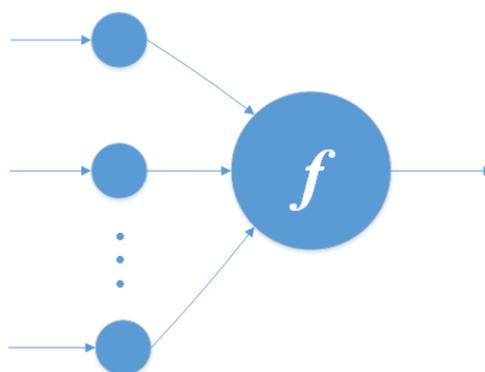
A Figura 20 mostra a classificação dos dados com base nos vetores de suporte, indicados com a borda vermelha. O mapeamento dos dados é realizado por uma função, chamada função de *kernel*, que mapeia os dados para espaço de características e com o apoio dos vetores de suporte, delimita a fronteira que separam as classes.

Em 1998, o pesquisador Platt (1998) propôs o algoritmo Otimização Mínima Sequencial, do inglês *Sequential Minimal Optimization* (SMO), para treinamento de SVM. O algoritmo SMO utiliza uma abordagem de “divisão e conquista” para reduzir o processamento necessário para encontrar o hiperplano que delimita a distância máxima entre os vetores de suporte de classes distintas, apresentando um desempenho superior ao algoritmo padrão de treinamento de SVM. Este algoritmo apresenta ainda resultados promissores em classificação multi-classe utilizando classificação em pares.

### 2.4.3 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNA) são sistemas inteligentes, cuja meta é interpretar e resolver problemas computacionais baseando seu funcionamento no cérebro humano. São compostas por unidades de processamento interconectadas, denominadas neurônios artificiais, que processam sinais para computar funções matemáticas. Os neurônios artificiais possuem uma estrutura simples, como ilustrada na Figura 21.

Figura 21 – Neurônio artificial

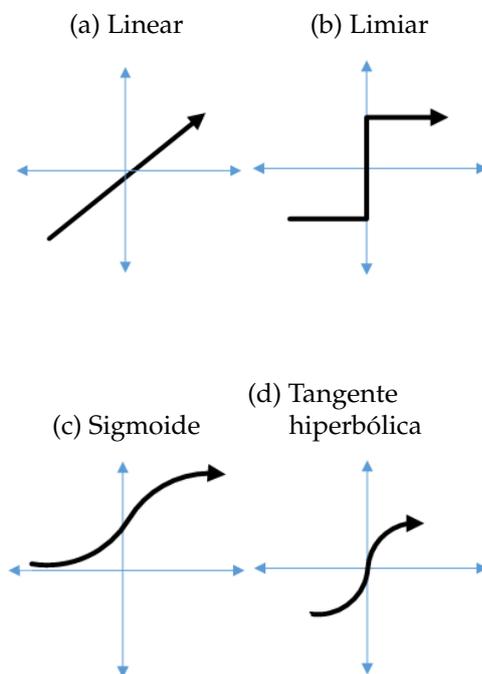


Fonte: Autoria própria

O neurônio artificial é composto por receptores que captam o dados de entrada e que são conectados a uma unidade de processamento, onde uma função é aplicada para gerar o sinal de saída, ou sinapse. A rede neural possui dois tipos de sinapses: inibitória ou excitatória. A sinapse inibitória ocorre quando o valor da função de ativação fica abaixo de um dado limiar, resultando em uma resposta negativa. A sinapse excitatória ocorre quando o valor da função de ativação fica acima de um dado limiar, resultando na ativação do neurônio ou resposta positiva.

A Figura 22 ilustra 4 funções de ativação de uma rede neural.

Figura 22 – Funções de ativação de redes neurais

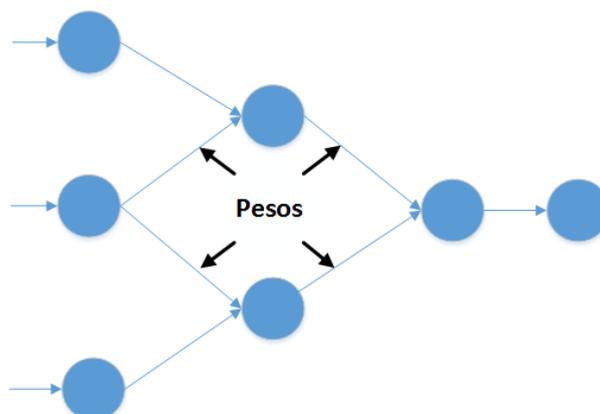


Fonte: Autoria própria

Os gráficos das funções linear, limiar, sigmoide e tangente hiperbólica estão ilustrados, respectivamente, nas Figuras 22-a, 22-b, 22-c e 22-d. Uma rede neural pode ser formada por diversos neurônios artificiais, organizados em uma ou mais camadas interligadas, chamadas de redes neurais *MultiLayer Perceptron* (MLP). No modelo MLP, os neurônios da camada de entrada são responsáveis por receber os dados de entrada. De acordo com a função de ativação, cada neurônio pode ou não ser ativado ao receber um dado sinal de entrada. Ao ser ativado, o neurônio passa o o sinal recebido para a camada seguinte por meio das conexões.

As conexões entre os neurônios de cada camada possuem pesos, isto é, valores que alteram os sinais propagados de uma camada para outra. A propagação de sinais pode ser unidirecional, como em redes neurais do tipo *Feedforward*, ou bidirecionais, como em redes do tipo recorrente. O processo de aprendizagem de uma rede neural consiste na adaptação dos pesos das conexões durante o processo de treinamento. No processo de treinamento são apresentados dados de entrada para a rede neural aprenda a reconhecer os padrões desejados. Após o treinamento, a rede pode ser testada para avaliar seu desempenho na classificação de dados. A Figura 23 ilustra um exemplo de rede neural de múltiplas camadas.

Figura 23 – Rede perceptron de múltiplas camadas



Fonte: Autoria própria

Na Figura 23, a rede neural possui 3 neurônios na camada de entrada, 2 neurônios na camada oculta e 1 neurônio na camada de saída, intercalados por conexões com seus respectivos pesos. O número de neurônios na camada de entrada corresponde ao número de atributos das instâncias do problema, enquanto que o número de neurônios na camada de saída corresponde ao número de classes que deve ser preditas. O número de neurônio na camada oculta é determinado de modo empírico.

## 2.5 Conclusão

Neste capítulo são descritos os principais conceitos sobre extração de características, métricas de similaridade e algoritmos de classificação utilizados neste trabalho.

### 3 REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta a revisão bibliográfica de trabalhos e pesquisas sobre reconhecimento de padrões faciais. Nele é apresentada uma base de informações sobre repositórios de imagens de face e suas principais características, e uma análise comparativa de métodos de reconhecimento de faces de diferentes abordagens, indicando as vantagens e desvantagens de cada método com base em avaliações experimentais.

#### 3.1 Repositórios de imagens faciais

Na literatura é possível encontrar diversos repositórios de faces utilizados pelos pesquisadores na validação de suas propostas para o uso de reconhecimento automático de faces. Tais repositórios apresentam características diferentes entre si em relação ao número total de imagens e número de imagens de cada indivíduo, além do tipo de características extraídas. O Quadro 3 apresenta informações de 15 repositórios de imagens de face encontrados na literatura.

No Quadro 3, a coluna “Nome” apresenta o nome do referido banco de imagens de faces; a coluna “Indivíduos” apresenta o número de indivíduos cujas imagens estão no banco, juntamente com a proporção entre homens e mulheres; a coluna “Amostras” indica quantas imagens foram obtidas de cada pessoa; a coluna “Acesso” mostra a disponibilidade de cada banco, classificando em duas categorias: “Livre” (qualquer pessoa pode ter acesso ao banco de imagens) ou “Restrito” (banco de imagens é liberado mediante solicitação ao pesquisador ou instituição responsável pelas imagens). As variações de aparência são fatores que modificam temporariamente a aparência da face e, portanto, influenciam na qualidade do reconhecimento. As variações de aparência nas imagens são indicadas por:

- **A** As imagens foram obtidas em diferentes ambientes, variando o plano de fundo.
- **E** As imagens apresentam faces com expressões, tais como alegria, tristeza ou surpresa.
- **I** As imagens apresentam variação de intensidade e ângulo da iluminação.
- **O** As imagens apresentam algum tipo de obstrução no rosto do indivíduo, tais como óculos, cachecol ou chapéu.
- **P** As imagens apresentam o rosto do indivíduo em posições diferentes (cabeça inclinada para um dos lados, abaixada ou levantada).

- T As imagens foram obtidas com diferença de alguns dias ou meses, podendo apresentar características de envelhecimento, crescimento ou corte de cabelo e/ou barba.

Quadro 3 – Bancos de imagens de face

Nome	Total de imagens	Indivíduos	Amostras	Acesso	Variações					
					Ambiente	Expressões	Iluminação	Obstruções	Poses	Tempo
CVL Database (SOLINA et al., 2003)	798	114 (108/6)	7	Restrito	✓				✓	
FacePix Database (ASU) (BLACK et al., 2011)	16290	30	181	Livre			✓			✓
FEI Face Database (THOMAZ; GIRALDI, 2010)	2800	200 (100/100)	14	Livre	✓	✓			✓	
Indian Face Database (KHARE; NAIK, 2014)	440	61 (39/22)	~11	Restrito	✓					✓
Labeled Faces in the Wild (LFW) (HUANG et al., 2007)	13233	13233	≥ 1	Livre	✓	✓	✓	✓	✓	✓
ORL Database of faces (SAMARIA; HARTER, 1994)	400	40 (36/4)	10	Livre		✓	✓		✓	✓
Psychological Image Collection at Stirling (PICS) < <a href="http://pics.stir.ac.uk/">http://pics.stir.ac.uk/</a> >	312	35	9	Livre			✓			✓
The AR Face Database < <a href="http://www2.ece.ohio-state.edu/~aleix/ARdatabase">http://www2.ece.ohio-state.edu/~aleix/ARdatabase</a> >	4000	126 (70/56)	26	Restrito		✓	✓	✓		✓
The Facial Recognition Technology (FERET) (PHILLIPS; MOON; RAUSS, 2000)	14126	1199	~11	Restrito	✓	✓	✓		✓	✓
The Japanese Female Facial Expression (JAFFE) (LYONS et al., 1998)	213	10 (0/10)	7	Livre		✓	✓			✓
The Sheffield Face Database (GRAHAM; ALLINSON, 1998)	564	20 (16/4)	26-54	Livre						✓
Yale Face Database (BELLHUMER; HESPANHA; KRIEGMAN, 1997)	165	15 (14/1)	11	Livre		✓	✓			
Yale Face Database B (GEORGHIADES; BELHUMEUR; KRIEGMAN, 2001)	5670	10	576	Livre			✓			✓
UND Biometrics < <a href="http://www.cse.nd.edu/~cvrl/CVRL/Data_Sets.html">http://www.cse.nd.edu/~cvrl/CVRL/Data_Sets.html</a> >	33287	487	68	Restrito	✓	✓	✓		✓	✓
University of Essex Face Database < <a href="http://cswww.essex.ac.uk/mv/allfaces/">http://cswww.essex.ac.uk/mv/allfaces/</a> >	7900	395	20	Livre		✓	✓	✓	✓	

Fonte: Autoria própria

Esses bancos de imagens são descritos com mais detalhes a seguir.

- 1. CVL Database** está disponível para acesso mediante a permissão de seu administrador por meio do endereço eletrônico <<http://www.lrv.fri.uni-lj.si/facedb.html>>. O banco de imagens CVL Face Database é um repositório de imagens de face desenvolvido por Solina et al. (2003), no Laboratório de Visão Computacional da Universidade de Ljubljana, Slovenia. Este banco de imagens de face é formado por 798 imagens de face de 114 pessoas, 7 imagens por pessoa, a maioria homens. As imagens possuem 640x480 pixels em formato JPEG e foram obtidas sob iluminação uniforme e plano de fundo neutro. As fotografias foram obtidas de diferentes ângulos e expressões faciais.
- 2. FacePix Database** está disponível para livre acesso no endereço eletrônico <<https://cubic.asu.edu/content/facepix-database>>. O banco de imagens de face FacePix Database foi

desenvolvido por *FacePixDatabase* no Centro para Computação Ubíqua Cognitiva na Universidade do Estado Arizona, EUA. Este banco possui 16290 imagens de 30 pessoas, 543 imagens por pessoa. As imagens foram obtidas de diferentes ângulos sob diferentes condições de iluminação.

3. **FEI Face Database** está disponível para livre acesso por meio do endereço eletrônico <<http://fei.edu.br/~cet/facedatabase.html>>. O banco de imagens FEI é um repositório de imagens de face desenvolvido no Laboratório de Inteligência Artificial da Faculdade de Engenharia Industrial (FEI) de São Bernardo do Campo, Brasil, por Thomaz e Giraldi (2010). Este banco de imagens possui 2800 imagens de face de 200 indivíduos, 100 homens e 100 mulheres, 14 imagens por indivíduo. As imagens estão em formato JPG e possuem 640x480 *pixels*.
4. **Indian Face Database (IFD)** está disponível para livre acesso por meio do endereço eletrônico <<http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase/>>. O banco de imagens IFD é um repositório de imagens de face desenvolvido Jain e Mukherjee (2002). Este banco possui cerca de 11 imagens de 40 indivíduos. As imagens estão em escala de cinza, no formato JPEG e possuem resolução de 640x480 *pixels*.
5. **Labeled Faces in the Wild (LFW)** está disponível para livre acesso por meio do endereço eletrônico <<http://vis-www.cs.umass.edu/lfw/>>. O banco de imagens *Labeled Faces in the Wild* (LFW) é um repositório de imagens de face desenvolvido na Universidade de Massachusetts, EUA, por Huang et al. (2007), em 2007, com objetivo de servir de material para pesquisas de reconhecimento facial. Este banco de imagens é formado por 13233 imagens de faces de 5749 indivíduos, com número variável de imagens por pessoa (1680 pessoas tem duas imagens ou mais). As imagens possuem 250x250 *pixels* em formato JPEG.
6. **Psychological Image Collection at Stirling (PICS)** está disponível para livre acesso por meio do endereço eletrônico <<http://pics.psych.stir.ac.uk/>>. O banco de imagens *Psychological Image Collection at Stirling* (PICS) é um repositório de imagens de face desenvolvido na Escola de Psicologia de Ciências Naturais da Universidade de Stirling, Inglaterra. Este banco é formado por 312 imagens de 35 indivíduos, com número variável de imagens por indivíduo (cada indivíduo possui ao menos 9 imagens). As fotografias foram obtidas com diferentes ângulos e expressões faciais. As imagens possuem 274x350 *pixels* em, escala de cinza, no formato GIF.
7. **The ORL Database of faces** está disponível para livre acesso por meio do endereço eletrônico <<http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>>. O bando de imagens Olivetti Research Limited (ORL), ou *The ORL Database of Faces* (SAMARIA; HARTER, 1994) é um repositório de imagens de faces utilizado em pesquisas de métodos de reconhecimento facial desde 1994. Este banco de imagens é formado

por 400 imagens de face de 10 indivíduos, 36 homens e 4 mulheres, que apresentam diversas expressões faciais sob diferentes iluminações e poses. As imagens estão no formato PGM e possuem 92x112 *pixels* em escala de cinza.

8. **The AR Face Database** está disponível para livre acesso por meio do endereço <<http://www2.ece.ohio-state.edu/~aleix/ARdatabase.html>>. O banco de imagens *The AR Face Database* foi desenvolvido no Centro de visão Computacional da Universidade do Estado de Ohio, EUA. Este repositório de imagens de face contém 4000 imagens coloridas de 126 pessoas, 70 homens e 56 mulheres, com número variante de imagens por pessoa. As imagens foram obtidas de diferentes ângulos, em diferentes condições de iluminação com pessoas apresentando diferentes expressões faciais ou objetos (óculos de sol e cachecol). As imagens possuem resolução de 768x576 *pixels*.
9. **JAFFE** está disponível para livre acesso através do endereço eletrônico <<http://www.kasrl.org/jaffe.html>>. O banco de imagens *The Japanese Female Facial Expression (JAFFE) Database* desenvolvido por Lyons et al. (1998), no departamento de psicologia na universidade Kyushu, Japão. Este repositório de imagens de face possui 213 imagens de 7 expressões faciais (6 expressões básicas e 1 neutra) de 10 modelos japonesas. As imagens estão no formato TIFF e possuem 256x256 *pixels* em escala de cinza.
10. **FERET** está disponível para acesso mediante solicitação de acesso ao Instituto Nacional de Padrões e Tecnologia (*National Institute of Standards and Technology - NIST*) no endereço eletrônico <<http://www.nist.gov/itl/iad/ig/feret.cfm>>. O banco de imagens *The Facial Recognition Technology (FERET)* é um dos resultados do Programa de Desenvolvimento de Tecnologia Contra Drogas do Departamento de Defesa norte-americano. Este repositório de imagens de face possui 14126 imagens de 1199 indivíduos, com número variável de imagens por indivíduo. O banco possui subconjuntos de imagens em escala de cinza ou colorida, e de tamanhos distintos.
11. **The Sheffield Face Database** está disponível para livre acesso no endereço eletrônico <<https://www.sheffield.ac.uk/eee/research/iel/research/face>>. O banco de imagens *The Sheffield Face Database* foi desenvolvido por Graham e Allinson (1998) na Universidade de Sheffield, Reino Unido. Este repositório possui 564 imagens de 20 indivíduos, 16 homens e 4 mulheres, com número variável de imagens por indivíduo. As imagens possuem 220x220 *pixels* em escala de cinza.
12. **Yale Face Database** está disponível para livre acesso no endereço eletrônico <<http://vision.ucsd.edu/content/yale-face-database>>. O banco de imagens *Yale Face Database* foi desenvolvido por Bellhumer, Hespanha e Kriegman (1997) na Universidade Yale. Este repositório possui 165 imagens de 15 indivíduos, 11 imagens por indivíduo.

O banco apresenta imagens com diferentes condições de iluminação, expressões faciais, poses e objetos (óculos). As imagens possuem 320x243 *pixels* em escala de cinza.

13. **Yale Face Database B** está disponível para livre acesso no endereço eletrônico <<http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/Yale%20Face%20Database.htm>>. O banco de imagens *Yale Face Database B* foi desenvolvido por **Georghiadis, Belhumeur e Kriegman (2001)** na Universidade Yale. Este repositório é uma extensão *Yale Face Database B* e possui 5760 imagens de 10 indivíduos, 576 imagens por indivíduo. O banco apresenta imagens com diferentes condições de iluminação, poses e expressões faciais. As imagens possuem 320x243 *pixels* em escala de cinza.

14. **UND Biometrics** está disponível para acesso mediante permissão do administrador da coleção de imagens no endereço eletrônico <[http://www.cse.nd.edu/~cvrl/CVRL/Data\\_Sets.html](http://www.cse.nd.edu/~cvrl/CVRL/Data_Sets.html)>. O *UND Biometrics* é uma coleção bancos de imagens de face em 2 e 3 dimensões desenvolvidos na Universidade de Notre Dame. O banco de imagens em 2 dimensões contém 33287 imagens de face de 487 indivíduos, possuindo de 4 a 227 imagens por indivíduo. As imagens possuem 1600x1200 ou 2272x1704 *pixels*.

15. **University of Essex Face Database** está disponível para livre acesso no endereço eletrônico <<http://cswww.essex.ac.uk/mv/allfaces/>>. O banco de imagens *University of Essex Face Database* foi desenvolvido na Universidade de Essex, Inglaterra. O banco apresenta 7900 imagens de 395 indivíduos, sendo 20 imagens por indivíduo. As imagens possuem 180x200 *pixels* no formato JPEG e apresentam diferentes condições de iluminação e oclusões (barba e óculos).

Ao realizar uma análise comparativa de algoritmos de reconhecimento facial, é recomendado o uso de conjuntos de dados padronizados para que os resultados possam ser diretamente comparados. Embora existam diversos bancos de imagens, a escolha adequada de um banco de imagens deve ser feita com base em determinadas características das faces que ele contém, de acordo com o foco do método proposto, como por exemplo, presença de expressões faciais, número de indivíduos e número de imagens por indivíduo contidas no banco.

### 3.2 Métodos de Reconhecimento Facial

O desempenho dos métodos apresentados nessa seção são comparados de acordo com suas taxas de acurácia, visando informações relevantes na escolha de métodos para validação de novas propostas.

Como dito anteriormente, o processo de reconhecimento automático de faces consiste em construir uma representação dos padrões faciais de um indivíduo e, por meio de comparações com outros padrões faciais, determinar a identidade do mesmo. Essa representação é construída a partir da extração de características de imagens de face, e a identidade, por sua vez, é determinada por meio de uma métrica de similaridade ou classificador.

A seguir, é apresentado uma análise comparativa de desempenho entre métodos de reconhecimento facial encontrados na literatura.

### 3.2.1 Métodos holísticos

Métodos holísticos ou baseados em aparência utilizam a imagem completa da face para extrair características gerais que representem o padrão facial do indivíduo. O reconhecimento é realizado calculando-se a similaridade entre uma imagem de teste e os demais padrões conhecidos. *Eigenfaces*, redes neurais artificiais e ICA, do inglês *Independent Component Analysis* são exemplos de métodos clássicos de reconhecimento facial que utilizam a abordagem holística.

A análise dos métodos holísticos apresentada é realizada com base na comparação entre os experimentos feitos com os métodos PCA, LDA e ICA arquitetura I e II (BARTLETT; MOVELLAN; SEJNOWSKI, 2002). Os algoritmos PCA, LDA e ICA são métodos de extração de características que, a partir de um conjunto de imagens, gera uma representação das características do indivíduo, que é utilizada para reconhecê-lo em comparações com imagens de teste. A similaridade entre as representações geradas para cada indivíduo e as imagens de teste podem ser estimadas por métricas de similaridade ou classificadores.

Os algoritmos podem apresentar variações no resultado de acordo com o conjunto de imagens no qual é testado e com a forma com que classifica as identidades. A métrica de similaridade pode influenciar de modo determinante no desempenho de um método, como é mostrado no trabalho de Delac, Grgic e Grgic (2005), no qual é proposto um estudo comparativo entre os algoritmos PCA, LDA e ICA, arquitetura I e II, aplicados ao problema de reconhecimento facial.

Em seu estudo comparativo, Delac, Grgic e Grgic (2005) selecionou o banco de de imagens de faces FERET, com base em suas características, para realizar os experimentos. Os experimentos avaliaram o desempenho dos métodos em realizar o reconhecimento facial na presença de 4 variações de aparência presentes nas imagens (Expressões faciais, Iluminação, Tempo 1 e Tempo 2), utilizando 4 métricas de similaridade. As métricas utilizadas foram a distância Euclidiana (EUC), distância de Manhattan (MAT), distância

Cosseno(COS) e distância de Mahalanobis(MAB). Os resultados dos experimentos são apresentados na Tabela 1, na qual estão em destaque os maiores índices de acurácia para cada métrica avaliada. As imagens utilizadas no teste de “Expressões faciais” apresentam faces com variações de expressões faciais, tais como felicidade, tristeza ou surpresa. As imagens utilizadas no teste de “Iluminação” apresentam variações no ângulo e intensidade da iluminação. As imagens utilizadas no teste “Tempo 1”, foram obtidas todas no mesmo dia, já as imagens utilizadas no teste “Tempo 2” foram obtidas com diferença de 18 meses, em relação às imagens do teste “Tempo 1”. Estão em destaque os valores que apresentam as maiores taxas de acurácia de cada algoritmo em cada teste. A coluna “Melhor método” apresenta a melhor combinação de método e métrica para o teste avaliado. As linhas “Ranking” e “Ranking geral” indicam a ordem e com qual métrica os algoritmos obtiveram as maiores taxas de acurácia em cada teste e durante todo o experimento, respectivamente. Para estes testes, as maiores taxas de acurácia foram obtidas com o uso da métrica cosseno, juntamente com o algoritmo ICA2.

Tabela 1 – Desempenho dos métodos PCA, LDA, ICA-I e ICA-II

Extrator	Taxa de acurácia (%)				Melhor método
	Métricas				
	Euclidiana	Cosseno	Mahalanobis	Manhattan	
Expressões faciais					
ICA-I	80.92	81.00	<b>81.51</b>	64.94	ICA-I+MAB
ICA-II	64.94	<b>83.85</b>	74.31	64.94	ICA-II+COS
LDA	78.08	81.51	<b>82.76</b>	<b>70.88</b>	LDA+MAB
PCA	<b>82.26</b>	81.00	82.18	64.94	PCA+EUC
Ranking	2	1	3	4	
Iluminação					
ICA-I	18.04	12.89	17.53	32.99	ICA-I+MAT
ICA-II	15.98	<b>64.95</b>	44.85	32.99	ICA-II+COS
LDA	26.80	20.62	<b>26.80</b>	<b>41.24</b>	LDA+MAT
PCA	<b>55.67</b>	12.56	25.26	32.99	PCA+EUC
Ranking	2	1	4	3	
Tempo 1					
ICA-I	<b>32.55</b>	32.27	31.86	25.62	ICA-I+EUC
ICA-II	28.81	<b>44.66</b>	31.99	25.62	ICA-II+COS
LDA	<b>34.76</b>	33.38	32.96	<b>27.70</b>	LDA+EUC
PCA	<b>36.29</b>	33.52	<b>33.52</b>	25.62	PCA+EUC
Ranking	2	1	4	3	
Tempo 2					
ICA-I	8.97	8.97	7.69	<b>14.53</b>	ICA-I+COS
ICA-II	16.24	<b>28.21</b>	<b>19.66</b>	14.53	ICA-II+COS
LDA	16.24	10.68	10.26	<b>16.67</b>	LDA+MAT
PCA	<b>17.09</b>	11.11	10.68	14.53	PCA+EUC
Ranking	2	1	4	3	
Ranking geral	2	1	4	3	

Fonte: (DELAC; GRGIC; GRGIC, 2005)

Os resultados apresentados na Tabela 1 mostram que o algoritmo ICA-II obteve taxa de acurácia maior que os demais algoritmos avaliados neste comparativo. O método ICA-II obteve seus melhores resultados combinado com a distância de Cosseno, indicando esta como a melhor métrica para obter os melhores resultados com este algoritmo. De forma análoga, o PCA obteve seus melhores resultados com a distância Euclidiana. Não é possível determinar qual métrica é mais adequada aos métodos ICA-I

e LDA, pois os mesmos apresentaram seus melhores resultados em diferentes métricas.

Foi observado que as menores taxas de acurácia foram obtidas nos testes relacionados a variações de aparência ao longo do tempo (Tempo 1 e Tempo 2), sendo que, nos testes que envolviam maior intervalo de tempo, os métodos apresentaram suas menores taxas de acurácia. Isto indica que fatores como envelhecimento, crescimento de cabelo ou barba são fatores que influenciam mais no reconhecimento que variações de aparência causadas por iluminação e expressões faciais.

A análise geral dos resultados realizados por [Delac, Grgic e Grgic \(2005\)](#), apresentados na Tabela 1, indicam que a qualidade do método de reconhecimento facial depende tanto de um algoritmo capaz de extrair as características que representem bem um padrão facial quanto de uma medida de similaridade adequada ao extrator selecionado.

Embora alguns métodos apresentem taxa de acurácia maior que outros, não é possível determinar que uma única combinação de entre os métodos e métricas avaliadas, seja realmente a melhor no reconhecimento de faces, pois não é possível afirmar que as diferenças entre as taxas de reconhecimento são estatisticamente significantes, apenas que para as características apresentadas pelo banco de imagens FERET, as combinações descritas na Tabela 1 apresentadas são as mais eficientes dentre as comparadas. Para melhor avaliar a capacidade do método, é necessário testá-lo com imagens com diferentes características.

Para que se possa inferir sobre o desempenho desses métodos no reconhecimento de faces com diferentes características, é necessário testá-los com diferentes bancos de imagens de face. A Tabela 2 apresenta o estudo comparativo realizado por [Khare e Naik \(2014\)](#) entre os algoritmos PCA, LDA e ICA (arquitetura I e II) com 5 repositórios de imagens com diferentes características. Os experimentos foram realizados com os bancos de imagens FEI *Face Database* ([THOMAZ; GIRALDI, 2010](#)), *Indian Face Database* (IFD) ([KHARE; NAIK, 2014](#)), ORL ([SAMARIA; HARTER, 1994](#)), FERET ([PHILLIPS; MOON; RAUSS, 2000](#)), *University of Essex* (UoE) e Sheffield (anteriormente UMIST) ([GRAHAM; ALLINSON, 1998](#)). Na primeira linha encontram-se os nomes dos bancos de imagens de face, juntamente com seus respectivos índices no Quadro 3, seguido do número total de imagens contidas no banco e do número de imagens por indivíduo. Os experimentos avaliam a taxa de acurácia dos métodos no reconhecimento de faces utilizando a distância euclidiana como métrica de similaridade.

Tabela 2 – Comparação de desempenho de métodos holísticos

Métodos	3-FEI (200/14)	4-IFD (61/11)	7-ORL (40/10)	10-FERET (3737/10)	15-UoE (395/20)	11-Sheffield (26-54)
PCA	85.23	74.17	91.25	82.26	88.54	95.80
LDA	<b>94.32</b>	<b>86.25</b>	<b>94.38</b>	78.06	<b>100.00</b>	<b>98.32</b>
ICA-I	87.50	71.67	91.25	<b>81.00</b>	88.45	95.80
ICA-II	88.54	74.17	90.63	64.94	88.54	94.96

Fonte: ([KHARE; NAIK, 2014](#))

Na Tabela 2 estão em destaque os maiores índices de acurácia obtidos por cada repositório. O método LDA apresenta o melhor desempenho em 5 dos 6 bancos de imagens testados, apresentando sua menor taxa de acurácia no banco de imagens FERET, indicando sensibilidade quando o número de classes(indivíduos) é muito maior que número de imagens por classe(amostras) quando se trata de um banco de imagens com tais variações. O algoritmo ICA apresenta sua maior taxa de acurácia no banco de imagens *Sheffield*, que apresenta apenas 1 variação de aparência, e a menor taxa de acurácia no banco de imagens FERET, que apresenta 5 variações de aparência, indicando que quanto mais variações, ou mais intensas elas foram, menor o desempenho do método.

Métodos de reconhecimento facial são formados por duas etapas: a extração de características, na qual são geradas representações de padrões faciais; e a classificação de características, na qual as características extraídas são classificadas como pertencentes ou não a um indivíduo. Alguns algoritmos como *eigenfaces*, *fisherfaces* e ICA utilizam um método para extração e outro para classificação de características. Porém, existem algoritmos, como alguns modelos de redes neurais, que realizam essas duas etapas em sua arquitetura, unido extrator e classificador em um único algoritmo.

O *DeepFace* é uma arquitetura de rede neural profunda, um modelo de rede neural que utiliza múltiplas camadas ocultas, desenvolvida por (TAIGMAN et al., 2014) para reconhecer faces na rede social *Facebook*. Dentre suas contribuições, seu trabalho apresenta uma arquitetura de rede neural profunda (*Deep Neural Net - DNN*) que, a partir de um banco de imagens de faces, obtém uma representação facial capaz de proporcionar um reconhecimento de alta acurácia, comprovada por meio de comparação com outros métodos.

A arquitetura da DNN do *DeepFace* possui 9 camadas e foi treinada utilizando o algoritmo *back-propagation*. Para o treinamento e teste da rede foram utilizados 3 bancos de imagens de face: *Labeled Faces in the Wild dataset (LFW)*, *The Social Face Classification (SFC) dataset* e *YouTube Faces (YTF) dataset*. O SFC foi utilizado apenas para testar as configurações da DNN, portanto seus resultados não foram comparados com os de outros algoritmos. O banco de imagens *YouTube Faces (YTF) dataset* é composto por imagens de vídeos obtidas do *YouTube*. Os resultados obtidos nos testes com o banco LFW são apresentados na Tabela 3.

Tabela 3 – Comparação com o estado da arte no LFW *dataset*

Método	Acurácia (%)
Joint Bayesian	92.42
Tom-vs-Pete	93.30
High-dim LBP	95.17
TL-Joint Bayesian	96.33
DeepFace	97.35

Fonte: (TAIGMAN et al., 2014)

O *DeepFace* obteve a taxa de 97.37% de acurácia em testes no banco de faces

LFW, ultrapassando os trabalhos de [Cao et al. \(2013\)](#), [Berg e Belhumeur \(2012\)](#), [Chen et al. \(2012\)](#) e [Chen et al. \(2013\)](#), que até o momento da publicação de [Taigman et al. \(2014\)](#), eram os trabalhos que haviam obtido os melhores resultados de classificação das imagens deste *dataset*, segundo o comparativo de [Huang et al. \(2007\)](#).

A seguir, são apresentadas propostas que utilizam a abordagem baseada em características para reconhecimento de faces.

### 3.2.2 Métodos geométricos

Métodos geométricos ou baseados em características processam a imagem da face para segmentar regiões distintas, tais como olhos, boca e nariz. Em seguida, são calculadas relações geométricas entre essas regiões, como por exemplo, a distância entre os olhos, tamanho do nariz, tamanho da boca, entre outras. Desse modo, um vetor de características geométricas é extraído da imagem de face ([JAFRI; ARABNIA, 2009](#)). O método de Kanede, filtro de Gabor e Grupos de Grafos Elásticos (GGE) são exemplos de métodos clássicos de reconhecimento facial que seguem a abordagem geométrica.

A Tabela 4 apresenta uma seleção de estudos comparativos de diversos autores sobre métodos de reconhecimento facial que seguem a abordagem baseada em características. Foram avaliados os métodos *Elastic Bunch Group Matching* (EBGM) ([BHAT; WANI, 2015](#)), Gabor Wavelet ([KEPENECKI, 2001](#)), Modelo Oculto de Markov (HMM - *Hidden Markov Models*) ([SAMARIA; HARTER, 1994](#)) ([IANCU; CORCORAN, 2011](#)) e Padrão Binário Local (LBP - *Local Binary Patter*) ([GIRISHA; SREEPATHI; KARIBASAPPA, 2014](#)). Os bancos de imagens utilizados foram ORL *Database of Faces* e FERET.

Tabela 4 – Comparação de desempenho de métodos geométricos

Métodos	7-ORL (40/10)	10-FERET (14126/1199)
EBGM	91.50	77.00
Gabor Wavelet	95.25	87.00
HMM	87.00	84.00
LBP	<b>98.00</b>	<b>90.72</b>

Fonte: Autoria própria

A comparação entre os resultados apresentados na Tabela 4 mostra que os algoritmos obtiveram taxa de acurácia maior no banco de imagens ORL que no FERET. Ao comparar os bancos com os dados apresentados na Tabela 3, observa-se que as principais diferenças estão no número de indivíduos presente em cada banco e na variação de aparência “Ambiente”, que indica que as imagens foram obtidas em diferentes lugares, ou seja, com diferentes planos de fundo, presente apenas no banco de imagens FERET. Isso indica que métodos baseados em características possuem seu

desempenho afetado quando o número de indivíduos a ser reconhecido é muito grande em relação ao número de imagens por indivíduo.

Segundo [Iancu e Corcoran \(2011\)](#), o algoritmo pode melhorar o desempenho ao se ajustar o número de estados, isto é, realizar testes para verificar qual a melhor configuração para cada caso. O algoritmo EBGM é sensível a variações de ângulo e pose, embora seja robusto com variações de iluminação e expressões faciais ([BHAT; WANI, 2015](#)).

### 3.2.3 Métodos híbridos

Métodos híbridos utilizam múltiplas técnicas combinadas de modo complementar, com a finalidade de obter um sistema de reconhecimento facial automático mais robusto e com taxa de acurácia superior àquela apresentada pelas mesmas técnicas aplicadas separadamente. Redes neuro-fuzzy ([BHATTACHARJEE et al., 2010](#)), Wavelet/SVM ([GUMUS et al., 2010](#)) e LDA/NN ([ELEYAN; DEMIREL, 2006](#)) são exemplos de sistemas de reconhecimento facial híbridos.

A Tabela 5 apresenta a comparação entre métodos de reconhecimento facial em suas versões clássicas e híbridas. Os experimentos foram realizados com o banco de imagens *ORL Database fo faces*.

Tabela 5 – Comparação de métodos híbridos

Métodos	Híbrido (%)	Tradicional (%)
Fuzzy MLP	97.75	81.25
LDA-NN	97.00	94.40
PCA-NN	95.00	91.25
Wavelet-SVM	<b>98.10</b>	<b>97.00</b>

Fonte: Autoria própria

O método proposto por [Bhattacharjee et al. \(2010\)](#) utiliza Gabor Wavelets ([HUANG; ZHANG, 2007](#)) para extrair características das imagens de face e classificá-las com uma rede neural fuzzy MLP ([KELLER; HUNT, 1985](#)). Esta rede neural foi treinada com o algoritmo *back propagation* com função sigmoide, contendo 4 camadas, uma de entrada, duas ocultas e uma de saída, com 48, 15, 5 e 1 neurônios, respectivamente. Foi utilizada uma rede neural para cada classe (indivíduo), desse modo, cada rede é especializada em reconhecer um indivíduo. A versão tradicional deste método não utiliza lógica fuzzy.

O trabalho de ([ELEYAN; DEMIREL, 2006](#)) apresenta um estudo comparativo entre dois métodos clássicos, PCA e LDA, e dois métodos híbridos, PCA-NN e LDA-NN, de reconhecimento facial. Ambos os métodos híbridos possuem duas fases: uma fase de extração de características, que utiliza PCA ou LDA e a uma fase de classificação, que utiliza redes neurais. As componentes principais obtidas na fase de extração de

características foi utilizada como entrada para classificação pela rede neural, que foi desenvolvida com a arquitetura *Feedforward* e treinada com o algoritmo *back propagation*. As versões clássicas dos algoritmos PCA e LDA utilizam a distância euclidiana para determinar a similaridade entre as imagens de teste e a representação de cada indivíduo.

A proposta de Gumus et al. (2010) é uma comparação entre métodos de reconhecimento facial que utilizam a transformada discreta Wavelet. Ambos os métodos, tradicional e híbrido, utilizam a transformada Wavelet para extrair características das imagens de face. Porém, na fase de classificação, o método tradicional utiliza a distância euclidiana como critério para determinar a similaridade. Já o método híbrido utiliza o algoritmo SVM para classificar os indivíduos.

Nos experimentos apresentados na Tabela 5, observa-se que os métodos híbridos possuem maior taxa de acurácia que suas versões clássicas. A principal diferença entre os métodos está na fase de reconhecimento, onde as métricas de similaridade das versões tradicionais foram substituídas por classificadores nas versões híbridas, o que tornou o método mais eficiente e robusto.

### 3.3 Conclusão

Neste capítulo é apresentado um estudo comparativo de métodos de reconhecimento facial de diferentes abordagens. O estudo comparativo apresentado indica diversas diferenças entre os métodos avaliados, analisando as vantagens e desvantagens no uso de cada abordagem, tendo como exemplo, os experimentos mostrados.

Cada abordagem utiliza uma perspectiva do problema de reconhecimento facial. A abordagem holística apresenta robustez ao reconhecer faces de diferentes ângulos e expressões faciais, apresentando baixa variação de acurácia mesmo em casos de reconhecimento com faces parcialmente obstruídas, quando comparado aos métodos geométricos. Os métodos da abordagem geométrica apresentam tolerância a variações de iluminação, embora sejam sensíveis a variações de pose e ângulo da face na imagem. Os métodos híbridos podem ser criados para unir métodos holísticos e geométricos de forma a combinar as vantagens de ambos os métodos e reduzir suas desvantagens.

Apesar da existência de diversos métodos de reconhecimento facial automático, a escolha de um método adequado, assim como de seus parâmetros, depende do foco do problema, podendo ser realizada com base no resultado de experimentos com dados que reflitam o domínio da aplicação.

Com base em tal estudo, chegou-se a hipótese de que o uso das técnicas PCA e LDA poderiam apresentar ganhos na área.

## 4 PROPOSTA DE SOLUÇÃO

Este trabalho apresenta o desenvolvimento e avaliação de métodos híbridos de reconhecimento facial baseados na extração de características realizada pelas técnicas PCA e LDA. Neste trabalho as técnicas PCA e LDA são aplicadas às imagens de faces para extração de características de modo a minimizar sua dimensionalidade e maximizar sua capacidade de discriminação. Em seguida, as características são utilizadas em métricas de similaridade: distância euclidiana, distância Cosseno, distância de Mahalanobis e distância de Manhattan; e nos classificadores:  $k$ -NN, SVM e redes neurais MLP para realizar a classificação de faces.

Para a realização deste trabalho, foi feito um estudo comparativo de métodos de reconhecimento facial com base na revisão bibliográfica, com o objetivo de avaliar o desempenho de métodos de diferentes abordagens. Com base nesse estudo, chegou-se a hipótese de que o uso das técnicas PCA e LDA poderiam ser combinadas para compor um extrator de características que poderia apresentar avanços na área.

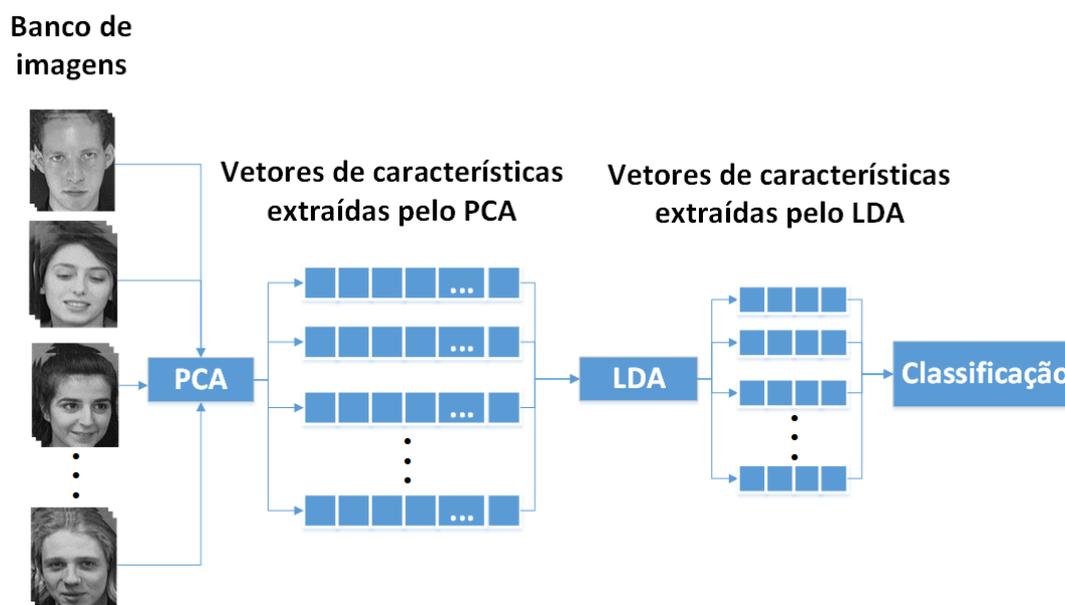
A técnica LDA foi selecionada com base em seu desempenho avaliado em diversos trabalhos, nos quais apresenta desempenho equivalente ou superior aos métodos avaliados com diferentes bancos de imagens (KHARE; NAIK, 2014), variações nas condições de iluminação (LIU; ZHOU; JIN, 2010) e expressões faciais (ABIDIN; HARJOKO, 2012). No entanto, esta técnica possui elevado nível de processamento devido a dimensionalidade das imagens. Para reduzir esse processamento, foi aplicada a técnica PCA para reduzir a dimensionalidade do conjunto de imagens original.

Estas técnicas foram estudadas e avaliadas para compor um método híbrido para extração de características utilizando as técnicas PCA e LDA (SILVA; CINTRA, 2015). Diferente das diversas propostas estudadas, neste trabalho foram realizados experimentos para selecionar apenas o número de características necessárias para descrever as imagens de face, de maneira a utilizar o número mínimo de atributos para obter o máximo de acurácia e reduzir o custo computacional do processo.

O princípio é que quanto maior a concentração de características nas componentes, mais fiel é a representação das faces, e quanto menor for o número de componentes selecionadas, menor o vetor de características. Assim, o ideal é selecionar o menor número de componentes que contenha a maior concentração de características.

Após o processo de extração de características, a classificação das faces foi avaliada de modo experimental por métricas de similaridade e classificadores. A Figura 24 apresenta o modelo completo da proposta.

Figura 24 – Modelo proposto para reconhecimento facial



Fonte: Autoria própria

No modelo apresentado na Figura 24, a partir das faces do banco de imagens, são extraídas características para formar o espaço de entrada para classificadores e métricas de similaridade. A seção 4.1 descreve o processo utilizado para extração de características.

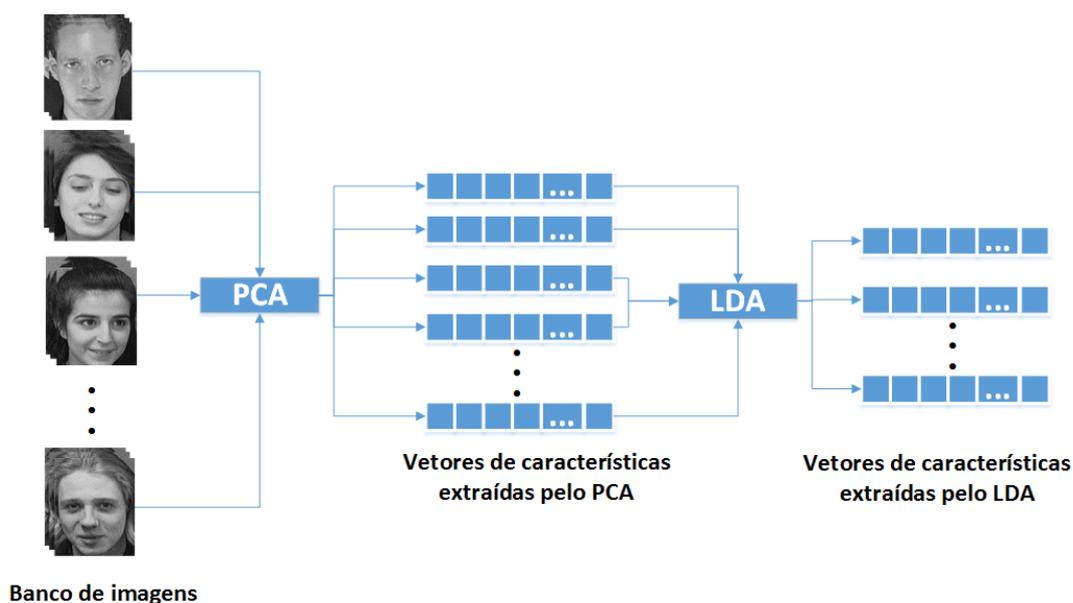
#### 4.1 Extração de características

Esta seção descreve a metodologia utilizada para extrair características de imagens de face, para posteriormente serem classificadas. Este módulo de extração de características utiliza a técnica PCA para reduzir o número de atributos das imagens de face, aumentando o desempenho da técnica LDA na extração de características.

A técnica LDA, a partir de um conjunto de imagens de treinamento, gera um conjunto de vetores característicos que representa a imagem no espaço de características. Considerando um conjunto de imagens com 400 imagens de face, cada imagem com 112 por 92 *pixels*, o vetor gerado por uma imagem de face possui 10304 dimensões. Calcular os vetores de características para instâncias dessa dimensionalidade demanda um alto processamento, que pode ser reduzido ao reduzir a dimensionalidade das imagens.

Neste trabalho, a solução proposta para esse problema é utilizar a técnica PCA para reduzir a dimensionalidade do conjunto de imagens. A Figura 25 apresenta o modelo proposto para a extração de características.

Figura 25 – Modelo proposto para extração de características



Fonte: Autoria própria

Como ilustrado na Figura 25, a técnica PCA foi utilizada para reduzir as dimensões das imagens do banco, gerando um conjunto de vetores característicos de menor dimensionalidade que as imagens originais, facilitando o processamento a ser realizado no passo seguinte, pela LDA. A partir dos vetores de características obtido pela PCA, a LDA gera um novo conjunto de vetores de características, com um número ainda menor de dimensões. Dessa forma, combinando a capacidade de redução de dimensionalidade da PCA e o poder de discriminação de dados da LDA, foi composto um método híbrido de extração de características.

Os bancos de imagens ORL e JAFFE foram selecionados com base nas características apresentadas no Quadro 3, tais como variações de iluminação e expressões faciais.

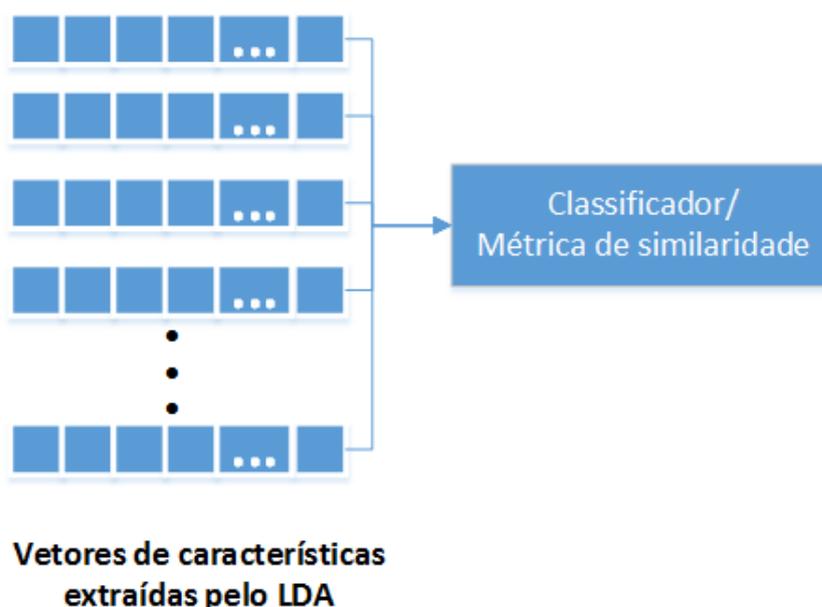
Com base no resultados dos experimentos descritos no Capítulo 5, para o banco de imagens ORL, a técnica PCA foi aplicada para extrair vetores de características de 30 dimensões. A partir dos vetores de características extraídas pela PCA, a LDA extrai novos vetores de características com 19 dimensões. Do mesmo modo, para o banco de imagens JAFFE, a PCA extrai vetores de características de 14 dimensões. Em seguida, a LDA extrai novos vetores de características com 10 dimensões.

Ao final do processo de extração de características, é gerado um número de vetores de características correspondente ao número de imagens do conjunto de treinamento, porém com o número de dimensões menor que o do conjunto original.

## 4.2 Reconhecimento de faces

A Figura 26 ilustra o modelo aplicado na classificação das características faciais.

Figura 26 – Modelo proposto para classificação de características



Fonte: Autoria própria

Na Figura 26, os vetores de características obtidos pelo LDA, foram utilizados como dados de entrada para os classificadores e métricas de similaridade.

O reconhecimento de faces por métricas de similaridade foi realizado pelo cálculo de uma distância mínima entre as projeções das imagens de treinamento e teste no espaço de características. A identidade de uma imagem de teste é estimada como a identidade da imagem do conjunto de treinamento, cuja projeção estiver mais próximo da projeção da imagem de teste. Foram utilizadas as métricas de similaridade: distância cosseno, distância euclidiana, distância de Mahalanobis e distância de Manhattan.

O reconhecimento facial com o classificador  $k$ -NN foi realizado testando 3 valores para  $k$ : 3, 5, 7. A métrica utilizada pelo  $k$ -NN foi a distância euclidiana.

O reconhecimento facial por redes neurais foi realizado pelo treinamento de uma rede neural com as características extraídas do conjunto de treinamento pelo LDA. A rede neural foi treinada utilizando o algoritmo de *backpropagation* com arquitetura composta por 19 neurônios na camada de entrada, correspondendo ao tamanho do vetor de características, 68 neurônios na camada oculta, e 40 neurônios na camada de saída, sendo tais valores definidos de forma empírica. Este rede foi configurada com *momentum* de 0.2, taxa de aprendizado igual 0.3 e 500 épocas de treinamento.

Ao final do processo de classificação, a métrica ou classificador identifica os indivíduos com base base características extraídas das imagens de face.

### 4.3 Validação da proposta

Para validação da proposta, foi desenvolvido um sistema de testes utilizando o software Matlab ([MATLAB, 2013](#)), no qual foram implementados os algoritmos PCA, LDA e as métricas de similaridade. Para avaliar o desempenho dos classificadores  $k$ -NN e da rede neural foi utilizado o software Weka ([HALL et al., 2009](#)). Os experimentos realizados estão no capítulo 5. No Anexo A estão códigos desenvolvidas para a realização dos experimentos.

## 5 EXPERIMENTOS

Neste capítulo são descritos os experimentos realizados para avaliar o desempenho do método PCA/LDA proposto neste trabalho com métricas de similaridade e classificadores em bancos de imagens de face com diferentes características. Os experimentos estão organizados por métodos. Cada seção apresenta a descrição da metodologia aplicada na realização do experimento e os resultados obtidos. Nas tabelas estão em negrito os maiores valores por método e hachurados o maior valor por experimento.

Os bancos de imagens de face selecionados para os experimentos foram: *ORL Database of faces* e *JAFFE*. Estes bancos foram selecionados por apresentarem características comuns a diversos bancos de imagens de face, tais como faces com diferentes iluminações, expressões faciais e poses. O banco de imagens *JAFFE* possui 213 imagens de 10 modelos japonesas. Como este banco não apresenta um número uniforme de imagens por indivíduo, foram selecionadas 13 imagens para serem removidas do conjunto utilizado nos experimentos, para que todos os indivíduos tenham o mesmo número de imagens. Assim, o banco de imagens *JAFFE* foi utilizado nos testes com 200 imagens, sendo exatamente 20 imagens para cada indivíduo.

### 5.1 Experimento para determinar a dimensionalidade dos vetores de características

Neste experimento são descritos os testes realizados para definir o número ótimo de dimensões dos vetores de características. Os vetores de características utilizados na proposta deste trabalho são extraídos, inicialmente, pela técnica PCA, que reduz a dimensionalidade das imagens, extraíndo vetores de características. Em seguida, a técnica LDA, a partir dos vetores extraídos pela PCA, extrai novas características que discriminam os indivíduos dos bancos de imagens *ORL* e *JAFFE*.

Nas propostas de [Turk e Pentland \(1991\)](#), [Sirovich e Kirby \(1987\)](#), [Yang, Zhang David an Frangi e Yang \(2014\)](#), [Delac, Grgic e Grgic \(2005\)](#), entre outras, o número de componentes utilizados para criar o vetor de características foi determinado de maneira empírica, nas quais foram testados diversos valores e destacado aquele que apresentou melhor desempenho. Nestas propostas o número de componentes utilizado foi cerca de 40% do número de imagens do conjunto de treinamento.

Neste trabalho é aplicada uma estratégia baseada no estudo das técnicas PCA e LDA, na qual são selecionados o número mínimo de componentes que representam

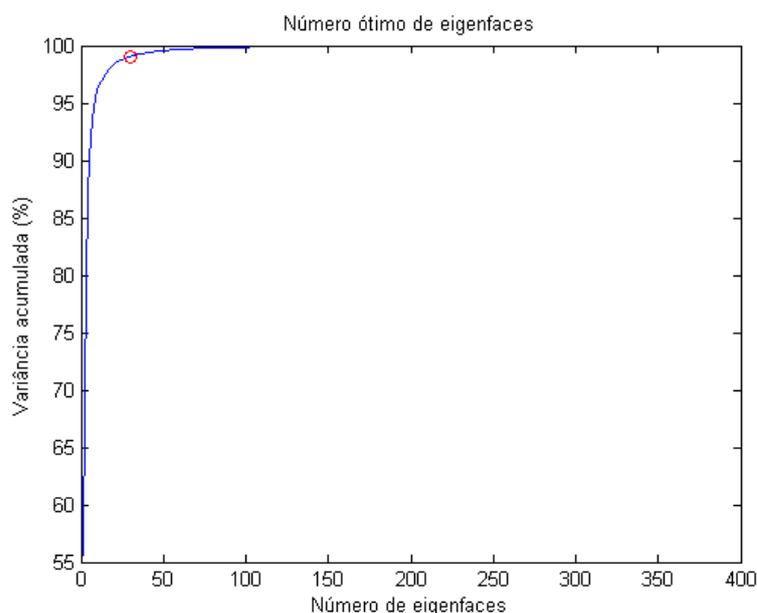
as principais características de cada indivíduo. Dessa forma, são gerados vetores de características com um número mínimo de dimensões que representa com máxima fidelidade os padrões faciais.

A face humana apresenta um padrão de características comuns presentes na maioria das pessoas, tais como olhos, boca e nariz. Essas características formam um padrão facial. Cada pessoa apresenta características únicas que representam variações desse padrão. Para a abordagem holística, essas características presentes nas imagens de face são as variações de aparência causadas por mudanças de iluminação, expressões faciais e poses.

As técnicas PCA e LDA representam essa variabilidade da aparência em um conjunto de componentes, as *eigenfaces* e *fisherfaces*, respectivamente. Cada componente concentra uma quantidade de variações, indicado por um valor associado a cada componente. Essa quantidade é medida pela variância, uma medida estatística que indica a variabilidade de um conjunto de dados. Quanto maior o valor associado, mais características sua componente possui, logo melhor ela representa as faces do conjunto de imagens. Ao ordenar as componentes de acordo com sua variância, é possível selecionar as componentes que melhor representem o conjunto de imagens e descartar as demais.

Neste trabalho, as técnicas PCA e LDA foram utilizadas para extrair características dos bancos de imagens ORL e JAFFE. Para o banco de imagens ORL, a PCA foi utilizada para extrair 400 componentes, uma para cada imagem do banco, das quais foram selecionadas 30 componentes, que representam 99% das características presentes no banco. Este valor foi selecionado com base no gráfico da Figura 27.

Figura 27 – Variância acumulada das componentes da PCA no banco de imagens ORL

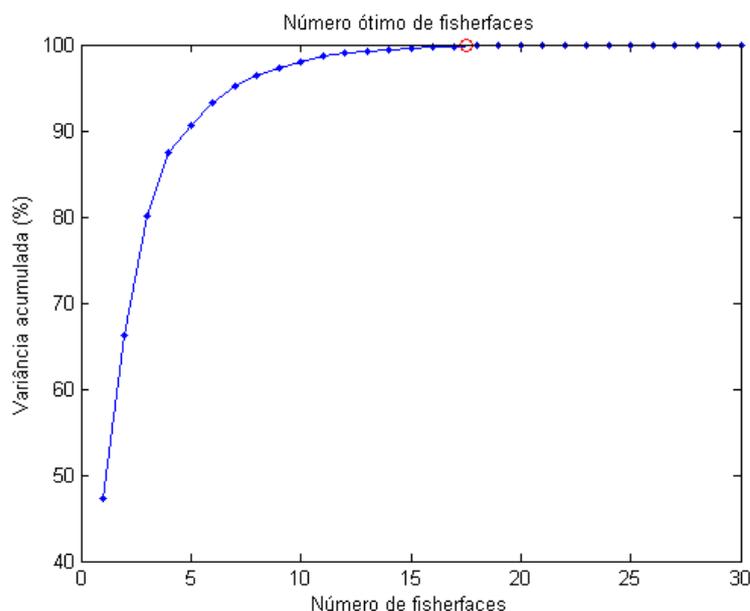


Fonte: Autoria própria

No gráfico 27, está traçada a relação entre a variância acumulada das *eigenfaces* e o número *eigenfaces* geradas pela PCA. A variância acumulada, no eixo vertical, é uma medida que expressa o percentual de variações do banco de imagens presentes nas *eigenfaces*. O número de componentes, no eixo horizontal, indica o número total de *eigenfaces* geradas pela PCA para o banco ORL. Para representar todas as características do banco são necessárias cerca de 110 *eigenfaces*, porém o ponto marcado no alto do gráfico, indica que com 30 componentes é possível representar 99% das características do banco de imagens ORL. O mesmo procedimento foi realizado com banco e imagens JAFFE, sendo utilizadas 14 componentes para representar 99% das características contidas nas imagens. Isto indica que os vetores de características gerados pela PCA possuem 30 dimensões, para o banco ORL, e 14 para o banco JAFFE.

Em seguida, os vetores de características da PCA são utilizados como dados de entrada para a técnica LDA. A Figura 28 apresenta o gráfico de variância acumulada para as componentes geradas pela LDA para o banco de imagens ORL.

Figura 28 – Variação acumulada das componentes da LDA no banco de imagens ORL



Fonte: Autoria própria

Na Figura 28, o gráfico traça a relação entre variância acumulada e as componentes geradas pelo LDA. O valor assinalado no alto do gráfico indica que 19 componentes (*fisherfaces*) representam 100% da variância do conjunto de características. Isto significa que com 19 componentes é possível representar 100% das características analisadas. O mesmo procedimento foi realizado para o banco de imagens JAFFE, selecionando-se 10 componentes para representar 100% das características das imagens de face.

Este experimento demonstra que com o estudo dos métodos e os testes realizados é possível obter uma representação do conjunto de imagens de faces que apresente

baixa dimensionalidade, em comparação a dimensionalidade do conjunto original de imagens. Os experimentos a seguir avaliam o desempenho das características em relação ao reconhecimento de faces.

## 5.2 Experimentos para avaliação de métricas de similaridade

Este experimento consiste em avaliar o desempenho das técnicas PCA/LDA utilizando 4 métricas de similaridade. Neste experimento foi utilizada a técnica *10-fold cross-validation*, que divide o conjunto de dados em 10 subconjuntos de tamanho igual, seleciona um subconjunto para teste e os demais para treinamento. Este processo é repetido 10 vezes, no qual a cada iteração um novo subconjunto é selecionado para teste e os demais são utilizados para treinamento do algoritmo, até que todos os subconjuntos tenham sido utilizados para teste.

No PCA/LDA foram utilizadas 30 dimensões para o PCA e 19 para o LDA. Para comparação de desempenho foi utilizada a técnica PCA com 30 componentes para construção do espaço de características. As técnicas foram avaliadas com as métricas: distância euclidiana, distância Cosseno, distância de Mahalanobis e distância de Manhattan.

A Tabela 6 apresenta os resultados dos experimentos com os bancos de imagens ORL e JAFFE. Para avaliar o desempenho dos métodos foram calculados o percentual da taxa média de acurácia e seu respectivo desvio padrão. Os valores em destaque apresentam as maiores taxas de acurácia obtidas pelas métricas em cada método.

Tabela 6 – Experimentos para avaliação de métricas de similaridade com o banco de imagens ORL

Taxa de acurácia (%)				
Bancos de imagens	ORL		JAFFE	
Métrica	PCA	PCA/LDA	PCA	PCA/LDA
Euclidiana	96.50 ± 2.10	99.00 ± 1.74	94.00 ± 6.99	96.50 ± 5.29
Cosseno	96.00 ± 2.10	<b>99.25 ± 1.20</b>	94.50 ± 5.98	96.50 ± 5.57
Mahalanobis	96.00 ± 2.10	<b>99.25 ± 1.20</b>	94.50 ± 5.98	<b>97.00 ± 5.37</b>
Manhattan	96.50 ± 2.10	90.00 ± 1.74	94.50 ± 5.98	<b>97.00 ± 5.29</b>
<b>Média</b>	96.25 ± 2.10	96.88 ± 1.47	94.38 ± 6.23	96.75 ± 5.38

Fonte: Autoria própria

Nos testes com o banco de imagens ORL, apresentados na Tabela 6, o método PCA obteve a maior taxa média de acurácia com as métricas euclidiana e Manhattan, com o valor de 96.50% e desvio padrão igual a 2.10%. O método PCA/LDA obteve a maior taxa média de acurácia com as métricas Cosseno e Mahalanobis, com o valor de 99.25% e desvio padrão igual a 1.20%.

Nos experimentos com o banco de imagens JAFFE, apresentados na Tabela 6, o método PCA obteve a maior taxa média de acurácia com as métricas distância Cosseno, distância de Mahalanobis e distância de Manhattan, com o valor de 94.50% e desvio padrão igual a 5.98. O método PCA/LDA obteve a maior taxa média de acurácia com as métricas Mahalanobis, com o valor de 97.00% e desvio padrão de 5.37, e Manhattan, com o valor de 97.00% e desvio padrão igual a 5.29.

### 5.3 Experimentos para avaliação do método $k$ -NN

Este experimento foi realizado com o objetivo de avaliar o desempenho do método  $k$ -NN na classificação de características faciais. Para extrair as características do banco de imagens ORL, foram implementado o método PCA/LDA *software* Matlab. As características extraídas foram organizadas em um banco de características no formato ARFF para ser utilizado no *software* Weka. Este banco de características foi utilizado no Weka, pelo algoritmo IBk, que implementa o classificador  $k$ -NN, com a técnica *10-fold cross-validation*. Para o  $k$ -NN, foram utilizados os valores  $k=3$ ,  $k=5$  e  $k=7$ .

A Tabela 7 apresenta os resultados dos experimentos com o classificador  $k$ -NN.

Tabela 7 – Experimentos para avaliação do método  $k$ -NN no banco de imagens ORL

Taxa de acurácia (%)		
$k$	ORL	JAFFE
3	98.75	98.50
5	98.75	98.50
7	98.75	98.50

Fonte: Autoria própria

Os resultados apresentados na Tabela 7 mostram que para os valores de  $k$  escolhidos, a taxa de acurácia permanece em 98.75%.

### 5.4 Experimentos com Máquina de Vetores de Suporte

Neste experimento foi utilizado o método SVM para classificação das características extraídas pelo método PCA/LDA proposto neste trabalho. Para a realização deste experimento foi desenvolvido o método PCA/LDA no *software* Matlab para a extração das características dos bancos de imagens ORL e JAFFE. As características extraídas de cada banco foram organizadas, separadamente, em bancos de características no formato ARFF para serem utilizadas pelo *software* Weka. O algoritmo SMO, implementado pela

biblioteca LIBSVM, foi utilizado no Weka para treinamento do SVM. O SVM foi testado com 3 funções de *kernel*: PolyKernel, *Normalized* PolyKernel e RBFKernel. Para as funções PolyKernel e *Normalized* foi definido o parâmetro *Normalize training data*, e para função RBFKernel foi utilizado o parâmetro *Standartize training data*.

A Tabela 8 apresenta os resultados dos experimentos com o método SVM.

Tabela 8 – Experimento com máquina de vetores de suporte

Taxa de acurácia (%)		
kernel	ORL	JAFFE
PolyKernel	98.25	98.50
RBFKernel	95.75	98.00
<i>Normalized</i> PolyKernel	97.75	96.50

Fonte: Autoria própria

Para o banco de imagens ORL e JAFFE, os resultados apresentados na Tabela 8 mostram que a maior taxa de acurácia foi obtida com a função PolyKernel, com a taxa de 98.25% para o banco ORL, e a taxa de 98.50% para o banco de imagens JAFFE.

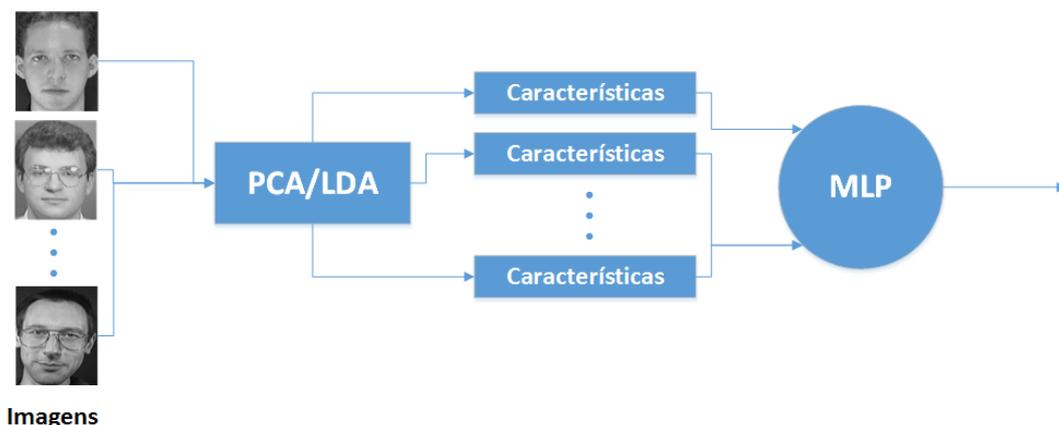
### 5.5 Experimento para avaliação de redes neurais

Este experimento foi realizado com o objetivo de desenvolver uma rede neural MLP e avaliar seu desempenho na classificação de características faciais extraídas pelo método PCA/LDA proposto neste trabalho. Para a realização deste experimento foi desenvolvido o método PCA/LDA no software Matlab para a extração das características dos bancos de imagens ORL e JAFFE. As características extraídas de cada banco foram organizadas, separadamente, em bancos de características no formato ARFF para serem utilizadas pelo *software* Weka. A rede neural MLP foi modelada utilizando a função *MultilayerPerceptron* do Weka. O modelo do método é ilustrado na Figura 29.

A Figura 29 apresenta o modelo de como o método realiza o reconhecimento facial. Para cada banco, as imagens de face são utilizadas como dados de entrada para o método PCA/LDA, que extrai as características das imagens, gerando vetores de padrões faciais. Esses vetores são os dados de entrada para a rede neural MLP, que classifica-os de acordo com as classes (indivíduos) do banco de imagens.

A classificação das imagens do banco ORL foi realizada por uma rede neural MLP com 19 neurônios na camada de entrada, que correspondem ao tamanho do vetor de características gerado pelo PCA/LDA, 68 neurônios na camada oculta e 40 neurônios na camada de saída, que correspondem as 40 classes (indivíduos) do banco de imagens. Para o treinamento da rede foi utilizado o algoritmo *backpropagation* com função sigmoide, taxa de aprendizado de 0.5 e *momentum* de 0.2. Para o banco de

Figura 29 – Classificação de características faciais por uma rede neural MLP



Fonte: Autoria própria

imagens JAFFE foi desenvolvida uma rede neural MLP com 10 neurônios na camada de entrada, correspondendo ao tamanho do vetor de características gerado pelo PCA/LDA, 19 neurônios na camada oculta, 10 neurônios na camada de saída, correspondendo as 10 classes presentes banco de imagens. Para o treinamento da rede foi utilizado o algoritmo *backpropagation* com função sigmoide, taxa de aprendizado de 0.3 e *momentum* de 0.2.

A Tabela 9 apresenta os resultados dos experimentos com redes neurais.

Tabela 9 – Resultado comparativo da distância Cosseno com rede neural MLP

Taxa de acurácia (%)		
RNA	ORL	JAFFE
MLP	99.00	97.50

Fonte: Autoria própria

Neste experimento, o método PCA/LDA obteve taxa de acurácia 99.00% para o banco de imagens ORL e 97.50% para o banco de imagens JAFFE.

### 5.6 Análise dos experimentos

A seguir é apresentada a comparação entre os resultados obtidos neste trabalho e os resultados dos demais trabalhos encontrados na literatura. A Tabela 10 apresenta resultados obtidos em todos os experimentos realizados neste trabalho.

Na Tabela 10 estão em destaque as maiores taxas de acurácia obtidas nos experimentos em cada banco de imagens. Para o banco de imagens ORL, as métricas Cosseno e Mahalanobis apresentam o melhor desempenho em relação as métricas e classificadores avaliados, com taxa de acurácia igual a 99.25%. O *k*-NN obteve a mesma acurácia, 98.75%, para os 3 valores de *k* selecionados, 3, 5 e 7. O SVM apresenta a taxa de

Tabela 10 – Experimentos com métricas e classificadores

Taxa de acurácia (%)		
Bancos de imagens	ORL	JAFFE
<b>Métricas</b>		
Euclidiana	99.00	96.50
Cosseno	<b>99.25</b>	96.50
Mahalanobis	<b>99.25</b>	97.00
Manhattan	99.00	97.00
<b>k-NN</b>		
3	98.75	<b>98.50</b>
5	98.75	<b>98.50</b>
7	98.75	<b>98.50</b>
<b>SVM</b>		
PolyKernel	98.25	<b>98.50</b>
RBFKernel	95.75	98.00
Normalized Polykernel	97.75	96.50
<b>Rede Neural</b>		
MLP	99.00	97.50

Fonte: Autoria própria

acurácia de 98.25% com a função *PolyKernel*. A rede neural MLP obteve taxa de acurácia de 99.00% na classificação das características de face.

Para o banco de imagens JAFFE, o melhor desempenho foi obtido com os classificadores *k-NN* e SVM com a função *PolyKernel*. Ambos os classificadores obtiveram 98.50% de acurácia. A rede neural MLP apresenta taxa de acurácia de 97.50% na classificação das características. Entre as métricas de similaridade, a métrica que apresenta melhor desempenho é a distância Cosseno, com taxa de acurácia de 97.00%.

As Tabelas 11 e 12 apresentam a comparação entre os métodos propostos neste trabalho e demais métodos encontrados na literatura. Para esta comparação foram selecionados os métodos que apresentam os melhores desempenhos nos experimentos apresentados neste trabalho. Os métodos PCA/LDA+Cosseno e PCA/LDA+Mahalanobis foram selecionados com base no desempenho apresentado nos experimentos com o banco de imagens ORL. Sob os mesmos critérios foram selecionados os métodos PCA/LDA+k-NN e PCA/LDA+SVM para o comparativo no banco de imagens JAFFE. Para que a comparação ocorra sob condições iguais, foram selecionados trabalhos na literatura cujos métodos tenha sido avaliados com a técnica 10-fold *cross-validation* e com os mesmos bancos de imagens utilizados nos experimentos.

A Tabela 11 apresenta comparação entre os métodos que apresentaram a maior taxa de acurácia nos experimentos realizados com o banco de imagens ORL e demais trabalhos encontrados na literatura. Com base nos critérios de seleção, foram selecionados os trabalhos Eleyan e Demirel (2006), Schwenker et al. (2006) e Thakur et al. (2008). O trabalho de Eleyan e Demirel (2006) utiliza a técnica LDA para extrair características das imagens de face do banco ORL e realiza a classificação com uma rede neural MLP. O método proposto por Schwenker et al. (2006) utiliza a técnica *Orientation Histogram*

(OH) para extração de características do banco de imagens ORL e realiza a classificação com o método  $k$ -NN. O trabalho de [Thakur et al. \(2008\)](#) utiliza a técnica PCA para extração de características do banco ORL e uma rede neural *Radial Base Function* (RBF) para classificação. A Tabela 11 apresenta a comparação dos resultados encontrados na literatura com os melhores resultados obtidos neste trabalho.

Tabela 11 – Experimentos com o banco de imagens ORL

Taxa de acurácia (%)	
Método	ORL
LDA/MLP	97.00
PCA/RBF	97.00
OH/ $k$ -NN	98.50
<b>PCA/LDA + Cosseno</b>	<b>99.25</b>
<b>PCA/LDA + Mahalanobis</b>	<b>99.25</b>

Fonte: Autoria própria

Os métodos PCA/LDA + Cosseno e PCA/LDA + Mahalanobis, propostos neste trabalho para o reconhecimento de faces do banco de imagens ORL, apresentam desempenho superior aos demais métodos encontrados na literatura, obtendo taxa de acurácia de 99.25%.

A Tabela 12 apresenta o comparativo entre os métodos que apresentaram a maior taxa de acurácia nos experimentos realizados com o banco de imagens JAFFE e demais trabalhos encontrados na literatura. Com base nos critérios de seleção, foram selecionados os trabalhos de [Qiu et al. \(2010\)](#), [Li e Yahya \(2014\)](#) e [Feng et al. \(2006\)](#). O trabalho de [Qiu et al. \(2010\)](#) utiliza as técnicas LBP, PCA e LDA para extrair características das imagens de face do banco JAFFE e realiza o reconhecimento com o classificador  $k$ -NN. O trabalho de [Li e Yahya \(2014\)](#) apresenta um método formado pelas técnicas Gabor Wavelet e SVD para extrair características e classificá-las com o método  $k$ -NN. O trabalho de [Feng et al. \(2006\)](#) utiliza a técnica AMM e LBP para extrair características das imagens de face e classificá-las utilizando o método  $k$ -NN.

Tabela 12 – Experimentos com o banco de imagens JAFFE

Taxa de acurácia (%)	
Método	JAFFE
LBP/PCA/LDA + $k$ -NN	90.00
Gabor Wavelet/SVD + $k$ -NN	87.50
AMM/LBP + $k$ -NN	83.00
<b>PCA/LDA + <math>k</math>-NN</b>	<b>98.50</b>
<b>PCA/LDA + SVM</b>	<b>98.50</b>

Fonte: Autoria própria

Os métodos PCA/LDA+ $k$ -NN e PCA/LDA+SVM, propostos neste trabalho para o reconhecimento de faces no banco de imagens JAFFE, apresentam desempenho superior aos demais métodos encontrados na literatura, obtendo taxa de acurácia de 98.50%.

## 6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Neste trabalho foi realizado um estudo sobre métodos de reconhecimento facial encontrados na literatura com o objetivo de desenvolver um método que apresentasse um desempenho superior aos demais estudados. Com base nesse estudo, foram selecionadas as técnicas de extração de características PCA e LDA, devido a sua capacidade de redução da dimensionalidade e extração de características relevantes que representam com fidelidade os padrões faciais dos indivíduos, para compor um método híbrido de extração de características de imagens de faces.

Para validação da proposta, foram utilizados os bancos de imagens de faces ORL e JAFFE, cujas características das imagens foram extraídas e comparadas pelas métricas de similaridade: distância Cosseno, distância euclidiana, distância de Mahalanobis e distância de Manhattan. Para classificação das características foram utilizados os classificadores  $k$ -NN, SVM e redes neurais artificiais. A análise dos resultados dos experimentos mostram que as métricas Cosseno e Mahalanobis obtiveram a maior taxa de acurácia, 99.25% nos experimentos realizados com o banco ORL, superando trabalhos encontrados na literatura. Com o banco de imagens JAFFE, os classificadores  $k$ -NN e SVM apresentaram o melhor desempenho com 98.50% de acurácia, apresentando desempenho superior aos métodos encontrados na literatura.

### 6.1 Trabalhos futuros

Como sugestões de trabalhos futuros, pretende-se ampliar os experimentos para mais bancos de imagens de face que apresentem características diferentes das encontradas nos bancos de imagens JAFFE e ORL. São sugeridos os bancos: *University of Essex Face Database*, *FEI Face Database* e *PICS*. Todos estes bancos estão descritos na Tabela 3.

Para algumas aplicações, é necessário o reconhecimento de faces parcialmente obstruídas. Poucos bancos de imagens apresentam essa característica, o que dificulta a validação de propostas que busquem solucionar este problema. Como sugestão para trabalhos futuros, propõe-se o estudo de métodos de segmentação facial para que, a partir de regiões faciais segmentadas, tais como olhos, nariz e boca, possa-se realizar o reconhecimento de faces mesmo o padrão facial não for completamente conhecido.

## 6.2 Publicações

Durante o desenvolvimento deste trabalho foram realizadas publicações e participações em eventos científicos que contribuíram para o desenvolvimento e aprimoramento de sua proposta.

### Publicados

- Estudo de métodos para reconhecimento de padrões faciais utilizando algoritmos genéticos e análise de componentes principais (SILVA; CINTRA, 2014).
- Reconhecimento de padrões faciais: Um estudo (SILVA; CINTRA, 2015)

### Submetidos

- Proposta de método híbrido para reconhecimento de faces (RITA - Revista de Informática Teórica e Aplicada)

## REFERÊNCIAS

- ABIDIN, Z.; HARJOKO, A. A neural network based facial expression recognition using fisherface. *International Journal of Computer Applications*, v. 59, n. 3, p. 30–34, Dezembro 2012. Citado na página 57.
- BARTLETT, M. S.; MOVELLAN, J. R.; SEJNOWSKI, T. J. Face recognition by independent component analysis. *IEEE Transactions On Neural Networks*, v. 13, n. 06, p. 1450–1464, Novembro 2002. Citado na página 50.
- BELLHUMER, P. N.; HESPANHA, J.; KRIEGMAN, D. Eigenfaces vs fisherfaces - recognition using class specific linear projection. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence, Special Issue on Face Recognition*. University of Yale, Yale: [s.n.], 1997. v. 19, n. 7, p. 711–720. Citado 3 vezes nas páginas 32, 46 e 48.
- BERG, T.; BELHUMEUR, P. N. Tom-vs-pete classifiers and identity-preserving alignment for face verification. *International Conference on Computer Vision*, p. 1–11, 2012. Citado na página 54.
- BHAT, F.; WANI, M. A. Elastic bunch graph matching based face recognition under varying lighting, pose and expression conditions. *Face Recognition Using Gabor Wavelet*, v. 19, n. 7, p. 775–779, Julho 2015. Citado 2 vezes nas páginas 54 e 55.
- BHATTACHARJEE, D. et al. Human face recognition using fuzzy multilayer perceptron. *Soft Computing*, 2010. Citado na página 55.
- BHATTACHARYYA, D. et al. Biometric authentication: A review. *International Journal of u- and e-Service, Science and Technology*, v. 2, n. 3, p. 13–28, Setembro 2009. Citado 2 vezes nas páginas 15 e 20.
- BLACK, J. J. A. et al. *FacePix Database*. [S.l.], 2011. Citado na página 46.
- CAO, X. et al. A practical transfer learning algorithm for face verification. In: IEEE. *IEEE International Conference on Computer Vision*. [S.l.], 2013. p. 3208–3215. Citado na página 54.
- CHEN, D. et al. Bayesian face revisited: A joint formulation. In: FITZGIBBON, A. W. et al. (Ed.). *Proceedings of the 12th European Conference on Computer Vision*. Florença, Itália: [s.n.], 2012. (ECCV '12, v. 3), p. 566–579. Citado na página 54.
- CHEN, D. et al. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In: *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*. Washington, DC, EUA: IEEE, 2013. (CVPR '13), p. 3025–3032. Citado na página 54.
- DELAC, K.; GRGIC, M.; GRGIC, S. Independent comparative study of PCA, ICA, and LDA on the FERET data set. *Int. J. Imaging Systems and Technology*, v. 15, n. 5, p. 252–260, 2005. Citado 4 vezes nas páginas 50, 51, 52 e 62.
- DEVIJVER, P. A.; KITTLER, J. *Pattern Recognition - A Statistical Approach*. 1. ed. Londres, UK: Prentice Hall, 1982. Citado na página 23.

- DINIZ, F. A. et al. Redface - um sistema de reconhecimento facial baseado em técnicas de análise de componentes principais e autofaces: comparação com diferentes classificadores. *Revista Brasileira de Computação Aplicada*, 2013. Citado na página 22.
- ELEYAN, A.; DEMIREL, H. PCA and LDA based face recognition using feedforward neural network classifier. In: *Proceedings of the 2006 International Conference on Multimedia Content Representation, Classification and Security*. Berlin, Heidelberg: Springer-Verlag, 2006. (MRCS'06), p. 199–206. Citado 2 vezes nas páginas 55 e 69.
- FACELI, K. et al. *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. 1. ed. Rio de Janeiro, RJ: LTC, 2011. Citado 2 vezes nas páginas 16 e 40.
- FENG, X. et al. A novel feature method for facial expression recognition. In: *Joint Conference On Information Science*. Kaohsiung, Taiwan: [s.n.], 2006. Citado na página 70.
- GEORGHIADES, A.; BELHUMEUR, P.; KRIEGMAN, D. J. From few to many: Illumination cone models for face recognition under variable lighting and poses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 23, n. 6, p. 643–660, 2001. Citado 2 vezes nas páginas 46 e 49.
- GIRISHA, H.; SREEPATHI, B.; KARIBASAPPA, K. Multi-view face recognition using local binary pattern. *International Journal of computer Science and Information Technology*, v. 5, n. 3, p. 2978–2981, 2014. Citado na página 54.
- GRAHAM, D. B.; ALLINSON, N. M. Characterizing Virtual Eigensignatures for General Purpose Face Recognition. In: WECHSLER, H. et al. (Ed.). *Face Recognition: From Theory to Applications*. The University of Sheffield, Western Bank, Sheffield: [s.n.], 1998. (NATO ASI Series F, v. 163), p. 446–456. Citado 3 vezes nas páginas 46, 48 e 52.
- GUMUS, E. et al. Evaluation of face recognition techniques using PCA, Wavelets and SVM. In: *Expert Systems with Applications*. [S.l.: s.n.], 2010. v. 37, n. 9, p. 6404–6408. Citado 2 vezes nas páginas 55 e 56.
- HALL, M. et al. The WEKA data mining software: An update. *SIGKDD Explorations*, v. 11, n. 1, p. 10–18, 2009. Citado na página 61.
- HIDAYAT, E. et al. A comparative study of feature extraction using PCA and LDA for face recognition. *International Conference on Information Assurance and Security (AIS)*, 2011. Citado na página 32.
- HUANG, G. B. et al. *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*. [S.l.], 2007. Citado 3 vezes nas páginas 46, 47 e 54.
- HUANG, L.; ZHANG, Z. Face recognition using multiscale gabor wavelet. In: *IEEE. Communications, Computers and Signal Processing*. [S.l.], 2007. (PACRIM '07), p. 280–283. Citado na página 55.
- IANCU, C.; CORCORAN, P. M. *Reviews, Refinements and New Ideas in Face Recognition*. [S.l.]: InTech, 2011. Citado 2 vezes nas páginas 54 e 55.
- JAFRI, R.; ARABNIA, H. R. A survey of face recognition techniques. *Journal of Information Processing Systems (JIPS)*, v. 5, n. 2, p. 41–68, 2009. Citado 2 vezes nas páginas 23 e 54.

- JAIN, A. K.; BOLLE, R.; PANKANTI, S. *Biometrics: Personal Identification in Networked Security*. 1. ed. Norwell, MA, EUA: Kluwer Academic Publishers, 1998. Citado na página 18.
- JAIN, A. K.; FLYNN, P.; ROSS, A. A. *Handbook of Biometrics*. 1. ed. Secaucus, NJ, EUA: Springer US, 2007. Citado 2 vezes nas páginas 18 e 19.
- JAIN, V.; MUKHERJEE, A. *The Indian face database*. 2002. Citado na página 47.
- KELLER, J. M.; HUNT, D. J. Incorporating fuzzy membership functions into the perceptron algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 7, n. 6, p. 693–699, 1985. Citado na página 55.
- KEPENECKI, B. *Face Recognition Using Gabor Wavelet Transform*. Dissertação (Mestrado) — The Graduate School Of Natural Sciences of The Middle East Technical University, 2001. Citado na página 54.
- KHARE, S. S.; NAIK, A. K. Comparative study of popular statistical methods for facial recognition. *IJISSET - International Journal of Innovative Science, Engineering & Technology*, v. 1, n. 5, p. 429–436, Julho 2014. Citado 3 vezes nas páginas 46, 52 e 57.
- LI, L. S.; YAHYA, N. Face recognition technique using gabor wavelet and singular value decomposition. In: IEEE. *Control System, Computing and Engineering*. [S.l.], 2014. p. 455–459. Citado na página 70.
- LINDEN, R. Técnicas de agrupamento. *Revista de sistemas de informação da FSMA Faculdade Salesiana Maria Auxiliadora*, n. 4, p. 18–36, 2009. Citado na página 39.
- LIU, Z.; ZHOU, J.; JIN, Z. Face recognition based on illumination adaptative LDA. In: *Pattern Recognition (ICPR)*. Istambul: [s.n.], 2010. p. 894–897. Citado na página 57.
- LYONS, J. M. et al. Coding facial expressions with gabor wavelets. In: IEEE COMPUTER SOCIETY. *Proceedings of the 3rd. International Conference on Face & Gesture Recognition*. Nara, Japão, 1998. (FG '98, v. 3), p. pp. 200–205. Citado 2 vezes nas páginas 46 e 48.
- MATLAB. *Matlab version 8.2.0 (R2013b)*. Natick, Massachusetts, 2013. Citado na página 61.
- PHILLIPS, P. J.; MOON, H. P. J.; RAUSS, S. R. The FERET evaluation methodology for face recognition algorithms. In: *Transactions on Pattern Analysis and Machine Intelligence*. [S.l.: s.n.], 2000. v. 22, n. 10, p. 1090–1104. Citado 2 vezes nas páginas 46 e 52.
- PLATT, J. C. Fast training for support vector machine using sequential minimal optimization. In: SCHÖLKOPF, B.; BURGESS, C. J. C.; SMOLA, A. J. (Ed.). *Advances in Kernel Methods*. Cambridge, MA, EUA: MIT Press, 1998. p. 185–208. Citado na página 42.
- QIU, G. et al. Curvelet entropy for facial expression recognition. In: QIU, G. et al. (Ed.). *Advances in Multimedia Information Processing - PCM 2010*. [S.l.: s.n.], 2010. (Lecture Notes in Computer Science, v. 6292), p. 617–628. Citado na página 70.
- SAMARIA, F.; HARTER, A. Parameterisation of a stochastic model for human face identification. *Proceedings of 2nd IEEE Workshop on Applications of Computer Vision*, p. 138–142, Dezembro 1994. Citado 4 vezes nas páginas 46, 47, 52 e 54.

SCHWENKER, F. et al. Orientation histogram for face recognition. In: SCHWENKER, F.; MARINAI, S. (Ed.). *Artificial Neural Network in Pattern Recognition*. [S.l.: s.n.], 2006. (Lecture Notes in Computer Science, v. 4087), p. 253–259. Citado na página 69.

SILVA, A. L.; CINTRA, M. E. Estudo de métodos para reconhecimento de padrões faciais utilizando algoritmos genéticos e análise de componentes principais. In: IFRN. *VII Escola Potiguar de Computação e suas Aplicações - TI como Fator de Desenvolvimento Regional*. Santa Cruz - RN, 2014. Citado na página 72.

SILVA, A. L.; CINTRA, M. E. Reconhecimento de padrões faciais: Um estudo. In: *Proceedings ENIAC 2015*. [S.l.: s.n.], 2015. p. 224–231. Citado 2 vezes nas páginas 57 e 72.

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. Citado na página 62.

SOLINA, F. et al. Color-based face detection in the "15 seconds of fame" art installation. *Conference on Computer Vision/Computer Graphics Collaboration for Model-based Imaging, Rendering, image Analysis and Graphical special Effects*, Univerza v Ljubljani, p. 38–47, Março 2003. Citado na página 46.

SRIVASTAVA, H. A comparison based study on biometrics for human recognition. *International Organization Of Scientific Research Journal of Computer Engineering*, v. 15, p. 22–29, Outubro 2013. Citado 2 vezes nas páginas 19 e 20.

TAIGMAN, Y. et al. Deepface: Closing the gap to human-level performance in face verification. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, EUA: IEEE Computer Society, 2014. (CVPR '14). Citado 2 vezes nas páginas 53 e 54.

THAKUR, S. et al. Face recognition using principal component analysis and RBF neural networks. In: *Proceedings of the 2008 First International Conference on Emerging Trends in Engineering and Technology*. Washington, DC, EUA: IEEE Computer Society, 2008. (ICETET '08), p. 695–700. Citado 2 vezes nas páginas 69 e 70.

THOMAZ, C. E.; GIRALDI, G. A. A new ranking method for principal components analysis and its application to face image analysis. *Image and Vision Computing*, São Bernardo do Campo, São Paulo, Brazil, v. 28, n. 6, p. 902–913, Junho 2010. Citado 3 vezes nas páginas 46, 47 e 52.

TURK, M.; PENTLAND, A. Eigenfaces for recognition. *J. Cognitive Neuroscience*, MIT Press, Cambridge, MA, EUA, v. 3, n. 1, p. 71–86, jan. 1991. Citado 2 vezes nas páginas 25 e 62.

YANG, J.; ZHANG DAVID AN FRANGI, F. A.; YANG, J.-y. Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 26, n. 1, janeiro 2004. Citado na página 62.

ZENICOLA, L. F. B.; SILVA, I. N. d. *Sistemas de Reconhecimento Facial*. Dissertação (Mestrado) — Escola de Engenharia de São Carlos da Universidade de São Paulo, 2013. Citado na página 14.

---

ZHAO, W. et al. Face recognition - a literature survey. *ACM Computing Surveys*, v. 35, n. 4, p. 399–458, Dezembro 2003. Citado na página [15](#).

## **Anexos**

## ANEXO A – Códigos

Este anexo apresenta os códigos das implementações em Matlab utilizadas nos experimentos realizados neste trabalho. Na seção [A.1](#) estão os códigos referentes aos experimentos para determinar a dimensionalidade dos vetores de características. Na seção [A.2](#) estão os códigos referentes aos experimentos para avaliação das métricas de similaridade. Na seção [A.3](#) estão os códigos referentes a extração de características e composição do banco de características a ser utilizado no *software* Weka.

### A.1 Código para determinar a dimensionalidade dos vetores de características

#### A.1.1 Banco de imagens JAFFE

O código a seguir implementa o método PCA e apresenta o gráfico de crescimento da variância em relação ao número de componentes utilizadas para obter a dimensionalidade dos vetores de características extraídos do banco de imagens JAFFE.

Código A.1 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA do banco de imagens JAFFE

```
n_pessoas = 10;
amostra = 19;
filename = '\caminho\jaffe\s';
linha = 256;
coluna = 256;

for kfold=1:20
    faces = zeros((linha*coluna), (n_pessoas*amostra));
    face_media = zeros(linha*coluna,1);
    facemedia_sub = zeros((linha*coluna), (n_pessoas*amostra));
    facetest = zeros(linha*coluna,1);
    facetest_media = zeros(linha*coluna,1);
    n = 1;
    c = 1;

    for i = 1:n_pessoas
        dir = strcat(filename,num2str(i),'\');
        for j = 1:20
            if j~=kfold
                final_path = strcat(dir,num2str(j),'.tiff');
                img = imread(final_path,'tiff');
                for p = 1:linha
                    for k = 1:coluna
```

```
                faces(c,n) = img(p,k);
                c = c+1;
            end
        end
        n = n + 1;
        c = 1;
    end
end

%% Cálculo da face média
face_media = mean(faces,2);

%% Subtração pela face média
for i = 1:(n_pessoas*amostra)
    facemedia_sub(:,i) = faces(:,i)-face_media;
end

%% Cálculo da matriz de covariância
mcov = cov(facemedia_sub);

%% Cálculo das eigenfaces
[autovet_pca,autoval_pca] = eig(mcov);
autoval_pca = eig(mcov);
autoval_pca = sort(autoval_pca,'descend');
variancia = zeros(size(autoval_pca));
total = sum(autoval_pca.^2);

%% Cálculo da variância
for i = 1:size(autoval_pca)
    parcial = 0;
    for j = 1:i
        parcial = parcial + autoval_pca(j)^2;
    end
    variancia(i) = 100*(parcial/total);
end
plot(1:size(autoval_pca),variancia,'b -');

title('Número ótimo de eigenfaces')
ylabel('Variância acumulada (%)');
xlabel('Número de eigenfaces');
hold on
plot(14,99,'ro');
end
```

Os códigos A.2 e A.3 implementam o método PCA/LDA e apresentam o gráfico de crescimento da variância em relação a número de componentes utilizadas para obter a dimensionalidade dos vetores de características extraídos do banco de imagens JAFFE. O código A.2 é referente a implementação do PCA e o código A.3 é referente ao LDA.

### Código A.2 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA/LDA do banco de imagens JAFFE - parte 1

```
%% Alocação de variáveis
pca=14; % número de componentes do PCA
k1=2;
n_pessoas = 10;
amostra = 18;
linha = 256;
coluna = 256;
database = '\caminho\jaffe\s'; %C:\Users\UFERSA\orl\s

for kfold=1:20
    faces = zeros((linha*coluna), (amostra*n_pessoas));
    face_media = zeros(linha*coluna,1);
    facetest = zeros(linha*coluna,1);
    facemedia_sub = zeros((linha*coluna), (amostra*n_pessoas));
    mcov = zeros((amostra*n_pessoas), (amostra*n_pessoas));
    prin_comp = 181-pca;
    padrao_medio = zeros(pca,1);
    pessoa_media = zeros(pca,10);
    mdisp_intraclass = zeros(pca,pca);
    mdisp_interclass = zeros(pca,pca);
    disp = zeros(pca,pca);
    n = 1;
    c = 1;

    %% Leitura das imagens
    for i = 1:n_pessoas
        dir = strcat(database,num2str(i),'\');
        for j = 1:20
            if j~=kfold
                image_file = strcat(dir,num2str(j),'.tiff');
                img = imread(image_file,'tif');
                c = 1;
                for p = 1:linha
                    for k = 1:coluna
                        faces(c,n) = img(p,k); % Transforma imagem em vetor
                        c = c + 1;
                    end
                end
            end
        end
    end
end
```

```

                n = n + 1;
            end
        end
    end

    %Cálculo da face média
    face_media = mean(faces,2);

    %Subtração pela face média
    for i = 1:n_pessoas*amostra
        facemedia_sub(:,i) = faces(:,i)-face_media;
    end

    %Matriz de covariância
    mcov = cov(facemedia_sub);

    %Calcula as eigenfaces
    [autovet_pca, autoval_pca] = eig(mcov);
    autoval_pca = eig(mcov);
    faces_pca = ...
        autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';

    %Projeção no espaço de características
    vetor_padrao_pca = faces_pca*facemedia_sub;
    save('jaffe');
    fisherOptimum2();

end

```

Código A.3 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA/LDA do banco de imagens JAFFE - parte 2

```

function [] = fisherOptimum2( )
    %% Carrega as características extraídas pelo PCA
    load('jaffe');
    padrao_medio = mean(vetor_padrao_pca,2);
    ini = 1;
    prox = amostra;

    for i = 1:n_pessoas
        pessoa_media(:,i) = mean(vetor_padrao_pca(:,ini:prox),2);
        ini = ini + amostra;
        prox = prox + amostra;
    end

    ini = 1;

```

```
prox = amostra;

%% Matriz de dispersão intra-classe
for i = 1:n_pessoas
    mdisp_intraclass = mdisp_intraclass + ...
        cov((vetor_padrao_pca(:,ini:prox))');
    ini = ini + amostra;
    prox = prox + amostra;
end

%% Matriz de dispersão entre classes
for i = 1:n_pessoas
    mdisp_interclass = mdisp_interclass + ...
        (padrao_medio-pessoa_media(:,i))*(padrao_medio-pessoa_media(:,i))';
end

%% Cálculo das fisherfaces
mdisp_interclass = 19.*mdisp_interclass;
inv_mdisp_intraclass = inv(mdisp_intraclass);
disp = inv_mdisp_intraclass * mdisp_interclass;
[autovet_lda,autoval_lda] = eig(disp);
autoval_lda = eig(disp);

%% Cálculo da variância
variância = zeros(size(autoval_lda));
total = sum(autoval_lda.^2);

for i = 1:size(autoval_lda)
    parcial = 0;
    for j = 1:i
        parcial = parcial + autoval_lda(j)^2;
    end
    variância(i) = 100*(parcial/total);
end

plot(1:size(autoval_lda),sort(variância),'b .-');
title('Número ótimo de fisherfaces')
ylabel('Variância acumulada (%)');
xlabel('Número de fisherfaces');
hold on
plot(10,100,'ro');
end
```

### A.1.2 Banco de imagens ORL

O código a seguir implementa o método PCA e apresenta o gráfico de crescimento da variância em relação ao número de componentes utilizadas para obter a dimensionalidade dos vetores de características extraídos do banco de imagens ORL.

Código A.4 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA do banco de imagens ORL

```

%% Alocação de variáveis
n_pessoas = 40;
amostra = 9;
database = '\caminho\orl\s'; %C:\Users\UFERSA\orl\s
linha = 112;
coluna = 92;

for kfold=1:10
    faces = zeros((linha*coluna), (n_pessoas*amostra));
    face_media = zeros(linha*coluna,1);
    facemedia_sub = zeros((linha*coluna), (n_pessoas*amostra));
    face_test = zeros(linha*coluna,1);
    facetest_sub = zeros(linha*coluna,1);
    n = 1;
    c = 1;

    for i = 1:n_pessoas

        person_path = strcat(database,num2str(i),'\');

        for j = 1:10
            if j~=kfold
                final_path = strcat(person_path,num2str(j),'.pgm');
                img = imread(final_path,'pgm');
                for p = 1:linha
                    for k = 1:coluna
                        faces(c,n) = img(p,k); % Transforma imagem em vetor
                        c = c+1;
                    end
                end
                n = n + 1;
                c = 1;
            end
        end
    end

    %% cálculo de face média

```

```

face_media = mean(faces,2);

%% Subtração pela face média
for i = 1:(n_pessoas*amostra)
    facemediasub(:,i) = faces(:,i)-face_media;
end

%% Cálculo da matriz de covariância
Mcov = cov(facemediasub);

%% Cálculo das eigenfaces
[autovet,autoval] = eig(Mcov);
autoval = eig(Mcov);

    %% Cálculo da variância
autoval = sort(autoval,'descend');
variancia = zeros(size(autoval));
total = sum(autoval.^2);
for i = 1:size(autoval)
    parcial = 0;
    for j = 1:i
        parcial = parcial + autoval(j)^2;
    end
    variancia(i) = 100*(parcial/total);
end

    plot(1:size(autoval),variancia,'b -');
    title('Número ótimo de eigenfaces');
    ylabel('Variância acumulada (%)');
    xlabel('Número de eigenfaces');
    hold on
    plot(30,99,'ro');

end

```

Os códigos [A.5](#) e [A.6](#) implementam o método PCA/LDA e apresentam o gráfico de crescimento da variância em relação ao número de componentes utilizadas para obter a dimensionalidade dos vetores de características extraídos do banco de imagens ORL. O código [A.5](#) é referente ao PCA e o código [A.6](#) é referente ao LDA.

Código A.5 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA/LDA do banco de imagens ORL - parte 1

```

n_pessoas = 40;
amostra = 9;
amostra_teste = 1;
amostra_total = 10;

```

```
database = '\caminho\database\s'; %C:\Users\UFERSA\orl\s
linha = 112;
coluna = 92;

%% Alocação de variáveis
pca=30;
for kfold=1:10;
faces = zeros((linha*coluna), (amostra*n_pessoas));
face_media = zeros(linha*coluna,1);
image_teste = zeros(linha*coluna,1);
facemedia_sub = zeros((linha*coluna), (amostra*n_pessoas));
mcov = zeros((amostra*n_pessoas), (amostra*n_pessoas));
prin_comp = 361-pca;
mean_projected_faces = zeros(pca,1);
pessoa_media = zeros(pca,40);
mtx_disp_intraclass = zeros(pca,pca);
mtx_disp_interclass = zeros(pca,pca);
optimum_pca = zeros(pca,pca);

n = 1;
c = 1;

%% Leitura das imagens
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\');
    for j = 1:10
        if j~=kfold
            image_file = strcat(dir,num2str(j),'.pgm');
            img = imread(image_file,'pgm');
            c = 1;
            for p = 1:linha
                for k = 1:coluna
                    faces(c,n) = img(p,k); % Transformação de ...
                        imagem em vetor
                    c = c + 1;
                end
            end
            n = n + 1;
        end
    end
end

%% Face média
face_media = mean(faces,2);

%% Subtração das imagens pela face média
for i = 1:n_pessoas*amostra
```

```

        facemedia_sub(:,i) = faces(:,i)-face_media;
    end

    %% Cálculo da Matriz de covariância
    mcov = cov(facemedia_sub);

    %% Cálculo das eigenfaces
    [autovet_pca, autoval_pca] = eig(mcov);
    autoval_pca = eig(mcov);

    %% Projeção no espaço de características do eigenfaces
    faces_pca = ...
        autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';
    vet_carac_pca = faces_pca*facemedia_sub; %vetores de ...
        características extraídos pelo PCA

    %% Fisherfaces %%
    save('optimum');
    xp2();
end

```

Código A.6 – Teste de dimensionalidade dos vetores de características extraídos pelo PCA/LDA do banco de imagens ORL - parte 2

```

function [ ] = xp2( )
%% Pré-alocação de variáveis
load ('optimum');
database = '\caminho\orl\s'; %C:\Users\UFERSA\orl\s

%% Face média
vet_carac_medio = mean(vet_carac_pca,2);
ini = 1;
prox = amostra;

for i = 1:n_pessoas
    pessoa_media(:,i) = mean(vet_carac_pca(:,ini:prox),2);
    ini = ini + amostra;
    prox = prox + amostra;
end

ini = 1;
prox = amostra;

%% Matriz de dispersão intra-classe
for i = 1:n_pessoas

```

```

    mtx_disp_intraclass = mtx_disp_intraclass + ...
        cov((vet_carac_pca(:,ini:prox))');
    ini = ini + amostra;
    prox = prox + amostra;
end

%% Matriz de dispersão entre classes
for i = 1:n_pessoas
    mtx_disp_interclass = mtx_disp_interclass + ...
        (vet_carac_medio-pessoa_media(:,i)) ...
        *(vet_carac_medio-pessoa_media(:,i))';
end

%% Cálculo da matriz de dispersão
mtx_disp_interclass = 9.*mtx_disp_interclass;
inverse_mtx_disp_intraclass = inv(mtx_disp_intraclass);
optimum_pca = inverse_mtx_disp_intraclass * mtx_disp_interclass;

%% Calcula as fisherfaces
[autovet_lda,autoval_lda] = eig(optimum_pca);
autoval_lda = eig(optimum_pca);
LDA = 19;

vet_carac_lda = autovet_lda(:,1:LDA);
faces_lda = vet_carac_lda'*vet_carac_pca;

%% Cálculo da variância
variância = zeros(size(autoval_lda));
total = sum(autoval_lda.^2);

for i = 1:size(autoval_lda)
    parcial = 0;
    for j = 1:i
        parcial = parcial + autoval_lda(j)^2;
    end
    variância(i) = 100*(parcial/total);
end

plot(1:size(autoval_lda),sort(variância),'b .-');
title('Número ótimo de fisherfaces')
ylabel('Variância acumulada (%)');
xlabel('Número de fisherfaces');
hold on
plot(19,100,'ro');

end

```

## A.2 Código para testes com métricas de similaridade

Nesta seção são apresentados os códigos dos experimentos para avaliação de métricas de similaridade na classificação de características extraídas pelo PCA/LDA nos bancos de imagens JAFFE e ORL.

### A.2.1 Banco de imagens JAFFE

A seguir são apresentados os códigos referentes aos experimentos para avaliação de métricas de similaridade na classificação de características extraídas pelo PCA/LDA no banco de imagens JAFFE. Com base nos resultados obtidos nos experimentos 5.1, foram utilizados 14 componentes para o método PCA e 10 componentes para o método LDA. O código A.7 é referente ao PCA e o código A.8 é referente ao LDA.

Código A.7 – Avaliação de métricas de similaridade na classificação de características extraídas pelo PCA/LDA no banco de imagens JAFFE (parte 1)

```
%% Alocação de variáveis
pca=14; % número de componentes do PCA
lda=10; % número de componentes do LDA
k1=2;
n_pessoas = 10;
amostra = 18;
linha = 256;
coluna = 256;
database = '\caminho\jaffe\s'; %C:\Users\UFERSA\jaffe\s

for kfold=1:10
    faces = zeros((linha*coluna), (amostra*n_pessoas));
    face_media = zeros(linha*coluna,1);
    facetest = zeros(linha*coluna,1);
    facemedia_sub = zeros((linha*coluna), (amostra*n_pessoas));
    mcov = zeros((amostra*n_pessoas), (amostra*n_pessoas));
    prin_comp = 181-pca;
    padrao_medio = zeros(pca,1);
    pessoa_media = zeros(pca,10);
    mdisp_intraclass = zeros(pca,pca);
    mdisp_interclass = zeros(pca,pca);
    disp = zeros(pca,pca);
    n = 1;
    c = 1;

    %% Leitura das imagens
    for i = 1:n_pessoas
```

```

dir = strcat(database,num2str(i),'\\');
for j = 1:20
    if j~=k1 && j~=(k1-1)
        image_file = strcat(dir,num2str(j),'.tiff');
        img = imread(image_file,'tif');
        c = 1;
        for p = 1:linha
            for k = 1:coluna
                faces(c,n) = img(p,k); % Transforma imagem em vetor
                c = c + 1;
            end
        end
        n = n + 1;
    end
end
end

%Cálculo da face média
face_media = mean(faces,2);

%Subtração pela face média
for i = 1:n_pessoas*amostra
    facemedia_sub(:,i) = faces(:,i)-face_media;
end

%Matriz de covariância
mcov = cov(facemedia_sub);

%Calcula as eigenfaces
[autovet_pca, autoval_pca] = eig(mcov);
autoval_pca = eig(mcov);
faces_pca = ...
    autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';

%Projeção no espaço de características
vetor_padrao_pca = faces_pca*facemedia_sub;
save('jaffe');
FLD2();
k1=k1+2;
end

```

### Código A.8 – PCA/LDA para o banco de imagens JAFFE (parte 2)

```

function [ ] = FLD2( )
%% Pré-alocação de variáveis
load ('jaffe');

```

```

database = '\caminho\jaffe\s'; %C:\Users\UFERSA\jaffe\s

padrao_medio = mean(vetor_padrao_pca,2);
ini = 1;
prox = amostra;

for i = 1:n_pessoas
    pessoa_media(:,i) = mean(vetor_padrao_pca(:,ini:prox),2);
    ini = ini + amostra;
    prox = prox + amostra;
end
ini = 1;
prox = amostra;

%% Matriz de dispersão intra-classe
for i = 1:n_pessoas
    mdisp_intraclass = mdisp_intraclass + ...
        cov((vetor_padrao_pca(:,ini:prox))');
    ini = ini + amostra;
    prox = prox + amostra;
end

%% Matriz de dispersão entre classes
for i = 1:n_pessoas
    mdisp_interclass = mdisp_interclass + ...
        (padrao_medio-pessoa_media(:,i)) ...
        *(padrao_medio-pessoa_media(:,i))';
end

%%
mdisp_interclass = 19.*mdisp_interclass;
inv_mdisp_intraclass = inv(mdisp_intraclass);
disp = inv_mdisp_intraclass * mdisp_interclass;
[autovet_lda,autoval_lda] = eig(disp);
autoval_lda = eig(disp);
vetor_padrao_lda = autovet_lda(:,1:lda);
faces_lda = vetor_padrao_lda'*vetor_padrao_pca;

%% Teste
acc = 0;
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\');
    for j = 1:20
        if j == k1
            image_file = strcat(dir,num2str(j),'.tiff');
            img = imread(image_file,'tiff');

```

```

c = 1;
for p = 1:linha
    for k = 1:coluna
        facetest(c,1) = img(p,k);
        c = c+1;
    end
end

facetest_sub = facetest - face_media;
facetest_eigen = faces_pca*facetest_sub;
facetest_fisher = vetor_padrao_lda'*facetest_eigen;
temp_dist = flintmax;
classe = 100000000;

for h = 1:(n_pessoas*amostra)
    %euclidiana
    dist = sqrt(sum((faces_lda(:,h) -facetest_fisher).^2));

    %Manhattan
    %distance = ...
        sqrt(sum(abs((fisher_faces_databases(:,h) ...
            -test_fisher_face).^2)));

    %Cosseno
    %norm_x = norm(fisher_faces_databases(:,h));
    %norm_y = norm(test_fisher_face);
    %distance = - (fisher_faces_databases(:,h)' ...
        *test_fisher_face)/(norm_x*norm_y);

    %Mahalanobis
    %[a,b]=size(fisher_faces_databases(:,h));
    %covar = eye(a,a);
    %norm_x = sqrt(fisher_faces_databases(:,h)' ...
        *covar*fisher_faces_databases(:,h));
    %norm_y = sqrt(test_fisher_face' *covar ...
        *test_fisher_face);
    %distance = - (fisher_faces_databases(:,h)' ...
        *covar*test_fisher_face)/(norm_x*norm_y);

    if dist < temp_dist
        temp_dist = dist;
        classe = h;
    end
end
classe = ceil(classe/amostra);
if classe == i
    acc = acc + 1;

```

```

        end
    end
end

for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\\');
    for j = 1:20
        if j == (k1-1)
            image_file = strcat(dir,num2str(j),'.tiff');
            img = imread(image_file,'tiff');
            c = 1;
            for p = 1:linha
                for k = 1:coluna
                    facetest(c,1) = img(p,k);
                    c = c+1;
                end
            end
        end

        facetest_sub = facetest - face_media;
        facetest_eigen = faces_pca*facetest_sub;
        facetest_fisher = vetor_padrao_lda'*facetest_eigen;
        temp_dist = flintmax;
        classe = 100000000;

        for h = 1:(n_pessoas*amostra)
            %euclidiana
            distance = sqrt(sum((fisher_faces_databases(:,h) ...
                -test_fisher_face).^2));

            %Manhattan
            %distance = ...
            sqrt(sum(abs((fisher_faces_databases(:,h) ...
                -test_fisher_face).^2)));

            %Cosseno
            %norm_x = norm(fisher_faces_databases(:,h));
            %norm_y = norm(test_fisher_face);
            %distance = - (fisher_faces_databases(:,h)' ...
                *test_fisher_face) / (norm_x*norm_y);

            %Mahalanobis
            %[a,b]=size(faces_lda(:,h));
            %covar = eye(a,a);
            %norm_x = sqrt(faces_lda(:,h)' *covar *faces_lda(:,h));
            %norm_y = sqrt(facetest_fisher' *covar ...
                *facetest_fisher);
        end
    end
end

```



```

pessoa_media = zeros(pca,40);
mtx_disp_intraclass = zeros(pca,pca);
mtx_disp_interclass = zeros(pca,pca);
optimum_pca = zeros(pca,pca);
n = 1;
c = 1;

%% Leitura das imagens
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\\');
    for j = 1:10
        if j~=kfold
            image_file = strcat(dir,num2str(j),'.pgm');
            img = imread(image_file,'pgm');
            c = 1;
            for p = 1:linha
                for k = 1:coluna
                    faces(c,n) = img(p,k); % Transformação de ...
                        imagem em vetor
                    c = c + 1;
                end
            end
            n = n + 1;
        end
    end
end

%% Face média
face_media = mean(faces,2);

%% Subtrair imagens da face média
for i = 1:n_pessoas*amostra
    facemedia_sub(:,i) = faces(:,i)-face_media;
end

%% Matriz de covariância
mcov = cov(facemedia_sub);

%% Cálculo das eigenfaces
[autovet_pca, autoval_pca] = eig(mcov);
autoval_pca = eig(mcov);

%% Projeção no espaço de características do eigenfaces
faces_pca = ...
    autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';
vet_carac_pca = faces_pca*facemedia_sub; %vetores de ...
    características extraídos pelo PCA

```

```

%% Fisherfaces %%
    save('eigen');
    FLD2();
end

```

### Código A.10 – PCA/LDA para o banco de imagens ORL (parte 2)

```

function [ ] = FLD2( )
%% Caregamento de variáveis
    load ('eigen');
    database = '\caminho\jaffe\s'; %C:\Users\UFERSA\jaffe\s

%% Cálculo da face média
    vet_carac_medio = mean(vet_carac_pca,2);

%% Cálculo de média por pessoa
    ini = 1;
    prox = amostra;
    for i = 1:n_pessoas
        pessoa_medio(:,i) = mean(vet_carac_pca(:,ini:prox),2);
        ini = ini + amostra;
        prox = prox + amostra;
    end

%% Matriz de dispersão intra-classe
    ini = 1;
    prox = amostra;
    for i = 1:n_pessoas
        mtx_disp_intraclass = mtx_disp_intraclass + ...
            cov((vet_carac_pca(:,ini:prox))');
        ini = ini + amostra;
        prox = prox + amostra;
    end

%% Matriz de dispersão entre classes
    for i = 1:n_pessoas
        mtx_disp_interclass = mtx_disp_interclass + ...
            (vet_carac_medio-pessoa_medio(:,i)) ...
            *(vet_carac_medio-pessoa_medio(:,i))';
    end

%% Cálculo da matriz de dispersão
    mtx_disp_interclass = 9.*mtx_disp_interclass;
    inverse_mtx_disp_intraclass = inv(mtx_disp_intraclass);

```

```

optimum_pca = inverse_mtx_disp_intraclass * mtx_disp_interclass;

%% Calcula as fisherfaces
[autovet_lda,autoval_lda] = eig(optimum_pca);
autoval_lda = eig(optimum_pca);
LDA = 19;

vet_carac_lda = autovet_lda(:,1:LDA);
faces_lda = vet_carac_lda'*vet_carac_pca;
acc = 0;

%% Teste
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\');
    for j = 1:10
        if j == kfold
            image_file = strcat(dir,num2str(j),'.pgm');
            img = imread(image_file,'pgm');
            c = 1;
            for p = 1:linha
                for k = 1:coluna
                    image_teste(c,1) = img(p,k);
                    c = c+1;
                end
            end
            facetest_sub = image_teste - face_media;
            test_eigen_face = faces_pca*facetest_sub;
            test_fisher_face = vet_carac_lda'*test_eigen_face;
            temp_distance = 10000000;
            classe = 10000000;
            for h = 1:(n_pessoas*amostra)
                % Métricas de similaridade
                %euclidiana
                distance = sqrt(sum((fisher_faces_databases(:,h) ...
                    -test_fisher_face).^2));

                %Manhattan
                %distance = ...
                sqrt(sum(abs((fisher_faces_databases(:,h) ...
                    -test_fisher_face).^2)));

                %Cosseno
                %norm_x = norm(faces_lda(:,h));
                %norm_y = norm(test_fisher_face);
                %distance = - (faces_lda(:,h)' ...
                    *test_fisher_face)/(norm_x*norm_y);
            end
        end
    end
end

```

```

    %mahalanobis
    %[a,b]=size(fisher_faces_databases(:,h));
    %covar = eye(a,a);
    %norm_x = sqrt(fisher_faces_databases(:,h)' ...
        *covar*fisher_faces_databases(:,h));
    %norm_y = ...
        sqrt(test_fisher_face'*covar*test_fisher_face);
    %distance = - (fisher_faces_databases(:,h)' ...
        *covar*test_fisher_face)/(norm_x*norm_y);

    if distance < temp_distance
        temp_distance = distance;
        classe = h;
    end
end
classe = ceil(classe/amostra);
if classe == i
    acc = acc + 1;
end

end
end
end

fprintf('Acurácia %f \n', (acc/40)*100);

```

### A.3 Código para extração de características

Nesta seção são apresentados dos códigos referentes a extração de características dos bancos de imagens ORL e JAFFE e a construção de seus respectivos bancos de características utilizados no *software* Weka.

#### A.3.1 Banco de imagens JAFFE

A seguir está o código referente a extração de características do banco de imagens JAFFE. Ao final da execução deste código é gerado um arquivo .ARFF no formato dos bancos de dados utilizados pelo Weka.

Código A.11 – PCA/LDA para extração de características do banco de imagens JAFFE (parte 1)

```

%% Alocação de variáveis
n_pessoas = 10;
amostra = 20;

```

```
database = '\caminho\database2\s';
linha = 256;
coluna = 256;
pca=14;
%for kfold=1:10;
faces = zeros((linha*coluna), (amostra*n_pessoas));
face_media = zeros(linha*coluna,1);
image_teste = zeros(linha*coluna,1);
facemedia_sub = zeros((linha*coluna), (amostra*n_pessoas));
mcov = zeros((amostra*n_pessoas), (amostra*n_pessoas));
prin_comp = 201-pca;
mean_projected_faces = zeros(pca,1);
pessoa_media = zeros(pca,10);
mtx_disp_intraclass = zeros(pca,pca);
mtx_disp_interclass = zeros(pca,pca);
optimum_pca = zeros(pca,pca);
n = 1;
c = 1;

%% Leitura das imagens
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\');

%% Transformação de imagem em vetor
    for j = 1:20
        image_file = strcat(dir,num2str(j),'.tiff');
        img = imread(image_file,'tiff');
        c = 1;
        for p = 1:linha
            for k = 1:coluna
                faces(c,n) = img(p,k);
                c = c + 1;
            end
        end
        n = n + 1;
    end
end

%% Face média
face_media = mean(faces,2);

%% Subtrair imagens da face média
for i = 1:n_pessoas*amostra
    facemedia_sub(:,i) = faces(:,i)-face_media;
end

%% Matriz de covariância
```

```

mcov = cov(facemedia_sub);

%% Cálculo das eigenfaces
[autovet_pca, autoval_pca] = eig(mcov);
autoval_pca = eig(mcov);

%% Projeção no espaço de características do eigenfaces
faces_pca = ...
    autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';
vet_carac_pca = faces_pca*facemedia_sub; %vetores de ...
    características extraídos pelo PCA

%% Fisherfaces %%
save('eigen');
FLD2();
end

```

### Código A.12 – PCA/LDA para extração de características o banco de imagens JAFFE (parte 2)

```

function [ ] = FLD2( )
%% Pré-alocação de variáveis
load ('eigen');
database = '\caminho\database2\s';

%% Cálculo da face média
vet_carac_medio = mean(vet_carac_pca,2);

%% Cálculo de média por pessoa
ini = 1;
prox = amostra;
for i = 1:n_pessoas
    pessoa_media(:,i) = mean(vet_carac_pca(:,ini:prox),2);
    ini = ini + amostra;
    prox = prox + amostra;
end

%% Matriz de dispersão intra-classe
ini = 1;
prox = amostra;
for i = 1:n_pessoas
    mtx_disp_intraclass = mtx_disp_intraclass + ...
        cov((vet_carac_pca(:,ini:prox))');
    ini = ini + amostra;
    prox = prox + amostra;
end

```

```

end

%% Matriz de dispersão entre classes
for i = 1:n_pessoas
    mtx_disp_interclass = mtx_disp_interclass + ...
        (vet_carac_medio-pessoa_media(:,i))*(vet_carac_medio-pessoa_media(:,i))';
end

%% Cálculo da matriz de dispersão
mtx_disp_interclass = 9.*mtx_disp_interclass;
inverse_mtx_disp_intraclass = inv(mtx_disp_intraclass);
optimum_pca = inverse_mtx_disp_intraclass * mtx_disp_interclass;

%% Calcula as fisherfaces
[autovet_lda,autoval_lda] = eig(optimum_pca);
autoval_lda = eig(optimum_pca);
LDA = 10;

vet_carac_lda = autovet_lda(:,1:LDA);
faces_lda = vet_carac_lda'*vet_carac_pca; %características extraídas

%% Escreve no arquivo
f1 = sprintf('DB.arff');
file = fopen(f1,'w');

%Define a relação
fprintf(file,'@RELATION face\n\n');

%Escreve a lista de atributos
for a=1:10
    fprintf(file,'@attribute f%i real\n',a);
end

% @attribute class {ID1, ID2, ID3, ... IDn}
fprintf(file,'@attribute class {ID1}');
for a=2:10
    fprintf(file,',ID%i',a);
end
fprintf(file,']\n\n');

%
fprintf(file,'@DATA\n');
for z1=1:20
    xk=z1;
    for z2=1:10
        fprintf(file,'%10f,', faces_lda(:,xk));
    end
end

```

```

        fprintf(file, 'ID%i\n', z2);
        xk=xk+20;
    end
end
fclose(file);
end

```

### A.3.2 Banco de imagens ORL

A seguir está o código referente a extração de características do banco de imagens ORL. Ao final da execução deste código é gerado um arquivo .ARFF no formato dos bancos de dados utilizados pelo Weka.

#### Código A.13 – PCA/LDA para extração de características do banco de imagens ORL (parte 1)

```

%% Alocação de variáveis
n_pessoas = 40;
amostra = 10;
database = '\caminho\database\s';
linha = 112;
coluna = 92;
pca=30;
%for kfold=1:10;
faces = zeros((linha*coluna), (amostra*n_pessoas));
face_media = zeros(linha*coluna,1);
image_teste = zeros(linha*coluna,1);
facemedia_sub = zeros((linha*coluna), (amostra*n_pessoas));
mcov = zeros((amostra*n_pessoas), (amostra*n_pessoas));
prin_comp = 401-pca;
mean_projected_faces = zeros(pca,1);
pessoa_media = zeros(pca,40);
mtx_disp_intraclass = zeros(pca,pca);
mtx_disp_interclass = zeros(pca,pca);
optimum_pca = zeros(pca,pca);
n = 1;
c = 1;

%% Leitura das imagens
for i = 1:n_pessoas
    dir = strcat(database,num2str(i),'\');

%% Transformação de imagem em vetor
    for j = 1:10
        image_file = strcat(dir,num2str(j),'.pgm');
        img = imread(image_file,'pgm');
    end
end

```

```

        c = 1;
        for p = 1:linha
            for k = 1:coluna
                faces(c,n) = img(p,k);
                c = c + 1;
            end
        end
        n = n + 1;
    end
end

%% Face média
face_media = mean(faces,2);

%% Subtrair imagens da face média
for i = 1:n_pessoas*amostra
    facemedia_sub(:,i) = faces(:,i)-face_media;
end

%% Matriz de covariância
mcov = cov(facemedia_sub);

%% Cálculo das eigenfaces
[autovet_pca, autoval_pca] = eig(mcov);
autoval_pca = eig(mcov);

%% Projeção no espaço de características do eigenfaces
faces_pca = ...
    autovet_pca(:,prin_comp:(amostra*n_pessoas))*facemedia_sub';
vet_carac_pca = faces_pca*facemedia_sub; %vetores de ...
    características extraídos pelo PCA

%% Fisherfaces %%
save('eigen');
FLD2();
%end

```

Código A.14 – PCA/LDA para extração de características do banco de imagens ORL (parte 2)

```

function [ ] = FLD2( )
%% Pré-alocação de variáveis
load ('eigen');
database = '\\caminho\database\s';

%% Cálculo da face média

```

```
vet_carac_medio = mean(vet_carac_pca,2);

% Cálculo de média por pessoa
ini = 1;
prox = amostra;
for i = 1:n_pessoas
    pessoa_media(:,i) = mean(vet_carac_pca(:,ini:prox),2);
    ini = ini + amostra;
    prox = prox + amostra;
end

% Matriz de dispersão intra-classe
ini = 1;
prox = amostra;
for i = 1:n_pessoas
    mtx_disp_intraclass = mtx_disp_intraclass + ...
        cov((vet_carac_pca(:,ini:prox))');
    ini = ini + amostra;
    prox = prox + amostra;
end

% Matriz de dispersão entre classes
for i = 1:n_pessoas
    mtx_disp_interclass = mtx_disp_interclass + ...
        (vet_carac_medio-pessoa_media(:,i))*(vet_carac_medio-pessoa_media(:,i))';
end

% Cálculo da matriz de dispersão
mtx_disp_interclass = 9.*mtx_disp_interclass;
inverse_mtx_disp_intraclass = inv(mtx_disp_intraclass);
optimum_pca = inverse_mtx_disp_intraclass * mtx_disp_interclass;

% Calcula as fisherfaces
[autovet_lda,autoval_lda] = eig(optimum_pca);
autoval_lda = eig(optimum_pca);
LDA = 19;

vet_carac_lda = autovet_lda(:,1:LDA);
faces_lda = vet_carac_lda'*vet_carac_pca; %características extraídas

% Escreve no arquivo
f1 = sprintf('DB.arff');
file = fopen(f1,'w');

%Define a relação
fprintf(file, '@RELATION face\n\n');
```

```
%Escreve a lista de atributos
for a=1:15
    fprintf(file, '@attribute f%i real\n', a);
end

% @attribute class {ID1, ID2, ID3, ... IDn}
fprintf(file, '@attribute class {ID1}');
for a=2:40
    fprintf(file, ', ID%i', a);
end
fprintf(file, '}\n\n');

%
fprintf(file, '@DATA\n');
for z1=1:10
    xk=z1;
    for z2=1:40
        fprintf(file, '%10f, ', faces_lda(:, xk));
        fprintf(file, 'ID%i\n', z2);
        xk=xk+10;
    end
end
fclose(file);
end
```