



UNIVERSIDADE FEDERAL DO CEARÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO

SAMUEL LIMA SILVA

ANÁLISE DE COMPORTAMENTO E DESEMPENHO DOS ALGORITMOS
COMPUTACIONAIS DE RECONHECIMENTO FACIAL

SOBRAL

2023

SAMUEL LIMA SILVA

ANÁLISE DE COMPORTAMENTO E DESEMPENHO DOS ALGORITMOS
COMPUTACIONAIS DE RECONHECIMENTO FACIAL

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia da Computação do Campus Sobral da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia da Computação.

Orientador: Prof. Dr. Ialis Cavalcante de Paula Júnior

SOBRAL

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- L71a Lima Silva, Samuel.
Análise de Comportamento e Desempenho dos Algoritmos Computacionais de Reconhecimento facial / Samuel Lima Silva. – 2023.
41 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Sobral, Curso de Engenharia da Computação, Sobral, 2023.
Orientação: Prof. Dr. Ialis Cavalcante de Paula Júnior.
1. Reconhecimento facial. 2. Análise de dados. 3. Detecção Facial. 4. Eigenfaces. 5. Fisherfaces. I. Título.

CDD 621.39

SAMUEL LIMA SILVA

ANÁLISE DE COMPORTAMENTO E DESEMPENHO DOS ALGORITMOS
COMPUTACIONAIS DE RECONHECIMENTO FACIAL

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia da Computação do Campus Sobral da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia da Computação.

Aprovada em: 18 de Julho de 2023

BANCA EXAMINADORA

Prof. Dr. Ialis Cavalcante de Paula
Júnior (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Me. Erick Aguiar Donato
Universidade Federal do Ceará (UFC)

Stefane Adna dos Santos
Universidade Federal do Ceará (UFC)

Dedico este trabalho a todas as pessoas que de alguma forma contribuíram para a minha formação acadêmica e pessoal. Em especial, agradeço aos meus pais por todo o amor, apoio e incentivo ao longo dessa jornada. Também dedico a minha família, amigos e professores, que me proporcionaram momentos únicos e inesquecíveis e que me ajudaram a chegar até aqui. Este trabalho é resultado da minha dedicação e do incentivo daqueles que sempre estiveram ao meu lado.

AGRADECIMENTOS

Gostaria de aproveitar este momento para expressar minha profunda gratidão a todas as pessoas que me apoiaram e permaneceram ao meu lado durante esta jornada acadêmica. Sem o apoio incondicional de cada um de vocês, a conclusão deste trabalho não seria possível.

Primeiramente, quero agradecer aos meus pais, Dulcivane e Eudes, por serem minha base sólida e me encorajarem a nunca desistir dos meus sonhos. Obrigado por estarem sempre presentes e por serem meu apoio inabalável em todos os momentos.

Aos meus irmãos, Silas e Sergio, vocês são mais do que familiares, são verdadeiros amigos que me acompanharam e me motivaram em cada etapa desta jornada acadêmica.

Aos amigos da nona turma de Engenharia da Computação, que se tornaram uma família durante esses anos de estudo, levo todos vocês no coração e serei eternamente grato pela amizade e pela cumplicidade que compartilhamos.

Aos meus companheiros de república, Silas, Serginho, Ian, Aglailson, Erick e Gabriel, obrigado por tornarem minha experiência na república vivida e por serem verdadeiros amigos que fizeram cada dia mais especial.

Ao meu grande amigo Hélio Saboia, com quem compartilhei histórias lendárias e momentos inesquecíveis durante nossa jornada na faculdade, agradeço por toda camaradagem e amizade que construímos de verdade.

À minha namorada e meu grande amor, Geordanna Geyse, sou profundamente grato por todo o apoio, incentivo e compreensão que você me ensinou ao longo desse caminho desafiador. Sua presença em minha vida fez toda a diferença, e sem você, nada disso seria possível.

Não posso deixar de agradecer a todos os meus professores da UFC, Sobral, que me forneceram conhecimento, orientação e inspiração ao longo da minha jornada acadêmica.

Em especial, sou imensamente grato ao meu orientador, Prof. Dr. Ialis Cavalcante de Paula Júnior, por sua orientação, paciência e expertise, que foram fundamentais para a realização deste trabalho.

Por fim, expresso minha gratidão a Deus, pois sem Sua graça e direção, nada disso teria se concretizado. A Ele atribui todas as conquistas e oportunidades em minha vida.

A todos vocês, meu sincero agradecimento por fazerem parte da minha história e me ajudarem a alcançar este grande passo em minha vida acadêmica.

Muito obrigado!

“A imaginação é mais importante que o conhecimento. O conhecimento é limitado. A imaginação envolve o mundo.”

(Albert Einstein)

RESUMO

Atualmente, as técnicas de inteligência artificial são consideradas as soluções mais poderosas para resolver diversos tipos de problemas comerciais. De acordo com pesquisas, o futuro e o presente da automação e do desenvolvimento de aplicações comerciais serão feitos utilizando essas técnicas, o que indica a necessidade de empresas contarem com profissionais qualificados nessa área. O objetivo deste trabalho é realizar uma análise do comportamento de algoritmos de reconhecimento facial e as técnicas utilizadas para executar esse processo. Através deste documento, é possível entender melhor o funcionamento de algoritmos como o *Eigenfaces* e o *Fisherfaces*, além de compará-los com base em um conjunto de dados, para determinar qual apresenta melhor desempenho exclusivo para o reconhecimento de padrões.

Palavras-chave: Reconhecimento facial; Análise de dados; Detecção Facial; *Eigenfaces*; *Fisherfaces*.

ABSTRACT

Currently artificial intelligence techniques are the most powerful solutions to solve practically any type of commercial problem, there are researches that indicate that the future and the present of automation and also of the development of commercial applications will be done using these techniques, this indicates that companies need, or will need, qualified professionals in this area. The focus of this work is to carry out an analysis on the behavior of facial recognition algorithms, and the techniques they use to carry out this process. Through this document we can get a better idea of how facial recognition and facial detection algorithms work, such as eigenfaces and fisherfaces, in addition to purchasing them according to a database, and defining which of the two performs better according to with the analyzed database.

Keywords: Facial recognition; Data analysis; Facial detection; Eigenfaces; Fisherfaces.

LISTA DE FIGURAS

Figura 1 – Segmentação - A esquerda: Imagem obtida através do smartphone; A direita: Imagem obtida através da câmera de segurança.	18
Figura 2 – Reconhecimento de Padrões - A esquerda: Imagem original; A direita: Partes reconhecidas	19
Figura 3 – Etapas do Reconhecimento Facial	21
Figura 4 – Análise dos Componentes Principais	23
Figura 5 – (a) Conjunto de Imagens Yale; (b) Eigenfaces; (c) Face Média	24
Figura 6 – Imagens onde foram aplicadas fisherfaces	27
Figura 7 – (a) EigenFaces num plano 2D; (b) Fisherfaces num plano 2D	29
Figura 8 – Nomenclatura das imagens	33
Figura 9 – Banco AT&T - <i>EigenFaces</i>	35
Figura 10 – Banco Yale - <i>EigenFaces</i>	36
Figura 11 – Banco <i>Sam</i> - <i>EigenFaces</i>	36
Figura 12 – Banco AT&T - <i>FisherFaces</i>	37
Figura 13 – Banco Yale - <i>FisherFaces</i>	37
Figura 14 – Banco <i>Sam</i> - <i>FisherFaces</i>	38

LISTA DE TABELAS

Tabela 1 – Atributos Relacionados a Biometria	15
Tabela 2 – Fases Gerais Relacionadas a Visão Computacional	17

LISTA DE ABREVIATURAS E SIGLAS

AFGR	<i>International Conference on Automatic Face and Gesture Recognition</i>
LDA	Análise Discriminante Linear
PCA	Análise dos Componentes Principais
PDI	Processamento Digital de Imagem

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Objetivos Gerais	14
1.2	Objetivos Específicos	14
1.3	Justificativa	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Biometria	15
2.2	Processamento Digital de Imagem	16
2.3	Visão Computacional	16
2.3.1	<i>Segmentação</i>	17
2.3.2	<i>Reconhecimento de Padrões</i>	18
2.3.3	<i>Rastreamento</i>	20
2.4	Reconhecimento Facial	20
2.4.1	<i>Detecção Facial</i>	21
2.4.2	<i>Extração das Características</i>	22
2.5	Análise dos Componentes Principais e Algoritmo Eigenfaces	23
2.5.1	<i>Análise dos componentes principais</i>	23
2.5.2	<i>Eigenfaces</i>	23
2.6	Análise do Discriminante Linear e Algoritmo Fisherfaces	26
2.6.1	<i>Análise do Discriminante Linear</i>	26
2.6.2	<i>Fisherface</i>	27
2.7	Reconhecimento Facial	30
3	TRABALHOS RELACIONADOS	31
3.1	Trabalho Relacionado A	31
3.2	Trabalho Relacionado B	31
4	METODOLOGIA	32
4.1	Banco de dados	33
4.2	Detectando Faces e Processo de Treinamento	34
5	RESULTADOS	35
5.1	EigenFaces	35
5.2	FisherFaces	37

6	CONCLUSÃO	39
	REFERÊNCIAS	40

1 INTRODUÇÃO

Durante os últimos anos, houve uma migração significativa de produtos e serviços para plataformas digitais. Como consequência desse movimento, surge a necessidade de se criar processos ágeis para identificação de indivíduos. Cada pessoa é única, portanto, deve possuir senhas de acesso particulares. No entanto, isso pode não garantir a autenticidade de uma determinada pessoa que realiza o *login* em um sistema *online*.

Por esse motivo, diversos recursos tecnológicos têm sido desenvolvidos para promover maior qualidade e segurança para esses usuários, sendo um deles o método de reconhecimento facial. Essa técnica baseia-se em técnicas de biometria, que consistem em analisar e identificar padrões, como o formato do rosto, a distância entre os olhos, o tamanho da boca, entre outros (ORSO *et al.*, 2020).

Além disso, o reconhecimento facial oferece vantagens, como a eliminação da necessidade de contato direto entre um usuário e um sistema *online*. Por exemplo, podemos optar pela utilização de uma câmera capaz de capturar o rosto do usuário para efetuar o *login*. O custo de implementação é acessível para sistemas que utilizam plataformas digitais, pois esses sistemas geralmente contam com dispositivos que facilitam a obtenção de fotos e imagens.

O objetivo deste trabalho é avaliar o desempenho de alguns algoritmos de reconhecimento facial. Para isso, serão estudados algoritmos que utilizam técnicas amplamente difundidas na literatura, como a Análise Discriminante Linear (LDA) e a Análise dos Componentes Principais (PCA). Essas técnicas consistem em reduzir redundâncias e reconhecer padrões em um conjunto de dados, utilizando procedimentos que envolvem álgebra linear e estatística aplicada (ELEYAN; DEMIREL, 2006).

O algoritmo *EigenFaces* será analisado com base na técnica PCA, por apresentar uma implementação mais aprimorada e resultados melhores do que a técnica PCA sendo utilizada de maneira isolada. Nesse algoritmo, a técnica PCA é aplicada para extrair vetores nomeados *eigenvectors*, os quais são um grupo de elementos com maior relevância para uma base de dados de treinamento. O princípio desse algoritmo é construir uma “face média” e, a partir das *eigenfaces*, gerar inúmeras faces semelhantes aos dados do banco de treinamento (SAHA; BHATTACHARJEE, 2013).

O mesmo procedimento será executado para o algoritmo *FisherFaces*, utilizando a técnica LDA, conhecida por reduzir a dimensionalidade dos dados de um determinado conjunto, eliminando as *fisherfaces* que caracterizam uma base de dados consoante o grupo de dados

utilizados no processo de treinamento.

Para autenticar os algoritmos executados neste trabalho, serão utilizados *data set's* de imagens, como o AT&T, constituído por 400 imagens de pessoas, e o *Database Yale Face Type A*, que possui outras 165 imagens de pessoas. Ambos são disponibilizados gratuitamente para estudos em reconhecimento facial e outras áreas de biometria.

1.1 Objetivos Gerais

O objetivo geral deste trabalho consiste em analisar o comportamento e desempenho exclusivo para o reconhecimento de padrões dos algoritmos computacionais de reconhecimento facial *EigenFaces* e *FisherFaces*.

1.2 Objetivos Específicos

- Avaliar o desempenho individual de cada algoritmo no reconhecimento de padrões utilizando um conjunto de três bancos de dados, sendo eles o AT&T, o *Yale* e um banco de dados criado pelo autor.
- Concluir como é possível melhorar a eficiência na identificação de falsos positivos pelos algoritmos.

1.3 Justificativa

Os estudos relacionados à visão computacional e ao reconhecimento facial estão em destaque atualmente, uma vez que a utilização desses conceitos e técnicas pode proporcionar maior segurança em ambientes que exigem a identificação de indivíduos ou sistemas que requerem credenciais únicas para sua utilização.

A área de reconhecimento facial já faz parte do cotidiano das pessoas, seja em sistemas de segurança que utilizam câmeras para identificar indivíduos específicos, seja no processo de desbloqueio facial presente em *smartphones* mais modernos.

Nesse contexto, o estudo dos algoritmos responsáveis pelo procedimento de reconhecimento facial é crucial, uma vez que a compreensão e análise desses algoritmos permitem a criação de novas ferramentas ou aprimoramento das já existentes. Ademais, diversas empresas atuantes no setor tecnológico desenvolvem softwares capazes de realizar o reconhecimento facial, tornando este trabalho um ponto de partida para futuros estudos mais aprofundados.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Biometria

A biometria é a ciência que permite demonstrar a identidade de um indivíduo de acordo com seus atributos físicos, químicos e comportamentais. Essa ciência consiste em comprovar e demonstrar que cada indivíduo possui características únicas, sendo considerado singular (COSTA *et al.*, 2006).

A partir desse conceito, podemos definir que o corpo humano ou seu perfil comportamental pode ser utilizado para identificá-lo de maneira única por meio da biometria. Logo, os dados biométricos não são apenas fisiológicos, mas também comportamentais. A biometria pode ser dividida em duas categorias distintas, que utilizam métricas diferentes (SANTOS; FABRIS, 2019):

- **Biometria Fisiológica:** Está diretamente ligada ao corpo humano, apresenta extrema dificuldade de falsificação e poucas chances de perder suas características ao longo do tempo. As mais conhecidas são as impressões digitais, formato da mão, retina, íris e o DNA.
- **Biometria Comportamental:** É um fator dependente da condição psicológica ou mesmo do ambiente no qual o indivíduo está inserido. Pode sofrer variação segundo o estado emocional do indivíduo ou doenças, podendo modificar, por exemplo, sua voz e seu modo de andar. Como exemplos mais comuns, temos a voz e a caligrafia.

Cada gênero de biometria possui suas vantagens e desvantagens, portanto, medir essas características é algo essencial para avaliar a eficiência do modelo de biometria escolhido. A Tabela 1 apresenta os principais atributos relacionados a sistemas biométricos.

Tabela 1 – Atributos Relacionados a Biometria

Atributo	Definição
Acessibilidade	Analisa se o traço métrico é disponível para leitura, como por exemplo a leitura das impressões digitais realizadas por máquinas.
Desempenho	Analisa o índice de reconhecimento, ou seja, se a identificação do indivíduo através da biometria está sendo executada de maneira correta.
Evasão	Analisa se o traço biométrico pode ser alterado, como por exemplo considerar indivíduos gêmeos idênticos.
Mensurabilidade	Analisa se a característica capturada através da biometria pode ser mensurada.

Fonte: Autor, 2023

2.2 Processamento Digital de Imagem

O Processamento Digital de Imagem (PDI), é a forma de manipular uma imagem de modo que tanto a entrada como a saída de um processo seja uma imagem. A ideia em processar uma imagem é que ela seja retratada com mais efetividade no contexto computacional, visto que a mesma é convertida em dados. Esses dados capturados após o processamento são submetidos a técnicas com o intuito de obter variações favoráveis da imagem para as mesmas poderem possivelmente ser utilizadas em outros processamentos, tais como conversões e inserções (ALBUQUERQUE; ALBUQUERQUE, 2000).

Existe um espectro que varia desde o processamento digital de imagens até a visão computacional. Esse espectro possui 3 níveis distintos, sendo eles: baixo-nível (operações primitivas como, por exemplo, diminuição de ruído e melhoria de contraste), médio-nível (processos como, por exemplo, classificação e segmentação) e alto-nível (operações relacionadas às tarefas de cognição associadas à visão humana).

2.3 Visão Computacional

Defini-se como Visão Computacional um conjunto de dados e técnicas que tornam sistemas de computador capazes de extrair informações de imagens. É a ciência responsável pela forma como um computador enxerga o meio ao seu redor, para extrair informações significativas a partir das imagens capturadas no ambiente, seja por meio de câmeras, vídeos, sensores, scanners ou outros dispositivos (MARENGONI; STRINGHINI, 2009).

A visão computacional pode ser aplicada em diversas áreas do conhecimento, como inteligência artificial, aprendizado de máquina (visão cognitiva), neurobiologia (visão biológica), robótica (visão em robôs), dentre outras. Além disso, pode ser utilizada em aplicações para efetuar reconhecimento de padrões, segurança em ambientes, bem como diversas outras atividades que necessitam da visão biológica. O objetivo da visão computacional é tomar decisões eficazes sobre objetos físicos reais ou ter como objetivo construir descrições de cenas a partir de imagens (MILANO; HONORATO, 2014).

Anteriormente, os níveis de tratamento das imagens foram caracterizados. Podemos observar que durante o PDI, os processos de baixo-nível consistem na eliminação de ruídos e no melhoramento do contraste em imagens. Na visão computacional, trabalhamos com as imagens no nível médio, ou seja, realizamos segmentação e reconhecimento de padrões para que,

posteriormente, possamos trabalhar em nível-alto, onde podemos operar processos cognitivos como a identificação de um objeto numa sequência de imagens (MARENGONI; STRINGHINI, 2009).

Um sistema de visão computacional é composto por uma série de fases. É importante citar que não existe uma definição única para as fases de um sistema em visão computacional, pois cada sistema necessita de um conjunto específico de etapas durante seu processo de desenvolvimento (BACKES; JUNIOR, 2019). Porém, existem fases gerais que são aplicáveis durante a construção do sistema. As principais fases gerais são resumidas a seguir na Tabela 2.

Tabela 2 – Fases Gerais Relacionadas a Visão Computacional

Atributo	Definição
Aquisição	Responsável pela captura das imagens, ou seja, simulação da capacidade dos olhos.
Processamento de Imagem	Responsável por aplicar uma melhoria na imagem, para aumentar as chances de sucesso nas etapas seguintes.
Segmentação	Responsável por particionar a imagem em regiões de interesses.
Extração de Características	Responsável por obter um conjunto de características do objeto de interesse.
Reconhecimento de Padrões e Rastreo	Responsáveis pelo último passo referente a visão computacional onde depois de todo o tratamento que a imagem recebe (ou recebeu) torna-se possível reconhecer determinado indivíduo ou objeto, assim como rastreá-lo.

Fonte: Autor, 2023

2.3.1 Segmentação

O processo de segmentação é responsável por dividir a imagem em regiões ou objetos diferentes, guiado pelas características dos objetos ou das regiões, como a coloração de um objeto ou a distância em relação a outra coisa. A qualidade da segmentação depende diretamente da qualidade da imagem analisada (ANTONELLO, 2014).

A Figura 1 exemplifica o processo de segmentação. Se tirarmos uma foto de um estacionamento com vários carros de cores diferentes usando um *smartphone*, não conseguiremos encontrar um carro específico (como um carro amarelo) a partir dessa imagem, pois os carros ocupam grandes espaços em relação ao tamanho total da imagem, sendo segmentos classificados como pontos grandes. Por outro lado, se observarmos a imagem capturada pela câmera de segurança instalada em um local elevado no estacionamento, os carros ocupam espaços menores em relação ao tamanho total da imagem, sendo segmentos classificados como regiões menores

(podemos ver o carro amarelo no canto inferior esquerdo da imagem da câmera de segurança).

Figura 1 – Segmentação - A esquerda: Imagem obtida através do smartphone; A direita: Imagem obtida através da câmera de segurança.



Fonte: Autor, 2023

Embora o procedimento seja segmentar a imagem, a resolução das imagens diferem, logo o tratamento utilizado no processo de segmentação também podem ser diferentes. Existem métodos distintos de segmentação são eles: segmentação por detecção de borda, segmentação por corte e segmentação por crescimento de região (ANTONELLO, 2014).

2.3.2 Reconhecimento de Padrões

Padrões são características repetitivas que definem um objeto ou um indivíduo, tais como formas, texturas, cores, dimensões, entre outros. O processo de reconhecimento de padrões é responsável por gerenciar uma intersecção com a área de inteligência artificial. Para executar este processo, um sistema necessita de conhecimento sobre os objetos a serem reconhecidos, que pode ser obtido por meio de uma base de dados implementada diretamente no código (sistemas baseados em regras, por exemplo) ou por meio de um conjunto de amostras dos objetos a serem reconhecidos, manipulando técnicas de aprendizado de máquina (ANTONELLO, 2014).

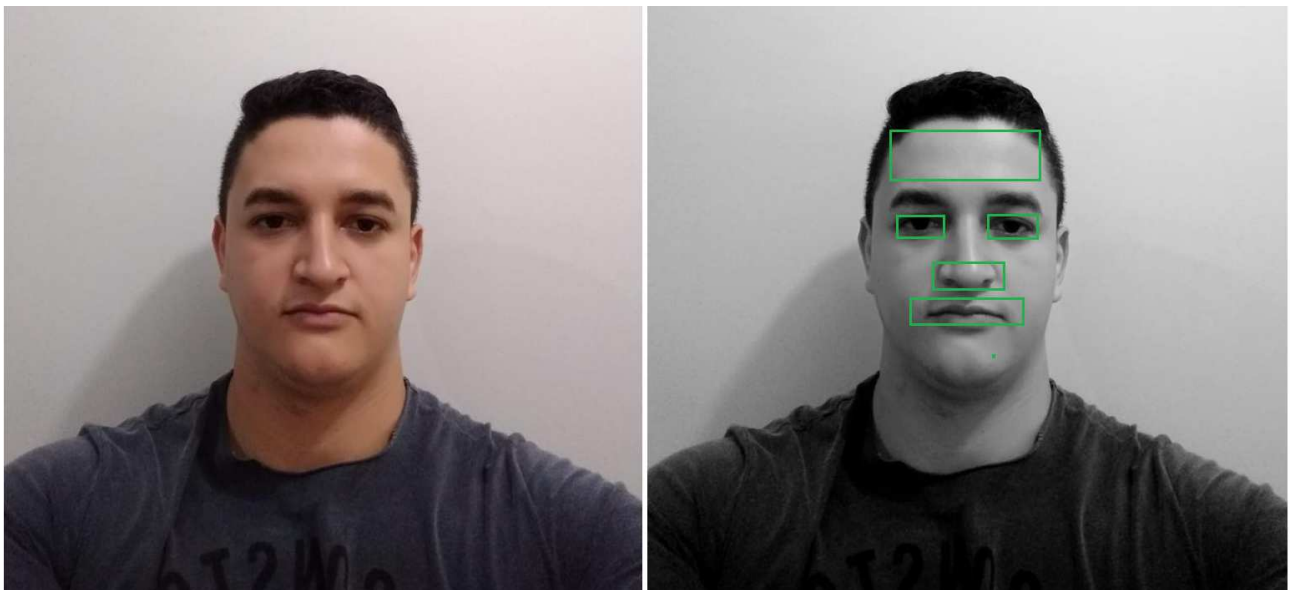
O reconhecimento de objetos é essencial na visão computacional e está diretamente ligado ao reconhecimento de padrões. Ao realizarmos o reconhecimento individual dos padrões citados anteriormente, é possível reconhecer um objeto ou um indivíduo na totalidade. Podemos imaginar os padrões como peças de um quebra-cabeça que, quando juntas, permitem reconhecer o objeto ou indivíduo.

As técnicas de reconhecimento de padrões podem ser divididas em dois grupos (ANTONELLO, 2014):

- **Grupo Estrutural:** Os padrões são retratados de forma simbólica e a estrutura é a forma como esses padrões se relacionam;
- **Teoria da Decisão:** Neste grupo os padrões são caracterizados por propriedades quantitativas e partir disso é tomada a decisão se o objeto/indivíduo possui ou não estas propriedades.

Ainda existe a possibilidade de o reconhecimento de padrões ser composto por técnicas usadas em ambos os grupos mencionados anteriormente. Por exemplo, no processo de reconhecimento de faces, usamos um modelo estrutural para determinar o local com maior probabilidade de encontrar os componentes de uma face (cabelo, testa, sobrancelhas, olhos, orelhas, nariz, boca, entre outros) para, em seguida, afirmar que o que está sendo reconhecido é uma face. A Figura 2 exibe o reconhecimento dos padrões da face de um indivíduo, como boca, testa, nariz e olhos.

Figura 2 – Reconhecimento de Padrões - A esquerda: Imagem original;
A direita: Partes reconhecidas



Fonte: Autor, 2023

Como mencionado anteriormente, o reconhecimento dessas partes pode ser feito de maneiras diferentes. Por exemplo, os olhos podem ser reconhecidos por meio de um algoritmo que usa rede neural, enquanto a boca, o nariz e a testa podem ser reconhecidos por um algoritmo que usa a distância mínima entre diferentes texturas captadas. Todas essas técnicas podem ser classificadas como técnicas de Teoria da Decisão.

2.3.3 Rastreamento

O rastreamento é o procedimento de reconhecer um conjunto de padrões em uma sequência de imagens. No entanto, buscar esses padrões em cada imagem de um conjunto de imagens sem a utilização de métodos específicos não é eficiente. O processo de rastreamento envolve uma noção sobre o movimento do objeto ou indivíduo que está sendo rastreado, a fim de minimizar a busca entre o conjunto de imagens. Esses procedimentos são aplicáveis a diversas áreas, desde sistemas de segurança até o uso em sistemas de interface humano-computador, como em sistemas de reconhecimento facial online, por exemplo (ANTONELLO, 2014).

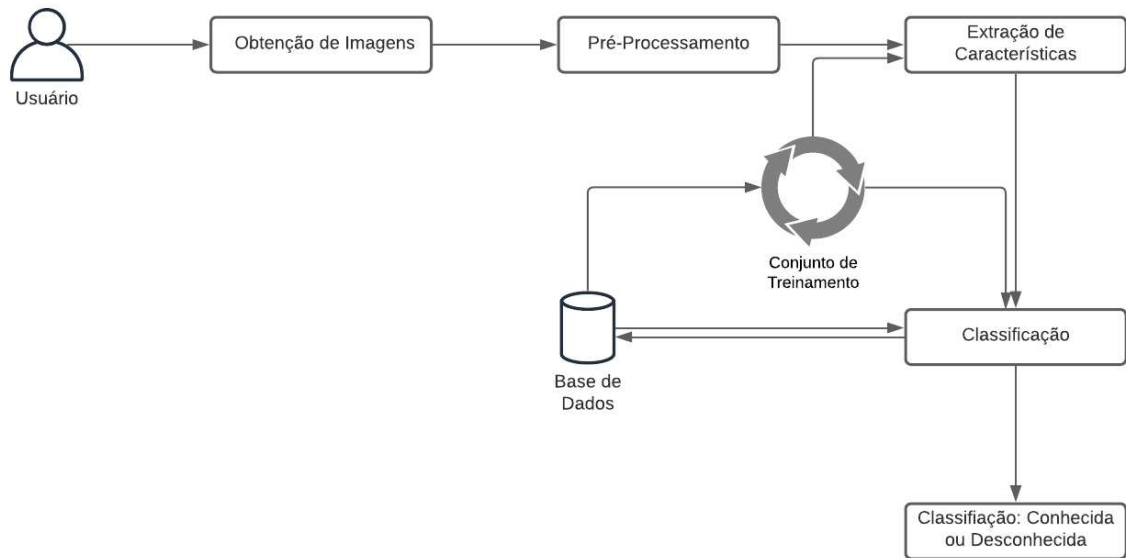
2.4 Reconhecimento Facial

Podemos dizer que o reconhecimento facial é algo simples, algo que realizamos diariamente. No entanto, quando essa tarefa é movida para os computadores, através da visão computacional e reconhecimento de padrões, temos um problema complexo que impõe diversos desafios para pesquisadores e desenvolvedores (SILVA *et al.*, 2020).

Quando um sistema computacional tenta executar a tarefa de reconhecer faces, utiliza-se um procedimento particular de reconhecimento de padrões. Esse procedimento tem sido cada vez mais aprimorado ao longo dos anos, devido à sua ampla gama de aplicações em diferentes setores. Nos últimos anos, os campos que têm se destacado são os de investigação forense e reconhecimento de foragidos (SIMOES, 2020).

O reconhecimento facial é tratado como um subproblema inserido no reconhecimento de padrões. Como não existe uma regra que defina um caminho correto para efetuar o reconhecimento facial, esse processo segue três tarefas fundamentais: aquisição/localização e detecção da face na imagem obtida, extração das características da face e, por fim, o reconhecimento propriamente dito. As etapas podem ser observadas na Figura 3 e serão detalhadas a seguir.

Figura 3 – Etapas do Reconhecimento Facial



Fonte: Autor, 2023

2.4.1 Detecção Facial

A ideia principal na detecção de uma face é criar uma imagem binária que represente os contornos de uma figura e, em seguida, remover atributos relacionados à face humana. No entanto, devido à grande variação de faces humanas que existem e seus diferentes atributos, como o uso de acessórios como óculos e chapéus, é possível que não haja detecção facial.

A pesquisa sobre detecção facial ficou estagnada por vários anos durante a década de 90 e retomada devido ao surgimento de conferências sobre o tema, sendo a principal delas a *International Conference on Automatic Face and Gesture Recognition (AFGR)*, e também devido à evolução das técnicas de reconhecimento (NORONHA *et al.*, 2019). Atualmente, existem quatro métodos bem difundidos:

- **Baseado em Aparência:** Este método não utiliza nenhum conhecimento prévio sobre o objeto ou sobre a característica que será detectada. Nesse tipo de algoritmo, surgem os conceitos de aprendizado e treinamento, visto que as informações fundamentais para realizar a tarefa de detecção são retiradas da própria mescla de imagens sem intervenção externa.
- **Baseado em Templates:** Uma forma esperada para detectar objetos é procurar pelo mesmo em uma imagem e analisar se ele corresponde a um arquétipo prévio da sua forma. Uma das formas mais simples de moldar a forma de um objeto é retratá-lo através de seus componentes geométricos básicos, como por exemplo, esferas, retângulos, triângulos. Essa

técnica é chamada de *template*. Portanto, a detecção de um objeto consiste em encontrar a melhor correlação, estabelecida por meio de uma função, entre o objeto presente na imagem e o seu molde *template*.

- **Baseado em Conhecimento:** Esse método retrata as técnicas de detecção de faces que utilizam uma base de regras definidas desde a compreensão prévia sobre o problema, isto é, métodos que contêm regras que definem o que é uma face, conforme o conhecimento do pesquisador. Por exemplo, a maioria das faces humanas possuem características válidas para a maioria dos indivíduos, dado que teoricamente todas as faces possuem dois olhos, um nariz e uma boca. Além disso, esses elementos estão configurados de uma maneira específica sobre a face. Com essas informações, é possível estabelecer regras que poderão identificar uma face humana. O problema que esse método sofre é em relação ao conjunto de regras, pois, se elas são muito gerais, pode ocasionar o risco de um sistema apresentar uma alta taxa de falsos-positivos, ou seja, elementos equivocadamente identificados como face.
- **Baseados em Características Invariantes:** Contrapondo o método baseado em conhecimento, os métodos baseados em características invariantes reúnem técnicas que visam achar características invariantes da face. Sobretudo, esse método é inspirado na habilidade que os seres humanos têm de identificar objetos independentemente do ponto de vista. A maior desvantagem que este método possui é que essas características podem sofrer alterações devido a certas condições tais como iluminação ou algum tipo de ruído, o que pode prejudicar sua eficiência (LOPES *et al.*, 2005). As principais características invariantes que os seres humanos possuem são a cor da pele e a textura da face, e utilizadas para para distinguir a face de outros objetos presentes em uma imagem.

2.4.2 *Extração das Características*

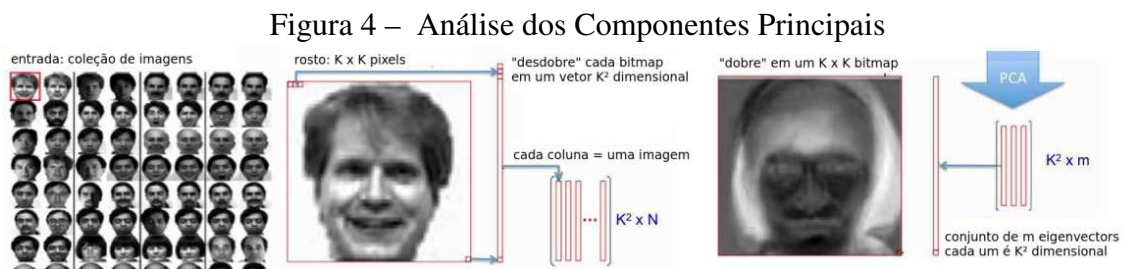
Uma imagem que contém faces humanas existe em um ambiente complexo que pode facilitar ou dificultar a detecção da face em todo o ambiente da imagem. Para realizar a extração de características da face, ela é isolada da melhor maneira possível. Podemos “recortar” a imagem nos limites da face e, em seguida, extrair as características faciais relevantes que serão utilizadas no processo de reconhecimento facial (BISSI *et al.*, 2018). A extração das características faciais gera um grupo de particularidades que retrata os traços do indivíduo. Neste trabalho, serão apresentados dois métodos que serão utilizados: o PCA em conjunto com o

algoritmo *eigenfaces* e o LDA em conjunto com o algoritmo *fisherfaces*.

2.5 Análise dos Componentes Principais e Algoritmo Eigenfaces

2.5.1 Análise dos componentes principais

O método *Eigenfaces* se baseia em projetar linearmente um grupo de imagens em um espaço de características com dimensões reduzidas obtidas a partir da análise dos componentes principais (PCA). No entanto, esse método produz direções de projeção que maximizam a dispersão dos pontos no gráfico em todas as classes, ou seja, em todas as faces são mantidas as variações indesejadas causadas por iluminação e expressões faciais (BISSI *et al.*, 2018). A Figura 4 mostra como é realizada a extração desses vetores.



Fonte: Adaptado de BISSI, 2023

Uma desvantagem ao utilizar o *Eigenfaces* é que, quando a PCA diminui a dimensionalidade, ela faz com que as projeções maximizem as dispersões entre as classes, assim como também maximiza a dispersão da própria imagem, o que é um efeito indesejável para a classificação. Assim, caso a base de dados contenha imagens de rostos variando sob iluminação forte, a projeção terá diversos atributos modificados graças à iluminação, diminuindo assim a efetividade no agrupamento dos dados.

2.5.2 Eigenfaces

O funcionamento do *Eigenfaces* é similar ao funcionamento da PCA, contudo a matriz de covariância é otimizada, reduzindo o processamento necessário para fazer o cálculo de seus autovetores e autovalores.

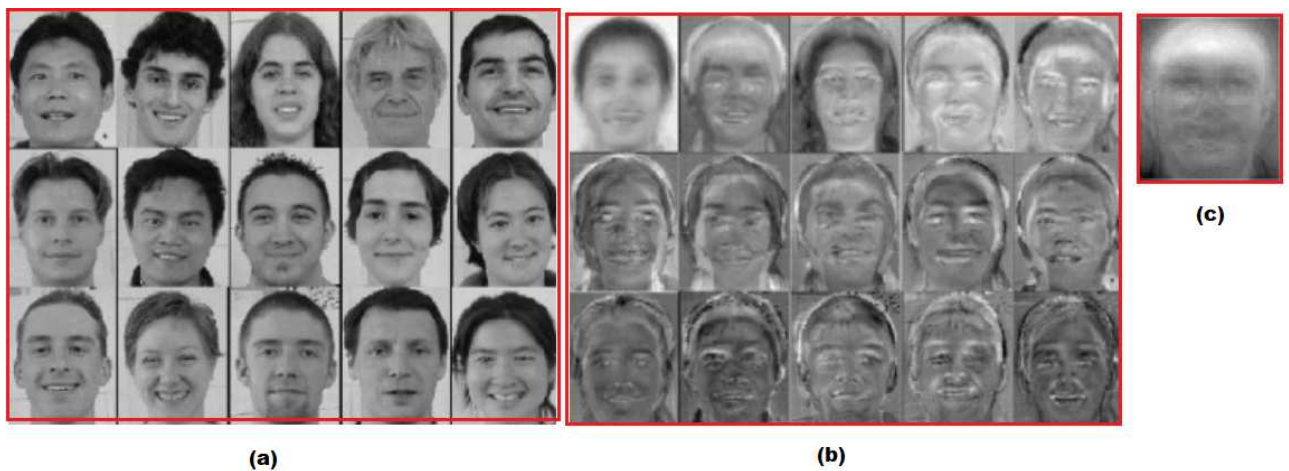
É importante lembrar que no *Eigenfaces*, a iluminação é um aspecto importante a ser considerado, visto que ele cria os *eigenvectors* baseado nas diferentes variações das faces, conforme as variações de luminosidade. Logo, se tivermos um grupo de imagens de faces de uma

mesma pessoa com variações de faces consideráveis, o resultado será o melhor possível. Porém, com a variação da iluminação para cada imagem, serão usadas as mesmas para determinar as *eigenfaces* de forma que essa quantidade de variações afetem a eficiência do algoritmo (SILVA; MARROQUIM, 2013).

Por ser baseado na Teoria da Informação, as *eigenfaces* procuram realizar a identificação de um pequeno número de características relevantes para diferenciar a face de um indivíduo da face de outro indivíduo. Esses atributos podem ser analisados apenas com a alteração dos valores assumidos pelos pixels em um conjunto de imagens de faces.

Nesse algoritmo, as imagens de treinamento irão retirar os aspectos mais relevantes da face humana e criar uma face média com base no conjunto de dados, e com a variação dos valores desses aspectos é possível reproduzir uma diversa gama de faces apenas executando multiplicações escalares utilizando os autovalores. Na realidade, o que acontece é que a combinação linear das *eigenfaces* representa cada face. Na Figura 5, temos em (a) um exemplo das faces do banco de dados *Yale*; em (b), podemos observar as *eigenfaces* extraídas e em (c), a face média (SILVA *et al.*, 2020).

Figura 5 – (a) Conjunto de Imagens Yale; (b) Eigenfaces; (c) Face Média



Fonte: Autor, 2023

Para explicar formalmente o *Eigenfaces* consideraremos um conjunto N de imagens x_1, x_2, \dots, x_N , que formam o grupo de vetores que se associam com os pixels em uma imagem. Primeiramente, é calculada a média das imagens.

$$M = 1/N \cdot \sum_{n=1}^N x_n$$

Após calculada a média (M) calculam-se os desvios de cada imagem em relação à média calculada.

$$\rho_i = x_i - M$$

O conjunto de vetores, que está definido por p , está normalizado em relação à média (M) e está pronto para ser utilizado no cálculo de autovalores α e autovetores β , que formam o conjunto de N vetores ortonormais, conforme a equação:

$$\alpha_k = 1/N \cdot \sum_{n=1}^N (\beta_k^T \cdot \rho_n)^2$$

O autovalor é sujeito a seguinte equação:

$$\beta_l^T \cdot \beta_k = \begin{cases} 1 & \text{se } l = k; \\ 0 & \text{se } l \neq k. \end{cases}$$

Esses autovetores e autovalores são designados a matriz de covariância definida por:

$$C = 1/N \cdot \sum_{n=1}^N \rho_n \rho_n^T = A \cdot A^T$$

Sendo a matriz A o conjunto de vetores normalizados em relação à média $A = [p_1, p_2, \dots, p_n]$. A matriz C é de $M^2 \times M^2$ e M^2 determina os autovalores e autovetores, inviável para o tamanho das imagens. Assim é necessário reduzir a dimensionalidade do M^2 , considerando V_i os autovetores A^T com a seguinte equação:

$$A^T \cdot A \cdot V = \alpha_i \cdot Vi$$

Multiplicando os dois lados por A resulta em:

$$A \cdot A^T \cdot A \cdot V = A \cdot \alpha_i \cdot Vi$$

Assim os autovetores de $C = A \cdot A^T$ serão iguais a $A \cdot Vi$. Desta forma é gerada a matriz de dimensão $M \times M$, conseqüentemente reduzindo o número de computações de M^2 para N número de imagens, fazendo com o que o conjunto de treinamento fique consideravelmente menor.

O algoritmo *Eigenfaces* é relevante por possibilitar não só a classificação das imagens, mas também sua reconstrução e compactação. Para classificar as faces, é utilizado um cálculo baseado na distância entre a imagem que está sendo examinada e a mesma imagem no novo espaço. Se esse valor estiver em um limite (*threshold*), a imagem é considerada uma face, caso contrário, é considerada um falso positivo (BISSI *et al.*, 2018). O valor dessa distância é definido conforme a técnica que melhor se adequar ao *data set* escolhido, sendo a mais utilizada a distância euclidiana, aplicando o algoritmo KNN.

2.6 Análise do Discriminante Linear e Algoritmo Fisherfaces

2.6.1 Análise do Discriminante Linear

Com o mesmo propósito do PCA, a Análise do Discriminante Linear (do inglês Linear Discriminant Analysis - LDA) também baseia-se na redução da dimensionalidade dos dados visando sua classificação. Em suma, a partir de uma gama de dados a serem utilizados, o LDA gera um conjunto com dados menores, ou seja, de menor dimensionalidade, mas que conseguem representar os dados originais (SILVA *et al.*, 2020).

Para realizar o processo de diminuição da dimensionalidade dos dados, o LDA cria um vetor capaz de construir um espaço de características que visa diminuir a dispersão intra-classe e aumentar a dispersão interclasse, ou seja, o LDA faz com que dados que tenham características semelhantes se afastem, e dados que tenham características distintas se afaste (SILVA *et al.*, 2020). Logo, após realizada a diminuição da dimensionalidade dos dados, o LDA cria um espaço de características onde os dados inseridos são afastados ou aproximados entre si de acordo com suas características gerais, na seção seguinte a Figura 7 irá comparar o PDA (*Eigenfaces*) e o LDA (Fisherfaces) em plano cartesiano.

2.6.2 Fisherface

O *Fisherface* é uma técnica que tem em vista aumentar a diferença entre dados, sua maior diferença em relação ao *Eigenfaces*, é que ele também diminui a dispersão em sua própria classe, o que não ocorre no algoritmo anterior, já que ele busca semelhança, uma vez que aumentar a diferença entre os dados ocasionaria a criação de falsos-positivos. Em suma, o *Fisherface* visa encontrar a melhor projeção em retas quem maximizam o distanciamento entre escaladores (GALIMBERTI, 2019). A figura 6 exibe uma amostra de imagens nas quais foram aplicadas *Fisherface*.

Figura 6 – Imagens onde foram aplicadas fisherfaces



Fonte: Adaptado de SILVA, 2023

Para exemplificar melhor o *Fisherface*, consideraremos duas classes distintas em um ambiente 2D, supondo um conjunto N de imagens (x_1, x_2, \dots, x_n) , inseridos em classes distintas, a classe 1 e a classe 2, denominados respectivamente por w_1 e w_2 , com M_1 e M_2 imagens. Para encontrarmos o melhor vetor para projeção, o vetor médio de cada classe inserido no espaço dimensional original e no espaço *Fisherface* é obtido pelas equações a seguir.

$$\alpha_i = 1/N_i \cdot \sum_{x \in w_i} x$$

$$\hat{\alpha}_i = 1/N_i \cdot \sum_{x \in w_i} y = 1/N_i \cdot \sum_{x \in w_i} W^T \cdot X = W^T \cdot \alpha_i$$

O *Fisherface* tem em vista aumentar a dispersão que é resultante da diferença entre as médias normalizadas inerentes a cada classe, a dispersão é correspondente a variância, representada a seguir.

$$\hat{S}_i^2 = \sum_{y \in w_i} (y - \hat{\alpha}_i)^2$$

O vetor de projeção das amostras é representado por W, cujo possui os atributos de uma mesma classe e faz com estas fiquem próximas e também deixa as médias entre outros conjuntos de dados mais distantes. O *Fisherface* é considerado uma função linear $W^T \cdot X$. Maximizando a função representada a seguir.

$$J(W) = \hat{\alpha}_i^2 = \sum_{y \in w_i} (y - \hat{\alpha}_i)^2$$

O objetivo de W é achar a melhor projeção possível, logo torna-se necessário expressar J(W) como uma função de W como na equação a seguir.

$$S_I = \sum_{x \in w_i} (x - \alpha_i) \cdot (X - \alpha_i)^T$$

Como o *Fisherface* é baseado na seleção dos vetores W que têm características semelhantes inseridas em uma mesma classe e assim, criar um espaço vetorial onde a dispersão entre as classes sejam melhoradas, a matriz de dispersão S_w é dada por $S_w = S_1 + S_2$, e a matriz de dispersão entre classes e inseridas nas mesmas é definida pelas respectivas equações.

$$S_B = \sum_{i=1}^c N_i \cdot (\alpha_i - \alpha) \cdot (\alpha_i - \alpha)^T$$

$$S_W = \sum_{i=1}^c \sum_{x_k \in X_i} (x_k - \alpha_i) \cdot (X_k - \alpha_i)^T$$

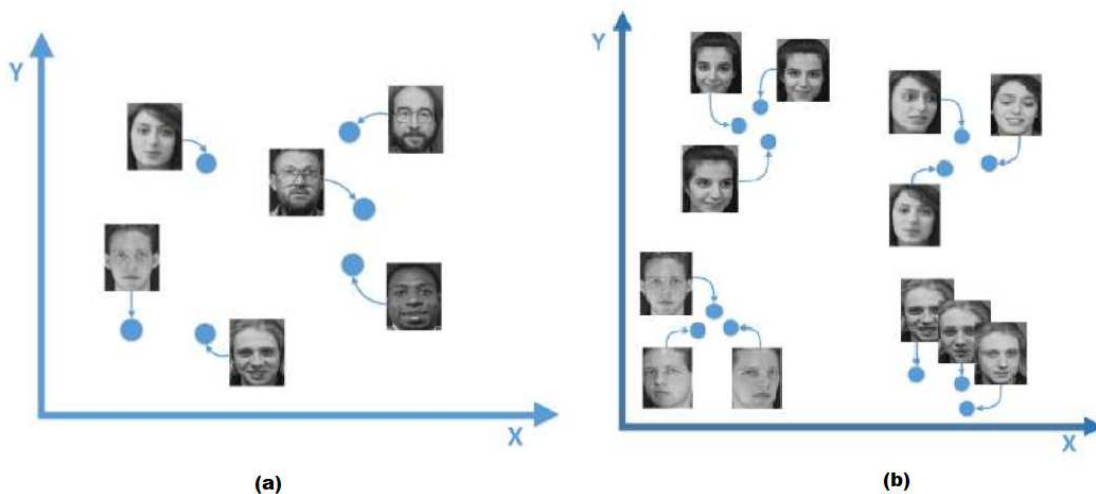
Onde α é a média do conjunto, α_i é a média das imagens de classe X_i , e o número de amostras inseridas na classe X_i é representada por N_i . O vetor de projeção $W_{\text{ótimo}}$ maximiza a razão entre o determinante da matriz de dispersão entre classes com a determinante matriz de dispersão inserida nas classes, isso se dá conforme a equação a seguir.

$$W_{\text{ótimo}} = \operatorname{argmax}(W^T \cdot S_B \cdot W) / (W^T \cdot S_W \cdot W) = [w_1, w_2, \dots, w_m]$$

Sendo X_i com $i = 1, 2, \dots, m$, o conjunto geral de autovetores S_B e S_W correspondente aos m maiores valores.

Como mencionado anteriormente, a Figura 7 exibe o funcionamento dos algoritmos em um espaço dimensional 2D, em (a) temos o funcionamento do *Eigenfaces*, pode-se observar que as imagens estão afastadas umas das outras e conforme as técnicas de PCA buscando características em comum para se agruparem, e (b) temos o funcionamento do *Fisherfaces* e podemos observar que as imagens estão afastadas das que possuem atributos diferentes, mas estão próximas das que possuem atributos semelhantes, utilizando assim as técnicas de LDA para se agruparem.

Figura 7 – (a) EigenFaces num plano 2D; (b) Fisherfaces num plano 2D



Fonte: Adaptado de Silva, 2023

2.7 Reconhecimento Facial

Uma vez finalizadas todas as etapas de localização, detecção de faces e extração de características, podemos afirmar que todos os atributos relevantes que podem ser adquiridos e representados pelas componentes principais estão prontos para serem utilizados, e temos então um espaço composto por inúmeras *eigenfaces* ou *fisherfaces*.

As imagens das faces de um banco de imagens são projetadas neste espaço, gerando vetores de características que representam a face média de um indivíduo. Esta face média é armazenada na base de dados e utilizada no processo de reconhecimento facial. Ao realizar uma consulta utilizando uma face de teste, ela é projetada no espaço de faces gerado, gerando seu próprio vetor característico, sendo então analisado. Após realizar a comparação com os vetores das imagens inseridas no espaço criado em um banco de dados por meio de um classificador, temos o reconhecimento facial final (ALVES *et al.*,).

3 TRABALHOS RELACIONADOS

3.1 Trabalho Relacionado A

Visando compreender os desafios e as soluções existentes na área de visão computacional, especificamente no que se refere ao reconhecimento facial, Kuriowa e Carro (2015) desenvolveram um estudo utilizando um *framework* criado para potencializar recursos de vigilância por meio de uma câmera digital. O estudo foi auxiliado por tecnologias *open source*, incluindo as bibliotecas *JavaCV* e *OpenCV*.

Segundo os autores, a detecção e o reconhecimento de faces são as etapas mais problemáticas do processo, e não atingem as taxas desejáveis ou são lentas na fase de detecção. Embora tenha sido alcançado o processo de reconhecer uma face como objeto, o número de falsos positivos era alto.

3.2 Trabalho Relacionado B

Visando utilizar classificadores, como o classificador KNN e classificadores por distância, como a média euclidiana, Fujikawa (2016) utilizou o banco de dados Yale para estudar quais métricas podem melhorar a eficiência dos algoritmos *Eigenfaces*, *Fisherfaces* e seus classificadores.

O autor afirma que o reconhecimento de faces não é uma tarefa trivial, uma vez que os resultados obtidos inicialmente, utilizando apenas as distâncias, não foram eficazes. Entretanto, foi possível aumentar consideravelmente a eficácia das abordagens de reconhecimento facial utilizando técnicas de pós-processamento e mapeamento, além da utilização de redes neurais para otimizar o funcionamento dos algoritmos.

4 METODOLOGIA

Durante todo o processo de implementação deste trabalho, a linguagem de programação *Python* na versão 3.8 foi utilizada, juntamente com as bibliotecas *PIL* para o tratamento de imagens, *Numpy* para operações matemáticas e a biblioteca *OpenCV* na versão 4.5 para o processamento das imagens. Todos esses recursos foram administrados pela IDE *PyCharm*, a qual é uma ferramenta intuitiva, rápida e eficaz.

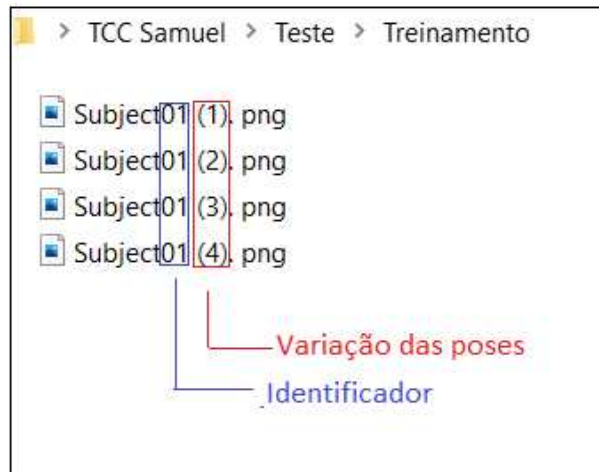
A escolha da linguagem de programação *Python* se deu por sua ampla utilização, grande número de comunidades ativas, boa curva de aprendizado e compatibilidade com a biblioteca *OpenCV*. Já a escolha da biblioteca *OpenCV* se deu pela sua grande variedade de ferramentas que possibilitam o processamento de imagens, como os algoritmos *EigenFaces* e *FisherFaces*, além de conter funções otimizadas que proporcionam uma união bem sucedida entre a biblioteca escolhida e a linguagem de programação.

O processo de reconhecimento facial completo compreende quatro etapas essenciais: coleta de imagens, detecção facial, treinamento com as imagens coletadas e, por último, reconhecimento facial. Cada uma dessas etapas será detalhada no decorrer deste trabalho, exibindo como foram definidas.

Os algoritmos *EigenFaces* e *FisherFaces* foram utilizados para as etapas de treinamento e reconhecimento facial, respectivamente. O primeiro utiliza a técnica de PCA para eliminar e diminuir a redundância de dados nas imagens, enquanto o segundo utiliza a LDA para diminuir a dimensão dos dados. A implementação desses algoritmos através da biblioteca *OpenCV*, em conjunto com *Python*, é simples e completa.

Foi utilizado o processo de aprendizado supervisionado, em que cada sujeito contido no banco de dados possui um identificador (ID) e diversas poses relacionadas. Dessa forma, é possível medir o grau de exatidão do algoritmo em relação às faces utilizadas nesse teste, além de identificar quais faces foram detectadas erradamente. A estrutura de cada uma das imagens pode ser visualizada na Figura 8.

Figura 8 – Nomenclatura das imagens



Fonte: Autor, 2023

4.1 Banco de dados

Para a elaboração deste trabalho, optou-se por utilizar bases de dados de imagens já existentes e consolidadas para testar os algoritmos *EigenFaces* e *FisherFaces*. Dessa forma, os bancos de dados utilizados foram o AT&T e o Yale Face. O Yale Face é um banco de dados que contém 165 imagens em escala de cinza, com dimensões de 320x243 pixels e formato .gif. Foram coletadas 10 imagens de cada uma das 15 pessoas, totalizando 150 imagens. As imagens apresentavam variações de iluminação, uso de óculos e expressões faciais como sonolência, tristeza e felicidade.

Para o treinamento do algoritmo, foram utilizadas 80 imagens, sendo 8 imagens por indivíduo, de um total de 15 indivíduos. Cinco indivíduos foram retirados do processo de treinamento e manipulados nos testes para avaliação dos falsos e verdadeiros negativos.

O banco de dados AT&T é composto por um arranjo de 400 fotos com imagens de 40 indivíduos, nas dimensões de 92x112 pixels e formato .pgm. Foram utilizadas 5 imagens de cada 30 indivíduo para treinamento, totalizando 150 imagens de treinamento. Um total de 10 indivíduos foram retirados do banco de dados para serem manipulados na fase de teste, a fim de analisar o comportamento do algoritmo de reconhecimento facial para rostos que não constam no banco de dados de treinamento. O banco de dados AT&T possui um total de 30 indivíduos e 10 indivíduos retirados da fase de treino, totalizando 40 indivíduos.

Durante o desenvolvimento do trabalho, optou-se pela criação de um novo banco de dados, composto pela junção dos bancos Yale e AT&T, denominado *Sam*. O banco de dados *Sam* é composto por um total de 500 imagens, todas com dimensões de 92x112 pixels e formato

.pgm. Esse banco de dados possui um total de 50 indivíduos, dos quais 10 foram retirados da fase de treinamento. Foram selecionadas 5 imagens de cada um dos 40 indivíduos restantes para treinamento, totalizando 200 imagens de treinamento. Na fase de teste, foram utilizadas 5 imagens de cada indivíduo, incluindo os 10 que haviam sido retirados para a criação do banco de dados de treinamento, somando um total de 250 imagens para teste. Essa divisão foi realizada para avaliar os algoritmos em situações em que alguns indivíduos não estivessem presentes no banco de dados de treinamento.

4.2 Detectando Faces e Processo de Treinamento

No âmbito da detecção facial, foi empregue o uso de arquivos com a extensão “.xml”, conhecidos como *Haar*. Estes são responsáveis por identificar informações acerca da presença de contrastes presentes nas imagens, e aplicados no processo de detecção e extração, através da implementação do Algoritmo *Viola-Jones* na biblioteca *OpenCV*. Estes arquivos são gerados a partir do treinamento de um conjunto de imagens de amostra, onde o objetivo específico é a captura de faces, sendo estas consideradas exemplos positivos. Por outro lado, uma amostra com imagens aleatórias com diversas formas diferentes de faces são consideradas exemplos negativos, por não corresponderem ao que se pretende.

Em síntese, o arquivo “.xml” conhecido como *Haar* é um arquivo de configuração utilizado pelo algoritmo de detecção de objetos baseado em características *Haar*, o qual utiliza um conjunto de características de *Haar* para detectar objetos em uma imagem. Este arquivo contém informações sobre as características de *Haar* utilizadas pelo algoritmo, como sua localização, tamanho e valores de limiar, gerados a partir do treinamento do algoritmo em um conjunto de imagens de treinamento positivas e negativas.

As bases de imagens utilizadas neste trabalho não apresentam amostras negativas, visto que as imagens de treinamento contêm apenas faces de pessoas. Para este fim, escolheu-se o arquivo “haarcascad-frontalfac-alt2.xml”, disponibilizado no repositório *GitHub*, dada a sua elevada utilização pela comunidade que trabalha com detecção de faces, bem como pelo seu elevado índice de detecção de faces com os conjuntos de banco de dados utilizados.

No processo de treinamento foram utilizados três limitantes diferentes, que variavam conforme o algoritmo. Para o algoritmo *EigenFaces*, os limitantes variavam em 2000, 2500 e 3000, enquanto para o algoritmo *FisherFaces*, os limitantes utilizados foram de 1000, 1200 e 1500. Estas variações de limitantes foram executadas em todos os bancos de dados.

5 RESULTADOS

Durante a fase de testes, o objetivo principal foi encontrar uma configuração que minimizasse ao máximo os falsos positivos, ou seja, reduzisse os casos em que um indivíduo não está presente na base de dados sendo reconhecido erroneamente, e que maximizasse a especificidade, ou seja, a capacidade do sistema em identificar a ausência do indivíduo quando ele realmente não está presente na base de dados.

O desempenho real do sistema é avaliado por meio dos conceitos de especificidade e sensibilidade, representados pelas seguintes fórmulas:

$$E = TN/(TN+FP)$$

onde TN (*True Negative*) corresponde aos verdadeiros negativos e FP (*False Positive*) corresponde aos falsos positivos, e

$$S = TP/(TP+FN)$$

onde TP (*True Positive*) corresponde aos verdadeiros positivos e FN (*False Negative*) corresponde aos falsos negativos na fórmula da sensibilidade.

5.1 EigenFaces

A Figura a seguir apresenta os resultados obtidos a partir da utilização do algoritmo *EigenFaces* em conjunto com as três bases de dados utilizadas neste trabalho. A figura apresenta os valores limites avaliados para cada conjunto de dados AT&T.

Figura 9 – Banco AT&T - *EigenFaces*

Limitante = 2000				Limitante = 2500				Limitante = 3000			
TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
26,5	22,5	0,5	43,5	43	21,5	2	21	55	12	8	11
Especificidade	0,97			Especificidade	0,91			Especificidade	0,6		
Eficiência	0,63			Eficiência	0,72			Eficiência	0,68		
Sensibilidade	0,37			Sensibilidade	0,67			Sensibilidade	0,83		
Acurácia	0,5			Acurácia	0,7			Acurácia	0,73		
Taxa de Acerto	49			Taxa de Acerto	68,5			Taxa de Acerto	76		

Fonte: Autor, 2023

Observa-se que a especificidade do algoritmo de reconhecimento facial apresenta uma redução proporcional ao aumento de falsos positivos, o que indica que o algoritmo não está atingindo resultados satisfatórios. Embora o banco de dados AT&T disponha de uma quantidade

maior de imagens de treinamento, o algoritmo *EigenFaces* não demonstrou ser eficaz na melhoria da taxa de reconhecimento facial. Em relação ao banco de dados Yale, os dados da Figura 10 foram obtidos.

Figura 10 – Banco Yale - *EigenFaces*

Limitante = 2000				Limitante = 2500				Limitante = 3000			
TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
35,6	28,8	1,8	26,6	40	25,5	5,6	25,6	45,5	12,5	12,5	25,5
Especificidade	0,94			Especificidade	0,81			Especificidade	0,5		
Eficiência	0,62			Eficiência	0,77			Eficiência	0,55		
Sensibilidade	0,57			Sensibilidade	0,66			Sensibilidade	0,64		
Acurácia	0,69			Acurácia	0,66			Acurácia	0,61		
Taxa de Acerto	67,62			Taxa de Acerto	66,65			Taxa de Acerto	60		

Fonte: Autor, 2023

Pode-se observar que à medida que os limites aumentam, a taxa de falsos positivos também aumenta, o que resulta em uma baixa taxa de especificidade. Este resultado não é favorável, uma vez que mesmo com o limite em seu menor valor, a especificidade e os falsos positivos não são satisfatórios. É importante destacar que o valor da especificidade inicia em 0,94 e sofre uma queda até 0,5. Para o banco de dados *Sam*, apresentamos a Figura 11 a seguir.

Figura 11 – Banco *Sam* - *EigenFaces*

Limitante = 2000				Limitante = 2500				Limitante = 3000			
TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
24,5	29	0	42,3	39,8	26,5	1,45	27,7	50,62	25,45	4,6	17,9
Especificidade	1			Especificidade	0,94			Especificidade	0,84		
Eficiência	0,67			Eficiência	0,73			Eficiência	0,78		
Sensibilidade	0,36			Sensibilidade	0,58			Sensibilidade	0,58		
Acurácia	0,66			Acurácia	0,66			Acurácia	0,74		
Taxa de Acerto	54,22			Taxa de Acerto	68,01			Taxa de Acerto	76,02		

Fonte: Autor, 2023

O Banco de imagens denominado *Sam* possui um número significativo de imagens, o que permite avaliar o desempenho do algoritmo de reconhecimento facial sob diferentes parâmetros. Observa-se que, para o primeiro limite adotado, o algoritmo apresenta uma especificidade satisfatória e não há ocorrência de falsos positivos. No entanto, a taxa de acerto do algoritmo é baixa, atingindo apenas 54,22%. Com o aumento do limite, a taxa de acerto pode variar e alcançar um valor de até 76,02%. Entretanto, a especificidade e a taxa de falsos positivos também sofrem alterações, tornando o algoritmo não satisfatório. Em resumo, conclui-se que, para as três bases de dados analisadas, o algoritmo *EigenFaces* não atinge valores válidos de especificidade e falsos positivos.

5.2 FisherFaces

Utilizaremos agora o algoritmo *FisherFaces* para realizar os mesmos testes, tendo em vista que essa técnica é mais robusta, portanto, deve apresentar resultados mais confiáveis em relação às classificações. A figura 12 apresenta os resultados obtidos para a base de dados AT&T.

Figura 12 – Banco AT&T - *FisherFaces*

	Limitante = 1000				Limitante = 1200				Limitante = 1500			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
	34	22,5	0,5	33	45	20,5	4	22,5	60	10	20	9,5
Especificidade	0,97				Especificidade	0,83			Especificidade	0,33		
Eficiência	0,72				Eficiência	0,74			Eficiência	0,59		
Sensibilidade	0,4				Sensibilidade	0,47			Sensibilidade	0,86		
Acurácia	0,6				Acurácia	0,7			Acurácia	0,7		
Taxa de Acerto	59,8				Taxa de Acerto	71,3			Taxa de Acerto	70		

Fonte: Autor, 2023

Diferentemente dos resultados obtidos pelo algoritmo *EigenFaces*, o *FisherFaces* apresenta falsos positivos já no primeiro valor limitante, além de uma especificidade menor que 1.0, comprovando sua eficácia para essa base de dados. No entanto, é necessário observar os valores limitantes, em que são menores no *FisherFaces*. A taxa de acertos também é baixa, variando entre um mínimo de 59,8% e um máximo de 70%, o que é insuficiente para a construção de um sistema de reconhecimento facial. A figura 13 a seguir mostra os resultados obtidos para a base de dados Yale.

Figura 13 – Banco Yale - *FisherFaces*

	Limitante = 1000				Limitante = 1200				Limitante = 1500			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
	23,7	33	0	37	33,33	33,33	0	33,33	39,9	25,5	5,75	23
Especificidade	1				Especificidade	1			Especificidade	0,81		
Eficiência	0,72				Eficiência	0,75			Eficiência	0,7		
Sensibilidade	0,47				Sensibilidade	0,5			Sensibilidade	0,52		
Acurácia	0,4				Acurácia	0,6			Acurácia	0,7		
Taxa de Acerto	60				Taxa de Acerto	66,66			Taxa de Acerto	68,88		

Fonte: Autor, 2023

Ao contrário do algoritmo *EigenFaces*, o *FisherFaces* consegue obter uma especificidade de 1.0 sem falsos positivos, mesmo aumentando os valores limitantes. Somente quando o valor limitante chega ao seu máximo é que surgem os falsos positivos, e a especificidade cai de 1.0 para 0.81. Esse valor não é tão drástico quanto o observado no *EigenFaces*. Destaca-se que o maior valor limitante do *FisherFaces* é menor do que o menor valor limitante do *EigenFaces*, aumentando nossa confiança nos valores obtidos. A taxa de acertos, no entanto, não ultrapassa

os 66.66%.

Figura 14 – Banco Sam - *FisherFaces*

Limitante = 1000				Limitante = 1200				Limitante = 1500			
TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
25,3	26,5	0	42,2	42,2	28,5	0	26,3	54,54	18,04	10,12	15
Especificidade	1			Especificidade	1			Especificidade	0,64		
Eficiência	0,65			Eficiência	0,72			Eficiência	0,7		
Sensibilidade	0,38			Sensibilidade	0,52			Sensibilidade	0,54		
Acurácia	0,5			Acurácia	0,7			Acurácia	0,73		
Taxa de Acerto	56,3			Taxa de Acerto	69,8			Taxa de Acerto	72,23		

Fonte: Autor, 2023

A figura 14 apresenta os resultados obtidos para a base de dados Sam, o qual é uma junção das bases de dados AT&T e Yale. É possível obter uma taxa de acertos de cerca de 70% com especificidade 1.0 e nenhum falso positivo com o limite em 1200. Esse é o cenário no qual se obteve a maior taxa de acerto, mantendo-se as necessidades de um sistema de não reconhecer um indivíduo que não esteja no banco de dados. Com o aumento dos valores limitantes, é provável que o algoritmo perca sua especificidade, gerando um alto número de falsos positivos e reduzindo a eficiência do algoritmo. Conclui-se, portanto, que o algoritmo *FisherFaces* mostrou-se mais eficaz do que o algoritmo *EigenFaces* para o reconhecimento das bases de dados utilizadas.

6 CONCLUSÃO

Considerando os testes realizados, é evidente que as imagens das bases de dados foram fundamentais para o sucesso dos algoritmos testados. Isso porque, mesmo utilizando o mesmo algoritmo, os resultados variaram consideravelmente ao alterar-se as bases de dados. Outro ponto importante a ser considerado é que os algoritmos utilizados, como Eigenfaces e Fisherfaces, apesar de possuírem técnicas interessantes, são antiquados em relação às técnicas mais atuais de reconhecimento facial. Dessa forma, tais algoritmos possuem limitações no que se refere ao escopo de aplicações reais.

No caso do algoritmo Eigenfaces, percebe-se que os classificadores apresentaram confiança bastante reduzida, além de muitos falsos positivos e uma baixa taxa de especificidade, o que se afasta do objetivo almejado pelos testes. Se considerarmos a possibilidade de uma aplicação prática, a probabilidade de falhas é elevada, considerando-se a baixa taxa de acertos e os falsos positivos inaceitáveis para tal contexto.

No que diz respeito ao algoritmo Fisherfaces, nota-se que sua robustez é mais acentuada, o que se reflete na certeza que o algoritmo apresenta em suas respostas. Além disso, seus valores mais baixos de thresholds mantêm uma taxa de acerto aceitável, com mínimos falsos positivos e baixa especificidade.

Diante disso, pode-se considerar que uma solução para aprimorar as taxas de acerto consiste na utilização de algoritmos mais recentes, como o LBPH, que utiliza a fragmentação da imagem para realizar o reconhecimento facial, bem como o uso de redes neurais com deep learning. Portanto, em trabalhos futuros, a utilização de tais técnicas será inevitável e essencial no campo do reconhecimento facial.

REFERÊNCIAS

- ALBUQUERQUE, M. P. de; ALBUQUERQUE, M. P. de. Processamento de imagens: métodos e análises. **Centro Brasileiro de Pesquisas Físicas-CBPF/MCT**, 2000.
- ALVES, J. F.; REGIS, C. D. M.; CORREIA, S. E. Detecção de cores para pessoas com deficiência visual utilizando opencv.
- ANTONELLO, R. Introdução a visão computacional com python e opencv. **Engenharia de Controle e Automação do Instituto Federal Catarinense-IFC, campus Luzerna**, 2014.
- BACKES, A. R.; JUNIOR, J. J. d. M. S. **Introdução a Visão Computacional Usando Matlab**. [S.l.]: Alta Books Editora, 2019.
- BISSI, T. *et al.* Reconhecimento facial com os algoritmos eigenfaces e fisherfaces. Universidade Federal de Uberlândia, 2018.
- COSTA, L. R.; OBELHEIRO, R. R.; FRAGA, J. S. Introdução à biometria. **Livro-texto de Minicursos-VI SBSeg**, 2006.
- ELEYAN, A.; DEMIREL, H. Pca and lda based face recognition using feedforward neural network classifier. In: SPRINGER. **International Workshop on Multimedia Content Representation, Classification and Security**. [S.l.], 2006. p. 199–206.
- GALIMBERTI, L. H. d. O. **Estudo comparativo de algoritmos de biometria facial disponibilizados pela biblioteca OpenCV para controle de acesso**. Dissertação (B.S. thesis), 2019.
- LOPES, E. C.; FILHO, J. C. B.; NO, R. T. Detecção de faces e características faciais. **Porto Alegre: PUCRS**, 2005.
- MARENGONI, M.; STRINGHINI, S. Tutorial: Introdução à visão computacional usando opencv. **Revista de Informática Teórica e Aplicada**, v. 16, n. 1, p. 125–160, 2009.
- MILANO, D. de; HONORATO, L. B. **Visão computacional**. 2014.
- NORONHA, F. de A.; ALMEIDA, L. L. de; SILVA, F. A. da; CABRAL, F. P. A.; SISCOOTTO, R. A. Detecção de fadiga a partir da análise de imagens faciais. In: **Colloquium Exactarum. ISSN: 2178-8332**. [S.l.: s.n.], 2019. v. 11, n. 2, p. 34–45.
- ORSO, L. R.; FILASTRO, M. d. O.; MICHELS, C.; MADEIRA, K.; CORAL, S.; MARTINS, P. J. Estudo de caso de um sistema de reconhecimento facial utilizando o openface para identificação de faces em bancos de imagens. **Revista de Sistemas e Computação-RSC**, v. 10, n. 1, 2020.
- SAHA, R.; BHATTACHARJEE, D. Face recognition using eigenfaces. **International Jpournal of Emerging Technology and Advanced Engineering**, Citeseer, v. 3, n. 5, 2013.
- SANTOS, R. S. dos; FABRIS, J. P. Tecnologias biométricas de controle de acesso. In: **10th International Symposium on Technological Innovation**. [S.l.: s.n.], 2019.
- SILVA, A. B. N.; MARROQUIM, R. Reconhecimento facial utilizando eigenfaces. **Universidade Federal do Rio de Janeiro-UFRJ, Rio de Janeiro, RJ**, 2013.

SILVA, V.; NUNES, E. M.; ROCHA, A. M. O reconhecimento facial como uma das vertentes da inteligência artificial (ia): um estudo de prospecção tecnológica. **Cadernos de Prospecção**, 2020.

SIMOES, T. V. Sistema integrado de análise forense de dispositivos de armazenamento para reconhecimento visual automático. **Tese de Doutorado**, 2020.