UNIVERSIDADE ESTADUAL DO CEARÁ

CENTRO DE CIÊNCIAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

GISELLY SOARES DE SOUSA DAMASCENO

RECONHECIMENTO FACIAL COM VARIAÇÕES DE ILUMINAÇÃO UTILIZANDO PCA E MODIFICAÇÕES DA DCT ASSOCIADAS AOS CLASSIFICADORES GMM, NAÏVE BAYES E K-NN

FORTALEZA – CEARÁ

2017

GISELLY SOARES DE SOUSA DAMASCENO

RECONHECIMENTO FACIAL COM VARIAÇÕES DE ILUMINAÇÃO UTILIZANDO PCA E MODIFICAÇÕES DA DCT ASSOCIADAS AOS CLASSIFICADORES GMM, NAÏVE BAYES E K-NN

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientador: Thelmo Pontes de Araujo

FORTALEZA – CEARÁ

2017

GISELLY SOARES DE SOUSA DAMASCENO

RECONHECIMENTO FACIAL COM VARIAÇÕES DE ILUMINAÇÃO UTILIZANDO PCA E MODIFICAÇÕES DA DCT ASSOCIADAS AOS CLASSIFICADORES GMM, NAÏVE BAYES E K-NN

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Aprovada em:

BANCA EXAMINADORA

Prof. Thelmo Pontes de Araujo, Ph.D. (Orientador)
Universidade Estadual do Ceará – UECE

Prof. Dr. José Everardo Bessa Maia
Universidade Estadual do Ceará - UECE

Prof. Dr. Iális Cavalcante de Paula Júnior
Universidade Federal do Ceará - UFC

Ao meu filho **Pedro**, que apesar de seus somente 6 meses de idade, já me proporcionou os melhores 6 meses da minha vida.

AGRADECIMENTOS

Agradeço primeiramente a Deus pelo dom da vida e da sabedoria, pelo seu conforto, pela sua proteção, e por ser meu maior companheiro nos momentos de dificuldades, alegrias, derrotas e vitórias.

Aos meus pais pelo amor incondicional, pelo apoio constante, pela confiança, pela educação e por toda formação durante a minha jornada da vida.

Ao meu esposo Filipe pela sua imensurável paciência e momentos de carinho, amor, inspiração, compreensão e felicidade.

À minha irmã Pamella pelo seu amor, incetivo e por ser minha grande companheira de todas as horas.

Ao meu orientador, Prof. Thelmo de Araujo, pela confiança, incentivo, acessibilidade e pela sua excelente orientação. Agradeço-lhe ainda pela simpatia, simplicidade e amizade que me ofereceu.

Ao Prof. Dr. José Everardo por ter me proporcionado grandes conhecimentos nas suas disciplinas. Conhecimentos estes que foram de fundamental importância para realização deste trabalho.

À Capes por todo o apoio durante o mestrado.

Aos meus colegas de mestrado Amanda Souza, Anderson Couto, Janaide Nogueira, Marcelo Casademunt, Marcos Borges, Robson Oliveira e Vanessa Vasconcelos pelo companheirismo e alegrias, tristezas e sabedorias compartilhadas.

À Nina por me proporcionar momentos de alegrias.

Enfim, a todas as pessoas que de alguma forma contribuíram para a realização e conclusão do meu mestrado.

RESUMO

O reconhecimento facial é a biometria mais estudada no últimos anos, com aplicações em diversos campos tais como reconhecimento de padrões, processamento de sinal e visão computacional. Há diversos fatores que dificultam a obtenção de uma acurácia ótima: ruídos, expressões faciais, presença de óculos, barba e a variação de iluminação. A fim de lidar com um dos principais fatores da redução da acurácia, a variação de iluminação, este trabalho realiza um reconhecimento facial utilizando variações de métodos de extração de características associados a classificadores populares nas literaturas de reconhecimento de padrões. A extração de características das faces foram realizadas através dos métodos Análise de Componentes Principais (PCA), Transformada Discreta do Cosseno (DCT), e as variações DCT-mod, DCT-mod-delta e DCT-mod2. Para a classificação, foram utilizados os classificadores Modelo de Misturas Gaussianas (GMM), Naïve Bayes e K-Vizinhos mais Próximos (K-NN). O desempenho dos algoritmos foi analisado utilizando a base de imagem VidTIMIT, que possui imagens contendo expressões e poses, e, para analisar o desempenho dos métodos em imagens com grandes variações de iluminação, foram aplicadas nas mesmas várias iluminações artificiais. Os resultados mostraram que os melhores métodos de extração de características foram o DCT-mod-delta e DCT-mod2 em todos os classificadores, sendo o método DCT-mod-delta o que obteve a melhor acurácia quando associado ao classificador K-NN utilizando a medida de correlação, com 97% em imagens sem nenhuma variação de iluminação e com 92,6% em imagens com variações de iluminação extrema.

Palavras-chave: Reconhecimento facial. Análise de Componentes Principais. Modelo de Misturas Gaussianas. Naïve Bayes.

ABSTRACT

Facial recognition is biometrics most studied in recent years, with applications in various fields, such as pattern recognition, signal processing and computer vision. There are several factors that make it difficult to obtain optimum accuracy: noises, facial expressions, presence of glasses, beards and lighting variation. In order to deal with one of the main factors of accuracy reduction, the variation of illumination, this work performs facial recognition using variations of feature extraction methods associated with popular classifiers in pattern recognition literatures. The extraction of face characteristics was performed using the Principal Component Analysis (PCA), Discrete Cosine Transform (DCT) methods, and DCT-mod, DCT-mod-delta and DCT-mod2 variations. For the classification, the Gaussian Mixture Model (GMM), Naïve Bayes and Nearest K-Neighbors (K-NN) were used. The performance of the algorithms was analyzed using the VidTIMIT image base, which has images containing expressions and poses, and to analyze the performance of the methods in images with large variations of illumination, several artificial illumination were applied in the same. The results showed that DCT-mod-delta and DCT-mod2 were the best methods for extracting characteristics in all classifiers, with the DCT mod-delta method obtaining the best accuracy when associated to the KNN classifier using the correlation distance, with 97% in images with no illumination variation and with 92.6% in images with extreme illumination variations.

Keywords: Facial recognition. Principal Component Analysis. Gaussian Mixture Model. Naïve Bayes.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AC	Alternate Current
BPSO	Binary Particle Swarm Optimization
CA	Coeficiente de Aproximação
DC	Direct Current
DCT	Discrete Cosine Transform
DWT	Discrete Wavelets Transform
GMM	Gaussian Mixture Models
HMMs	Hidden Markov Model
K-NN	K-Nearest Neighbors
LDA	Linear Discriminant Analysis
PCA	Principal Component Analysis
NNs	Neural Networks
SVM	Suport Vector Machine

SUMÁRIO

1 INTRODUÇÃO

Sistemas baseados em biometria reconhecem padrões capazes de identificar um indivíduo através de suas características exclusivas, como assinatura, impressão digital, voz, íris, face, etc. Nesses sistemas, as informações biométricas das pessoas são regitradas em uma base de dados biométricos, podendo ser submetidos a algum tipo de pré-processamento. Por exemplo, em imagens, a equalização do histograma é capaz de atenuar diferenças acentuadas de iluminação. Para dados de áudio, a separação da voz do som ou ruído de fundo também é um pré-processamento (BUCIU; GACSADI, 2016).

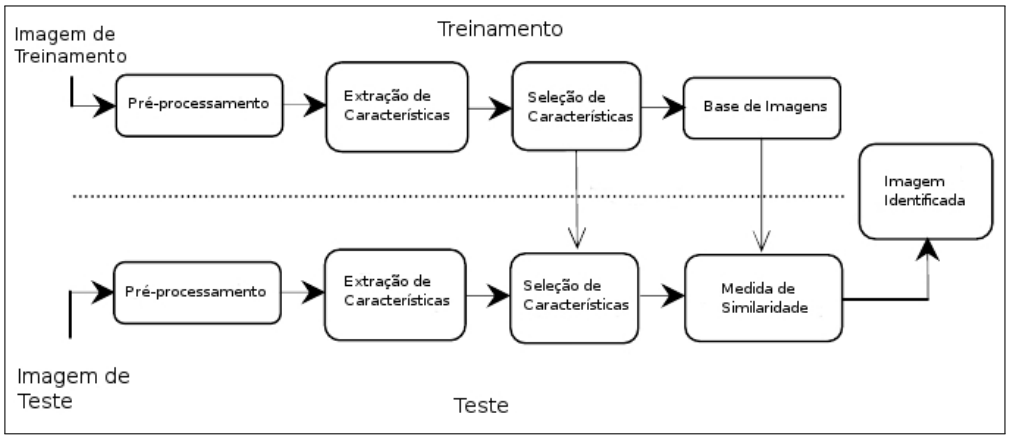
A maioria dos métodos confiáveis de reconhecimento baseados em características biométricas necessita de precisão e a cooperação do indivíduo, que precisa se submeter voluntariamente para à captura dos dados (ZHAO et al., 2003). Considerando todas essas exigências, o reconhecimento baseado em faces é menos invasivo. Além dessa vantagem, é a biometria mais estudada nos últimos anos, apresentando repercussões sobre diversos campos tais como reconhecimento de padrões, processamento de sinais e visão computacional (KODANDARAM et al., 2015), pois é uma biometria de difícil implementação por seu reconhecimento ser afetado por diversos fatores presentes na imagem.

Uma das principais etapas no reconhecimento facial é a extração de informações importantes a partir de uma imagem da face. Os desafios surgem principalmente devido a fatores como pose, expressões faciais, envelhecimento, oclusões e o principal, a variação de iluminação (GOMATHI; BASKARAN, 2014). Devido a esses desafios, muitos algoritmos diferentes foram desenvolvidos para extração de características e redução de dimensionalidade dos dados tais como Análise de Componentes Principais (PCA) (TURK; PENTLAND, 1991) (BELHUMEUR et al., 1997), Análise Discriminante Linear (LDA) (ZHAO et al., 1998), Transformada Discreta do Cosseno (DCT) (JING; ZHANG, 2004), Modelos Oculto de Markov (HMMs) (OTHMAN; ABOULNASR, 2003), Redes Neurais (NNS) (ER et al., 2002) (ER et al., 2005), Máquina de Vetores de Suporte (SVM) (LEE et al., 2002), etc.

A Figura 1 mostra um diagrama em blocos básico de um sistema de reconhecimento facial. Na primeira etapa é realizado um pré-processamento para melhorar a qualidade da imagem criando uma base para uma extração de características eficiente. A etapa de extração de características desempenha um papel importante no processo de reconhecimento, pois seleciona as melhores características discriminantes menos sensíveis a variação na pose, expressões faciais e variações de iluminação, reunindo-as em um vetor de características para a sua representação.

A etapa de seleção de características é importante em situações onde o conjunto de característica é grande e deseja-se selecionar um subconjunto adequado. Na maioria dos sistemas de reconhecimento facial essa etapa é importante por eliminar características irrelevantes e reduzir a complexidade computacional sem comprometer a precisão da classificação. Essa etapa nem sempre acontece para todos os reconhecimento faciais. Na fase de teste, o vetor de características da imagem de teste é comparado com cada um dos vetores características contido na base de treinamento, portanto, se houver similaridade entre os vetores característicos comparados, a imagem é reconhecida pelo sistema. A seleção de características é realizada através de um algoritmo de seleção adequado e a medida de similaridade é feita por meio de um classificador apropriado (KODANDARAM et al., 2015).

Figura 1 – Diagrama em blocos de um sistema de reconhecimento facial.



Fonte – (KODANDARAM et al., 2015)

O vetor de características é o principal fator no reconhecimento facial. As características que o compõe podem ser obtidas através de três métodos: os locais, que determinam as características individuais da face e suas relações geométricas, como olhos, nariz e boca, assim como suas medidas de distâncias e ângulos; os holísticos, que analisam a face como um todo sem se preocupar em identificar características isoladas, usando informações dos pixels; e os híbridos, que é uma combinaçao dos métodos anteriores (ZHAO et al., 2003).

Apesar de muito estudada, a biometria facial ainda continua sendo um desafio, um dos principais problemas encontrados é a sensibilidade do reconhecimento facial a grandes variações de iluminação. A fim de encontrar soluções para este problema, o presente trabalho implementa um reconhecimento facial utilizando os métodos PCA, DCT e suas variações DCT-

mod, DCT-mod-delta e DCT-mod2 para extração de características das faces e redução de dimensionalidade associados aos classificadores GMM, Naïve Bayes e KNN por meio do método holístico. Uma análise comparativa é realizada a fim de encontrar a melhor combinação para um reconhecimento facial robusto a imagens com grandes variações de iluminação.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Este trabalho tem como principal objetivo implementar um sistema de reconhecimento facial utilizando os métodos de extração de características PCA, DCT e suas variações DCT-mod, DCT mod-delta e DCT-mod2 e os classificadores GMM, Naïve Bayes e K-NN, realizando uma análise comparativa da sua robustez em imagens monocromáticas frontais com grandes variações de iluminação.

1.1.2 Objetivos Específicos

Dentre os objetivos específicos, destacam-se os seguintes:

a) Implementar os métodos de extração de características PCA, DCT e suas variações DCT-mod, DCT mod-delta e DCT-mod2;
b) Implementar as técnicas de classificação para o reconhecimento de faces (GMM, Naïve Bayes e K-NN);
c) Realizar os experimentos com a base de imagens VidTIMIT (SANDERSON; LOVELL, 2009) que apresentam variações de expressões e poses;
d) Analisar a robustez da metodologia proposta no reconhecimento de faces com efeitos de iluminação artificial por meio das medidas de avaliação: acurácia total, erro e índice *kappa*.

1.2 TRABALHOS RELACIONADOS

Vários métodos já foram implementados na literatura de reconhecimento facial visando a encontrar as melhores formas de um reconhecimento robusto a diversos fatores como pose, expressões faciais, oclusões e variações de iluminação. Nesta seção, apresentamos alguns trabalhos relacionados que servirão de base para a metodologia de reconhecimento facial deste

trabalho, bem como ajudar em uma análise comparativa da metodologia usada com os métodos já existentes.

Conrad Sanderson (SANDERSON, 2008) apresenta uma metodologia que realiza a fusão de uma verificação facial com verificação de voz. Para a verificação das faces frontais, Sanderson propôs três tipos de extração de características, denominadas DCT-mod, DCT-mod-delta e DCT-mod2. Esses três métodos usam coeficientes polinomiais derivados de coeficientes DCT bidimensionais de blocos vizinhos. A robutez e a performance desses métodos foram comparadas com três métodos populares da literatura (PCA, DCT e *Wavelets* de Gabor), aplicando mudanças na direção de iluminação nas imagens. Os experimentos foram realizados com sua própria base de imagens VidTIMIT, e os resultados mostraram que seus métodos DCT-mod, DCT-mod-delta e DCT-mod2 são mais robustos do que as *wavelets* de Gabor, os coeficientes DCT padrão e PCA (com ou sem equalização de histograma). Sanderson apresenta em seus resultados que os métodos Wavelets de Gabor, coeficientes DCT e PCA sofrem grandes quedas na acurácia à medida que a variação de iluminação é aplicada na imagem, enquanto, em seus métodos a acurácia permanece razoavelmente inalterada frente a variações de iluminação nas imagens. As três variações de DCT foram robustas a grandes mudanças de iluminação, mas o método DCT-mod2 foi o que obteve a menor taxa de erro, de aproximadamente 2%. Para a classificação das faces Sanderson utilizou o classificador GMM (do inglês, *Gaussian Mixture Models*) com 8 gaussianas.

Vaidehi e colegas (VAIDEHI et al., 2010) implementaram um reconhecimento facial com uma alta taxa de reconhecimento, bem com uma boa robustez em imagem com alterações de iluminação. O reconhecimento é realizado com os métodos Transformada Discreta do Cosseno (DCT), Discriminante Linear de Fisher (FLD) e o classificador dos K vizinhos mais próximos (K-NN) na base de imagens FERET. Primeiramente, é feita a redução de dimensionalidade da face utilizando o método DCT, após obtidos os coeficientes DCT, Vaidehi descarta os primeiros coeficientes por serem os mais afetadas quando há iluminação na imagem. Em seguida, FLD é aplicado aos coeficientes selecionados para discriminar as características faciais invariantes. Por fim o classificador K-NN é usado para reconhecimento das faces no conjunto de dados extraídos do FLD. A dimensionalidade dos dados foi muito reduzida através da DCT, usando somente 5% do número total de coeficientes (isto é, 50 dos 1000 coeficientes) e, o classificador K-NN realizou a classificação das imagens de teste mais facilmente devido ao método FLD, que tem como função aumentar separação entre as classes e diminuir dentro delas, além de extrair as

características mais importantes. A análise do desempenho foi realizada para 100 pessoas com 10 poses para cada uma e a taxa de reconhecimento foi de 97%.

Shermina (SHERMINA, 2011) desenvolveu um eficiente sistema de reconhecimento facial invariante a iluminação usando a Transformada Discreta do Cosseno (DCT) e a Análise de Componentes Principais (PCA). Para processar a imagem invariante a iluminação, os coeficientes DCT de baixa frequência são usados para normalizar a imagem iluminada, os coeficientes DCT ímpares e pares são usados para compensar a variação da iluminação. Por fim, o método PCA é usado para o reconhecimento das imagens faciais. A variação de iluminação pode ser facilmente compensada com base nos coeficientes DCT ímpares e pares devido à propriedade simétrica da face. Inicialmente, duas novas imagens são criadas a partir dos coeficientes ímpares e pares da direção horizontal das imagens originais. Em seguida, os *pixels* da metade esquerda com a metade direita são comparados uns com os outros, se o *pixel* do lado direito for positivo, mas o *pixel* correspondente ao lado esquerdo for negativo, ambos os valores de *pixel* são ajustados. A proposta é validada com a base de imagens Yale Face Database B, onde obteve uma acurácia de 94,2%, com 5,84% de falsa aceitação (FAR) e 7,51% de falsa rejeição (FRR), provando assim, que a técnica DCT com PCA gera uma boa taxa de reconhecimento em imagens com a iluminação.

1.3 ORGANIZAÇÃO DO TRABALHO

No Capítulo 2, são apresentados as conceituações e técnicas aplicadas em reconhecimento facial utilizados neste trabalho. O Capítulo 3 apresenta o detalhamento dos experimentos bem como a base de imagens VidTIMIT. O capítulo posterior apresenta a análise dos resultados e discussões segundo as metodologias aplicadas. E, por fim, o Capítulo 5 expõe as conclusões obtidas deste trabalho e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

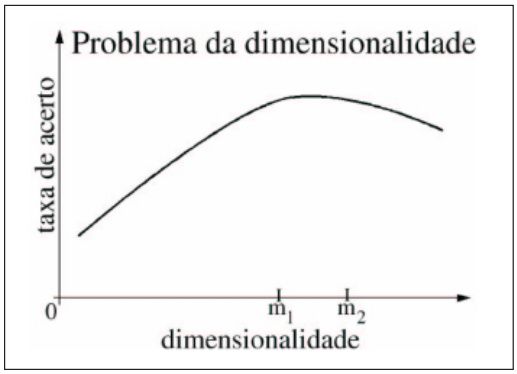
Este capítulo expõe o conhecimento teórico necessário para compreensão deste trabalho. Inicialmente, é apresentado o conceito de redução de dimensionalidade. Em seguida, extração e seleção de características são apresentadas, seguidas dos dois métodos de extração de características utilizados. E, por fim, os classificadores utilizados são apresentados.

2.1 REDUÇÃO DE DIMENSIONALIDADE

Dimensionalidade refere-se ao número de características de uma representação, ou seja, a dimensão do espaço de características (atributos). Há duas razões principais para que a dimensionalidade seja a menor possível: o custo de medição e a precisão do classificador.

A redução de dimensionalidade é necessária para evitar o problema da dimensionalidade que afeta a precisão de um classificador (MARTINS, 2004). O problema da dimensionalidade acontece quando a quantidade de amostras de treinamento para que um classificador obtenha um bom desempenho é uma função monotonicamente crescente da dimensão dos padrões (número de características) (JAIN et al., 2000). Em poucos casos, pode-se mostrar que essa função é exponencial, pois, em reconhecimento de padrões a quantidade de amostras necessárias para a classificação cresce exponencialmente com a dimensionalidade (PERLOVSKY, 1998). A Figura 2 apresenta o comportamento da taxa de acerto de um classificador com o aumento da dimensão do espaço de características.

Figura 2 – Problema da Dimensionalidade.



Fonte – Figura retirada de (CAMPOS, 2001)

Na Figura 2, onde a dimensionalidade está compreendida entre 0 e m_1, a taxa de

acerto é diretamente proporcional à dimensionalidade, pois, ao adicionar novas características, o desempenho do classificador melhora. Isso deve-se ao fato de espaços com dimensões muito pequenas não possuírem informações suficientes para distinguir-se as classes de padrões (CAMPOS, 2001).

Na segunda faixa da Figura 2, onde a dimensionalidade está entre m_1 e m_2, o aumento da dimensionalidade não altera (ou altera sutilmente) a precisão do classificador. Contudo, características redundantes ou irrelevantes ao problema também são processadas gerando um desperdício de recursos e aumentando o custo de medição (CAMPOS, 2001).

Na terceira faixa da Figura 2, onde a dimensionalidade é maior que m_2, a adição de características prejudica o desempenho da classificação devido a quantidade insuficiente de amostras em relação à quantidade de características, gerando uma redução na taxa de acerto, ou seja, o desempenho do algoritmo tende a degradar-se, causando o problema da dimensionalidade. (CAMPOS, 2001) (MARTINS, 2004).

A redução de dimensionalidade ajuda na retirada de dados irrelevantes e redundantes, pois estes dados influenciam na precisão e no custo da classificação. Características irrelevantes são aquelas que não possuem informação útil para o problema, já as características redundantes possuem a mesma informação útil para o problema, por exemplo, dois atributos contendo os mesmo valores para cada instância.

O problema da dimensionalidade está sempre presente no reconhecimento facial, pois cada pixel da imagem, que é uma característica da face, é, em princípio, importante, e geralmente as matrizes que representam as faces possuem grande dimensão (SANDMANN et al., 2002). O desafio é encontrar um conjunto menor de características que, ainda assim, possa identificar de forma exclusiva uma face. Porém, um reduzido número de características pode levar a uma fraca discriminação e, consequentemente, a uma precisão inferior no sistema de reconhecimento resultante. Toda redução de dimensionalidade implica uma perda de informação, e isto pode vir a ser fundamental para discriminação das faces. Por isso, o objetivo principal das técnicas de redução de dimensionalidade é preservar o máximo possível da informação relevante dos dados. Isso pode ser feito por meio das técnicas de extração e seleção de características.

2.2 EXTRAÇÃO E SELEÇÃO DE CARACTERÍSTICAS

Seleção de características se refere a técnicas que procuram selecionar o melhor subconjunto de um conjunto de características de entrada. Já a extração de características consiste

de métodos que criam novas características a partir de transformações ou combinações das características originais. Geralmente, a extração precede a seleção, pois inicialmente as características são extraídas a partir dos dados de entrada e em seguida, algumas das características extraídas com baixa poder de discriminação são descartadas (CAMPOS, 2001).

Na extração de características, escolhe-se uma transformação a ser aplicada nos dados de tal forma que haja uma alta concentração de informação em poucas características, e que a redundância nos dados seja reduzida. A geração de novas características pode ser feita por meio de transformações lineares. São exemplos de técnicas de extração de características: Análise de Componentes Principais, Transformadas Discretas de *Wavelets*, Redes Neurais, Transformada de Fourier, Transformada Discreta do Cosseno, etc.

Existem inúmeras técnicas para a seleção de características, sendo estas categorizadas como métodos dependentes do modelo (*Model Based*) e métodos independentes do modelo (*Model-Free*). Dentre os métodos dependentes do modelo pode-se mencionar técnicas baseadas em redes neurais, em modelos neurofuzzy e em algoritmos genéticos. No caso dos métodos independentes do modelo há métodos estatísticos, análise de componentes principais, correlação e entropia. Cada tipo de técnica tem suas próprias características, apresentando vantagens e desvantagens (CONTRERAS, 2002).

2.2.1 Análise de Componentes Principais

Segundo Jolliffe (JOLLIFFE, 2002), a ideia central da Análise de Componentes Principais (do inglês, *Principal Component Analysis*, PCA) é reduzir a dimensionalidade de um conjunto de dados que consiste de um grande número de variáveis inter-relacionadas. Isto é obtido através da transformação dos dados originais em um novo conjunto de variáveis, chamadas componentes principais, que são não correlacionadas e organizadas de forma que as primeiras componentes contêm a maior parte da variância contida no conjunto de dados original. Matematicamente, as componentes principais são os autovetores associados aos autovalores de uma matriz de covariância, e, que as maiores variâncias são os maiores autovalores, isto está demonstrado no Apêndice A.

2.2.2 Transformada Discreta do Cosseno

A Transformada Discreta do Cosseno (do inglês, *Discrete Cosine Transform*, DCT) foi apresentada por Ahmed et al. em 1974, e desde então tem sido amplamente explorada pela

comunidade de processamento de sinais, processamento de imagens, principalmente nas áreas de compressão, filtragem e extração de características (PEDRINI; SCHWARTZ, 2008).

A DCT é uma função linear e invertível, $\mathbb{R} \to \mathbb{R}$, que apresenta sinais como soma de funções de cossenos. A DCT leva o sinal original do domínio temporal para o domínio da frequência, podendo ser convertido de volta para o domínio do tempo pela da aplicação da DCT inversa.

Quando o sinal é convertido para o domínio da frequência obtemos os coeficientes DCT, que informam a importância das frequências presentes no sinal original. Os coeficientes DCT podem ser agrupados em duas faixas de frequência: as frequências mais baixas e as frequências mais altas. As frequências mais baixas estão contidas nos primeiros (início) coeficientes DCT, apresentando o comportamento geral do sinal (informações mais importantes). Já as frequências mais altas estão nos últimos coeficientes DCT, representando informações mais detalhadas ou finas do sinal, onde em muitos casos consistem predominantemente de ruídos (GONZALEZ; WOODS, 2008). Assim, após a aplicação da DCT, os coeficientes de frequências mais baixas são os mais apropriados para representar os diferentes padrões, no caso deste trabalho, as diferentes faces dos indivíduos. Sendo também considerada uma redução de dimensionalidade.

Há quatro definições para DCT: DCT-I, DCT-II, DCT-III e DCT-IV. Sendo a DCT-II a mais utilizada em processamento de sinais e de imagens, que também possui uma forte capacidade de compactação de energia e, muitas das informações do sinal tendem a se concentrar em poucas componentes de baixas frequência (MATOS, 2008). A DCT-II é definida por:

$$F(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{a-1} \sum_{y=0}^{b-1} f(x,y) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}, \tag{2.1}$$

sendo

$$\alpha(u)\alpha(v) = \begin{cases} \sqrt{\dfrac{1}{N}}, \text{ se } u,v = 0 \\ \sqrt{\dfrac{2}{N}}, \text{ caso contrário} \end{cases}$$

Na Equação 2.1, a matriz da imagem original é representada por $f(x,y)$, onde a e b são as dimensões da imagem, com $a \times b = N$. A DCT-II gera uma matriz $F(u,v)$ que contém os coeficientes DCT, também de dimensão $a \times b$ quando aplicada na imagem completa. As variáveis

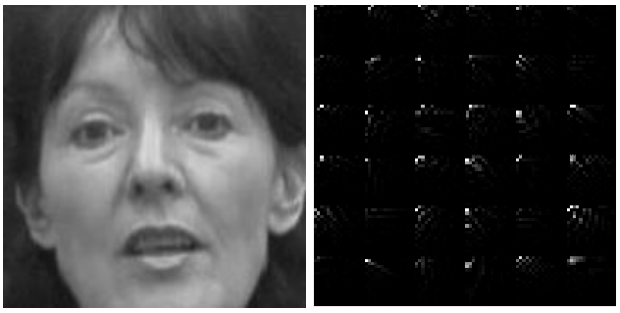
x e y representam as coordenadas no domínio espacial e u e v representam as coordenadas no domínio da frequência.

O brilho da imagem é representado pelo primeiro coeficiente $F(1,1)$, conhecido como coeficiente DC (*Direct Current*). O restante dos coeficientes $F(u,v)$ representam a amplitude correspondente a componente da frequência de $f(x,y)$ e são conhecidos como coeficientes AC (*Alternate Current*).

A DCT pode ser aplicada na imagem por completa, mas para aumentar a eficiência aconselha-se particionar a imagem em blocos, e aplicar a DCT sobre cada um dos blocos de forma independente. A determinação do tamanho do bloco pode afetar tanto na quantidade de erro introduzida na imagem quanto na complexidade computacional. Os tamanhos dos blocos podem ser de 8×8, 16×16, 32×32, 64×64, etc., sendo os mais usuais 8×8 e 16×16.

Na Figura 3, é apresentado o resultado da aplicação da DCT-II em blocos de 16×16 de uma imagem da face com dimensão 96×96 *pixels*. A Figura 3 (à esquerda) contém a imagem da face original e na Figura 3 (à direita) a imagem resultante após a aplicação da DCT-II.

Figura 3 – Imagem original da base de imagens VidTIMIT (à esquerda) e sua transformada DCT (à direita).



Fonte – Figura elaborada pela autora.

Podemos observar na Figura 3 (à direta) uma grande compactação de energia nos cantos superior esquerdo de cada um dos blocos 16×16, isso corresponde aos componentes de mais baixa frequência, ou seja, é onde estão concentradas as informações mais importante da imagem. A Figura 4 apresenta um bloco 16×16 da escala de cinza da imagem original da Figura 3, e a Figura 5 apresenta este mesmo bloco após a aplicação da DCT-II (coeficientes DCT-II). É possível observar na Figura 5 que a amplitude do coeficiente DC (1,1) é tipicamente muito mais alto do que todos os demais (ordem de 10 vezes mais alto). Este valor é expressivamente maior do que os demais pelo fato de o coeficiente DC representar todo o brilho da imagem,

enquanto os valores dos coeficientes AC, se analisados em módulo, expressam a importância dos componentes de frequência correspondentes.

Figura 4 – Escala de cinza da imagem da Figura 3 (à esquerda) - coordenada (1,1) até (16,16).

125	131	137	141	143	146	146	144	145	145	139	132	129	127	128	131
125	128	133	137	139	143	145	146	145	145	139	132	130	128	129	131
131	128	131	134	138	142	145	145	143	142	138	132	130	130	130	132
137	131	131	134	135	138	141	142	141	140	136	129	128	131	132	132
141	135	133	135	132	132	138	140	137	136	133	129	127	129	132	132
147	141	134	131	131	129	131	134	134	134	132	128	127	129	131	133
150	146	137	130	129	129	128	130	133	133	132	130	129	129	130	133
148	149	141	130	128	133	131	130	132	133	133	132	131	128	130	133
141	147	145	133	128	134	132	130	133	133	135	136	134	128	129	134
135	143	145	138	131	130	134	135	134	135	139	139	134	128	130	135
126	133	138	139	134	127	136	140	138	139	144	144	137	133	137	140
108	117	129	138	140	132	135	140	142	144	147	147	143	139	142	143
92	107	125	138	143	138	135	139	146	146	147	149	147	142	142	141
105	112	125	134	143	145	143	142	145	145	148	152	150	146	144	142
133	123	124	131	141	149	149	147	147	146	149	152	151	147	144	141
153	142	131	131	140	151	153	150	150	148	150	153	151	147	144	139

Fonte – Figura elaborada pela autora.

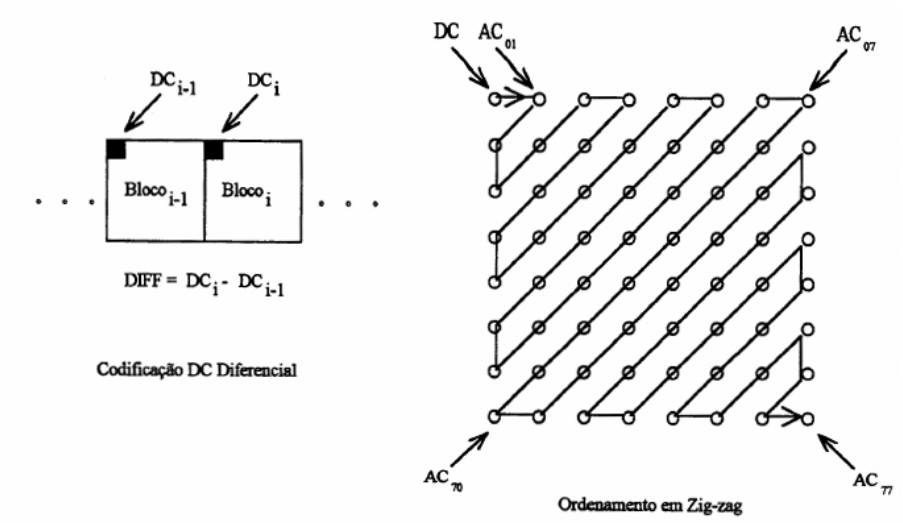
Figura 5 – Coeficientes da DCT-II aplicada sobre a Figura 3 (à esquerda) - coordenada (1,1) até (16,16).

2182,69	-16,61	-41,66	1,10	2,52	-7,79	4,68	-5,83	1,19	-0,75	-0,14	0,95	-0,30	0,79	-0,42	1,32
-32,25	53,49	7,71	0,36	25,10	-4,94	-0,62	3,11	-4,64	1,30	4,63	-2,49	0,79	0,22	-0,58	-1,62
35,35	-29,07	-49,70	-23,34	-8,73	9,03	4,23	11,18	1,76	2,53	2,12	1,98	0,49	0,23	0,30	-0,17
-5,89	-15,07	-20,49	-13,98	-21,42	-25,38	-9,89	-11,73	0,04	-0,94	-0,54	-1,61	-1,83	0,47	0,24	0,81
13,89	31,93	18,37	17,60	12,02	14,74	7,44	-3,79	-6,06	-8,46	0,15	2,01	1,49	-0,58	-1,59	-0,66
-12,88	-17,20	-7,25	-10,53	-13,02	-10,56	-2,75	4,62	0,91	-4,07	-2,01	-0,40	-0,18	0,22	0,20	0,14
6,65	10,82	8,91	8,26	8,73	0,14	0,41	-0,44	2,74	5,73	-0,36	-2,33	-2,63	-0,60	0,56	0,14
-3,68	1,42	-1,59	-4,67	-4,96	-1,94	-1,96	-3,16	-0,98	-0,31	1,16	1,36	0,56	-0,26	-0,09	-0,13
2,06	0,72	-0,85	-1,16	-1,22	-1,49	-1,02	1,12	-1,69	-2,08	-0,39	1,16	-0,51	-0,45	-1,18	-0,18
2,41	0,68	2,42	0,43	3,02	2,78	2,42	0,86	0,41	2,82	0,73	-0,04	0,09	-0,09	0,49	0,18
-0,34	2,81	0,59	0,72	-2,06	-2,37	-1,25	0,35	-0,81	-1,83	-0,98	-0,28	0,25	-0,93	0,13	0,24
-0,35	-1,46	0,66	-0,55	-0,29	1,15	2,39	2,42	0,84	-0,76	0,61	0,15	1,44	0,31	0,28	0,40
2,02	0,86	1,40	-0,12	0,37	1,06	0,57	-1,36	-1,08	-0,44	-0,80	-0,25	-0,27	-0,24	0,73	0,40
-1,00	0,95	-1,49	-0,83	-0,26	0,02	0,20	-0,04	0,86	0,74	0,32	-0,12	-0,24	0,18	-0,05	-0,04
1,53	0,77	0,66	-0,43	-0,69	0,91	0,41	-1,39	-0,09	0,22	0,08	0,02	-0,24	0,13	0,52	0,05
-0,45	-0,79	-0,46	0,19	-0,62	-0,30	-0,86	-0,64	0,99	0,08	0,22	0,80	0,02	-0,59	-0,02	0,07

Fonte – Figura elaborada pela autora.

Para cada blocos 16×16 são gerados 256 coeficientes DCT, como pode ser visto na Figura 5. Esses 256 coeficientes são convertidos em um sequência em zigue-zague, como mostra a Figura 6. Através desse padrão zigue-zague, é possível ordenar os coeficientes em ordem de importância, alocando as frequências mais altas (menos importante) para o final do vetor gerado no zigue-zague. Isto é útil para facilitar o descarte das mesma.

Figura 6 – Ordenação dos coeficientes DCT pelo padrão zigue-zague.



Fonte - Figura retirada de (LUCCA, 1994).

Figura 7 – Reconstrução da face por DCT. Imagem (superior esquerda) usando 256 coeficientes, imagem (superior direita) usando 50% dos coeficientes, imagem (inferior esquerda) usando 25% dos coeficientes e imagem (inferior direita) 10% dos coeficientes.

Fonte – Figura elaborada pela autora.

Na Figura 7 é realizada a reconstrução da face da Figura 3 aplicando a DCT II e a DCT-II inversa. Como mencionado anteriormente a imagem é dividida em blocos de 16×16, resultando para cada bloco 256 coeficientes DCT. A Figura 7 superior esquerda apresenta a

reconstrução da imagem da face utilizando todos os 256 coeficientes DCT, pode-se observar que não há nenhuma alteração na imagem em relação a imagem original (Figura 3 à esquerda). Utilizando 50% (Figura 7 superior direita) e 25% (Figura 7 inferior esquerda) dos coeficientes DCT ainda há poucas alterações na imagem da face. Preservando poucos coeficientes DCT (apenas 10%), é possível observar grandes alterações na imagem mas ainda sendo possível perceber claramente uma face humana (Figura 7 inferior direta).

Através das imagens reconstruídas da Figura 7 é possível observar que a redução de dimensionalidade com DCT gera bons resultados. A reconstrução das imagens leva em consideração apenas os coeficientes DCT das frequências mais baixas, que como mencionada anteriormente, são as informações mais importantes, nas quais apresentam uma redução de detalhes preservando informações importantes que possam caracterizar um face humana. Com esses resultados percebemos que o método DCT é viável para o reconhecimento de faces com uma boa redução de dimensionalidade.

2.2.2.1 Variações da Transformada Discreta do Cosseno Criada por Sanderson

Como mencionado anteriormente na Seção 1.2, Sanderson (SANDERSON, 2008) desenvolve três variações da Transformada Discreta do Cosseno denominadas DCT-mod, DCT-mod2 e DCT-mod-delta. O autor apresenta em seus resultados que essas três variações são eficientes quando a imagem da face sofre grandes variações de iluminação, que considera a principal causa de erros em reconhecimentos faciais. A seguir serão descritas essas três variações criadas por Sanderson.

Inicialmente, Sanderson (SANDERSON, 2008) desenvolve o reconhecimento facial com a DCT-II tradicional, para compará-las com suas variações. Diferente de muitos autores que aplicam a DCT-II em blocos 8×8 adjacentes na imagem, Sanderson aplica a DCT-II tradicional em blocos 8×8 com sobreposição de 50% na horizontal e na vertical, como é ilustrado na Figura 8.

Figura 8 – Imagem (à esquerda) blocos espacialmente vizinhos. Imagem (à direita) blocos sobrepostos 50% na horizontal- Base de imagens VidTIMIT.

Fonte - Figura elaborada pela autora.

Os coeficientes foram ordenados de acordo com o padrão zigue-zague, descrito anteriormente, armazenando as frequências mais baixas às mais altas. O bloco localizado em (*b*,*a*) consiste do vetor de características composto da seguinte forma:

$$d^{(b,a)} = [d_0^{(b,a)} \quad d_1^{(b,a)} \quad ... \quad d_{M-1}^{(b,a)}]^T, \tag{2.2}$$

onde $d_n^{(a,b)}$ é o n-ésimo coeficiente DCT e M a quantidade de coeficientes DCT mantidos por Sanderson. Conhecendo a quantidade de Y linhas e X colunas, o total de blocos contido na imagem é dado pela Equação 2.3, sendo N o tamanho do bloco:

$$N_D = (2\frac{Y}{N} - 1) \times (2\frac{X}{N} - 1). \tag{2.3}$$

Sabendo que todo o brilho da imagem é refletido no coeficiente DC e nos primeiros coeficientes ACs, Sanderson propõe na sua primeira variação DCT que os três primeiros coeficientes DCT sejam descartados, denominando esta variação de DCT-mod. Assim, o vetor gerado pelo padrão zigue-zague apresenta a seguinte forma:

$$d^{(b,a)} = [d_3^{(b,a)} \quad d_4^{(b,a)} \quad ... \quad d_{M-1}^{(b,a)}]^T \tag{2.4}$$

Segundo Soong e Rosenberg (SOONG; ROSENBERG, 1988), a DCT-delta é usada em processamento de sinais para reduzir o ruído do fundo e o desalinhamento de canais. Para

imagens, definimos o n-ésimo coeficiente delta horizontal por:

$$\Delta^h d_n^{(b,a)} = \frac{\sum_{k=-K}^{K} k\, h_k\, d_n^{(b,a+k)}}{\sum_{k=-K}^{K} h_k\, k^2}, \tag{2.5}$$

e o n-ésimo coeficiente delta vertical por:

$$\Delta^v d_n^{(b,a)} = \frac{\sum_{k=-K}^{K} k\, h_k\, d_n^{(b+k,a)}}{\sum_{k=-K}^{K} h_k\, k^2}. \tag{2.6}$$

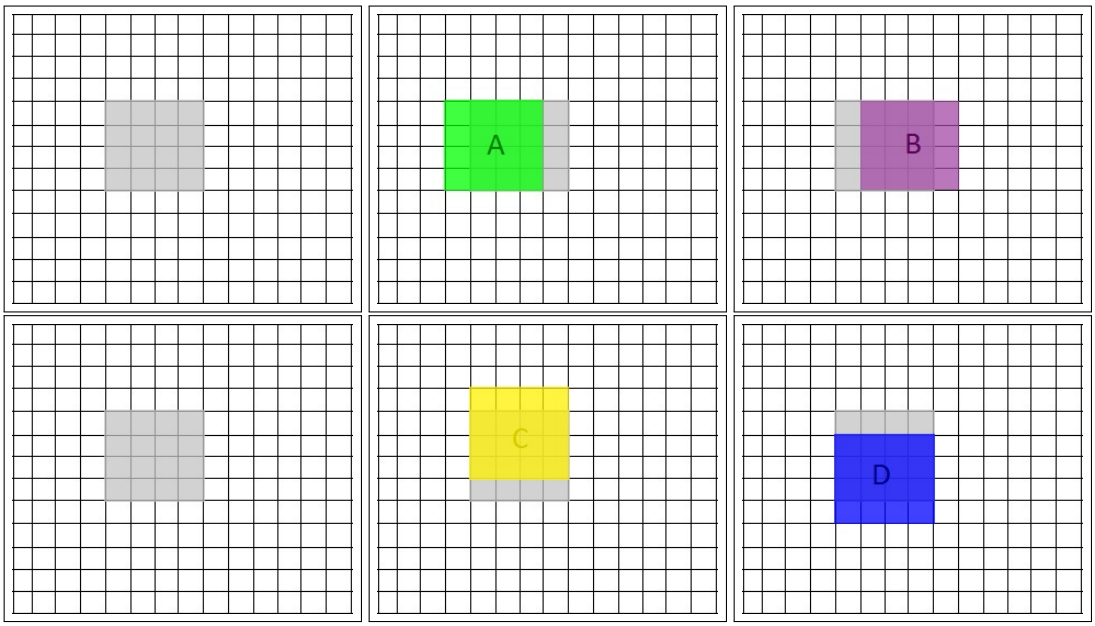
onde h é um vetor simétrico de dimensão $2K+1$. Por exemplo, para $K=1$ e $h = [1 \quad 1 \quad 1]^T$, as Equações 2.7 e 2.8 são reduzidas às equações de diferenças centradas de primeira ordem:

$$\Delta^h d_n^{(b,a)} = \frac{1}{2}(d_n^{(b,a+1)} - d_n^{(b,a-1)}), \tag{2.7}$$

$$\Delta^v d_n^{(b,a)} = \frac{1}{2}(d_n^{(b+1,a)} - d_n^{(b-1,a)}). \tag{2.8}$$

Para uma melhor compreensão da DCT-delta, a Figura 9 apresenta um exemplo passo a passo de seu funcionamento. Neste exemplo, uma parte da imagem é recortada, e cada *pixel* da imagem é representado por uma célula da Figura 9. Inicialmente, a DCT-delta despreza as bordas da imagem, despreza quatro *pixels* na horizontal e quatro *pixels* na vertical e tomada como referência o bloco 4×4 cinza, Figura 9 (à esquerda). A partir deste bloco tomado como referência, a DCT-II tradicional será aplicada no bloco 4×4 deslocado um *pixel* para a esquerda na horizontal, bloco A, e em seguida, aplica-se novamente a DCT-II tradicional no bloco 4×4 deslocado um *pixel* para a direita na horizontal, bloco B. Realizada as DCTs nos blocos A e B, seus coeficientes são armazenados em vetores através do padrão zigue-zague, ou seja, essa primeira operação resultou em dois vetores de características, um representando o bloco A e o outro o bloco B. Esse mesmo processo é realizado na forma vertical, onde a DCT-II tradicional é aplica nos blocos C e D, como é apresentado na Figura 9.

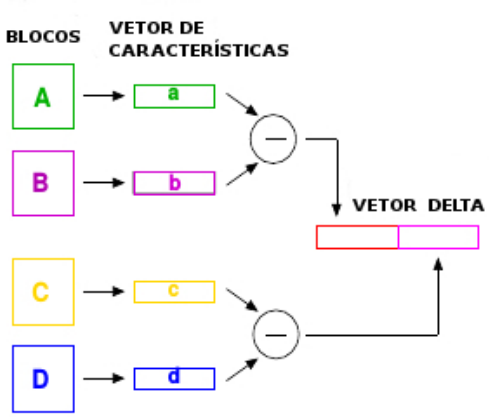
Figura 9 – Funcionamento da DCT-delta



Fonte – Figura elaborada pela autora.

Todo esse processo resultou em quatro vetores de caraterística a, b, c e d, gerados pelo padrão zigue-zague com *M* coeficientes DCT, representando os blocos A, B, C e D, respectivamente, como podemos observar na Figura 10. Depois de obtidos esses vetores de características, toma-se as diferenças a - b e c - d, como ilustrado na Figura 10. Em seguida, após a operação de diferença, os vetores resultantes são concatenados, formando assim o vetor-delta.

Figura 10 – Diagrama em blocos conceitual da extração de características DCT-delta.



Fonte - Figura retirada de (SANDERSON; PALIWAL, 2001).

A segunda variação, DCT-mod-delta, descarta os três primeiros coeficientes do vetor características resultante da DCT-II tradicional (DCT-mod), e concatena o vetor resultante com o vetor características correspondente à DCT-delta. Portanto, a DCT-mod-delta é a concatenação da primeira variação, DCT-mod, com o vetor-delta resultante da operação DCT-delta.

E por fim, a última variação, DCT-mod2, usa os três primeiros coeficientes dos deltas na horizontal e na vertical e concatena no início do vetor característica da variação DCT-mod:

$$x = \left[[\Delta^h d_0 \; \Delta^v d_0 \; \Delta^h d_1 \; \Delta^v d_1 \; \Delta^h d_2 \; \Delta^v d_2] \; [d_3 \; d_4 \; ... \; d_{M-1}] \right]^T . \tag{2.9}$$

2.3 MÉTODOS DE CLASSIFICAÇÃO

2.3.1 *K-Vizinhos Mais Próximo (K-NN)*

O classificador dos K-vizinhos mais próximo, (K-NN, do inglês, *K Nearest Neighbors*) é uma extensão do simples classificador vizinho mais próximo (NN, do inglês, *Nearest Neighbor*). A classificação do vizinho mais próximo é realizada através de uma simples decisão não paramétrica. Cada imagem de consulta I_q é analisada baseando-se na distância de suas características a partir das características das imagens da base de treinamento. O vizinho mais próximo é a imagem que tem a menor distância da imagem de consulta no espaço de característica (EBRAHIMPOUR; KOUZANI, 2007). Existem várias funções para calcular a distância entre duas características, tais como, distância *Manhattan*, distância euclidiana, distância de cosseno ou correlação, respectivamente:

$$d_1(x,y) = \sum_{i=1}^{N} |x_i - y_i| , \tag{2.10}$$

$$d_2(x,y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} , \tag{2.11}$$

$$d_{cos}(x,y) = 1 - \frac{\overrightarrow{x} . \overrightarrow{y}}{|x|.|y|} . \tag{2.12}$$

$$d_{corr}(x,y) = \frac{\sum_{i=1}^{N} (x_i - \mu_i)(y_i - \mu_i)}{\sqrt{\sum_{i=1}^{N} (x_i - \mu_i)^2 \sum_{i=1}^{N} (y_i - \mu_i)^2}} . \tag{2.13}$$

O classificador K-vizinhos mais próximos usa as K amostras mais próximas da imagem de consulta. Cada uma dessas amostras pertence a uma classe C_i conhecida. Dentre as K amostras selecionadas, observa-se a classe predominante entre elas e atribui a imagem de consulta I_q. O desempenho do classificador K-NN está altamente relacionado ao valor de K, ao número de amostras e sua distribuição no espaço de característica (EBRAHIMPOUR; KOUZANI, 2007).

2.3.2 Naïve Bayes

Conhecido como classificador Naïve Bayes ou bayesiano, consiste de uma abordagem estatística para resolver problemas de classificação de padrões. Essa abordagem é baseada na quantificação das comparações entre as várias decisões utilizando a probabilidade e o custo de tais decisões, admitindo que os problemas de decisão são postos em termos probabilísticos e que estes valores são conhecidos (DUDA et al., 2012). É um classificador bastante utilizado devido à sua simplicidade e sua eficiência, ou seja, um algoritmo de fácil implementação que consegue bons resultados de forma rápida.

Para classificar uma observação em uma determinada classe, utiliza-se o conceito da probabilidade condicional, e para serem desenvolvidas as suas funções discriminantes é utilizado o teorema de Bayes dado por:

$$P(\omega_i|x) = \frac{p(x|\omega_i)P(\omega_i)}{p(x)}\,, \text{para } i = 1, \ldots, c\,, \tag{2.14}$$

onde $P(\omega_i|x)$ é a probabilidade a posterior i da classe ω_i dado que foi observado o padrão x, $p(x|\omega_i)$ representa a função densidade de probabilidade condicional, para dados contínuos e função de probabilidade condicional, para dados discretos, $P(\omega_i)$ é a probabilidade a priori para cada classe e o valor de $p(x)$ é dado por:

$$p(x) = \sum_{i=1}^{c} p(x|\omega_i)P(\omega_i)\,, \tag{2.15}$$

$p(x)$ é a probabilidade a priori do vetor de treinamento x, e uma constante, pois não depende da variável ω_i que se está procurando, logo, podemos desprezá-la no momento da classificação (DUDA et al., 2012; CERQUEIRA, 2010; WEBB, 2011). A Equação 2.14 pode ser descrita informalmente como:

$$\text{posteriori} = \frac{\text{verossimilhança} \times \text{priori}}{\text{evidencia}}. \tag{2.16}$$

O classificador bayesiano escolhe a classe que maximize a probabilidade a posteriori, ou seja, que minimize o erro de uma escolha, assim, a regra de decisão pode ser escrita da seguinte forma:

$$p(x|\omega_j)p(\omega_j) > p(x|\omega_k)p(\omega_k) \quad \text{para} \quad k = 1, \ldots, c; \quad k \neq j.$$

Isto é conhecido como regra de Bayes para erro mínimo (WEBB, 2011). Para um exemplo de duas classes, a regra de decisão pode ser escrita como:

$$l_r(x) = \frac{p(x|\omega_1)}{p(x|\omega_2)} > \frac{p(\omega_2)}{p(\omega_1)} \quad \text{implica que x} \in \text{a classe } \omega_1.$$

Visto que a estrutura do classificador de Bayes é determinada pela densidade condicional $p(x|\omega_i)$, várias funções de densidade que foram estudadas, mas nenhuma tem recebido mais atenção do que a densidade normal ou gaussiana (DUDA et al., 2012). As Equações 2.17 e 2.18, representam, respectivamente, a distribuição univariada e multivariada:

$$p(x|\omega_1) = \frac{1}{\sqrt{2\pi\sigma}} exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right], \tag{2.17}$$

$$p(x|\omega_1) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right], \tag{2.18}$$

onde x representa vetor características com d componentes, μ a média de x, σ^2 a variância e Σ a matriz de covariância.

2.3.3 Modelo de Misturas Gaussianas (GMM)

O modelo de misturas gaussianas (GMM, do inglês, *Gaussian Mixture Model*) é a soma de funções gaussianas, cada uma dessas gaussianas é parametrizada por θ_i, que é composto por um vetor de média μ_i, uma matriz de covariância Σ_i e os pesos, onde $i = 1, 2, ..., M$.

$$\theta = \{\alpha_1, \mu_1, \Sigma_1, ..., \alpha_M, \mu_M, \Sigma_M\}.$$

Cada componente de densidade gaussiana possui um peso, resultando numa soma ponderada (ZHANG et al., 2015; SILVA, 2014). A Equação 2.19 apresenta a função ponderada das M componentes:

$$g(x|\theta) = \sum_{i=1}^{M} \alpha_i p(x|\mu_i, \Sigma_i), \tag{2.19}$$

como já apresentado anteriormente, x é um vetor de características de dimensão d, os pesos das misturas são representados por α_i, para $i = 1, 2, ..., M$, e $p(x|\mu_i, \Sigma_i)$, $i = 1, 2, ..., M$, representa as densidades das componentes gaussianas. Cada uma dessas componentes é uma função gaussiana d-variada, representada pela Equação 2.18. Os pesos das misturas devem respeitar ao seguinte critério: $\sum_{i=1}^{M} \alpha_i = 1$ (SILVA, 2014).

Assim sendo, para o treinamento do GMM deve estimar os parâmetros em $\theta = \{\alpha_1, \mu_1, \Sigma_1, ..., \alpha_M, \mu_M, \Sigma_M\}$ apresentados acima. O treinamento é realizado maximizando a verossimilhança dos dados de treinamento. Por exemplo, para $X = \{x_1, x_2, ..., x_T\}$, tem-se que:

$$\theta^* = arg\ max\ g(X|\theta),$$

com

$$g(X|\theta) = \prod_{t=1}^{T} g(x_t|\theta). \tag{2.20}$$

O treinamento pode ser realizado utilizando, por exemplo, o algoritmo iterativo *Expectation-Maximization* (*EM*), usado para determinar os parâmetros de GMM para um conjunto de padrões. Por ser um algoritmo iterativo ele atualiza os valores dos parâmetros do GMM em cada iteração, assim, tornando-o cada vez mais correlacionado ao conjunto de observações. Começa de um modelo inicial θ^0, a cada iteração, um novo modelo θ^{n+1} relaciona-se com o modelo anterior θ^n obedecendo a relação:

$$g(X|\theta^{n+1}) \geq g(X|\theta^n),$$

esse processo é repetido até que um limiar de convergência seja alcançado.

O algoritmo *EM* é realizado em duas fases. A primeira, chamada de *Expectation*, calcula a verossimilhança entre o modelo atual e os dados de treinamento. De acordo com a Equação 2.21, a verossimilhança deve ser calculada para cada um dos vetores de treinamento x_t do conjunto X.

$$Pr(i|x_t,\theta) = \frac{\alpha_i p(x_t|\mu_i,\Sigma_i)}{\sum_{k=1}^{M} \alpha_k p(x_t|\mu_k,\Sigma_k)}\,. \tag{2.21}$$

A segunda fase, chamada de *Maximization*, é responsável por atualizar os parâmetros do GMM. Essa fase altera o modelo atual para que haja uma maior correlação com os dados do modelo anterior, ou seja, os dados de treinamento e o modelo tenham maior semelhança (SILVA, 2014). O novo modelo é gerado partindo do anterior através das seguintes equações:

$$\overline{\alpha_i} = \frac{1}{T}\sum_{t=1}^{T} Pr(i|x_t,\theta)\,, \tag{2.22}$$

$$\overline{\mu_i} = \frac{\sum_{t=1}^{T} Pr(i|x_t,\theta)x_t}{\sum_{t=1}^{T} Pr(i|x_t,\theta)}\,, \tag{2.23}$$

$$\overline{\Sigma_i} = \frac{\sum_{t=1}^{T} Pr(i|x_t,\theta)(x_t-\overline{\mu_i})(x_t-\overline{\mu_i})'}{\sum_{t=1}^{T} Pr(i|x_t,\theta)}\,. \tag{2.24}$$

A inicialização do classificador GMM exige um modelo inicial, isto para que seja possível a estimação de um novo modelo. Há duas formas para se obter esse modelo inicial (SILVA, 2014):

- Inicialização por agrupamento: as médias são inicializadas selecionando o centro de cada grupo, o número de grupos deve ser igual à quantidade de componentes gaussianas do modelo. Os pesos são inicializados uniformemente e a matriz de covariância é a diagonalizada.
- Inicialização aleatória: as médias são obtidas através do conjunto de treinamento, escolhendo-se vetores características aleatórios para a inicialização. A matriz identidade é usada para inicializar a matriz de covariância e os pesos também são inicializados uniformemente.

O critério de parada acontece quando o algoritmo *EM* alcança um máximo local, para isso ou ele deve alcançar o número de iterações ou quando a diferença relativa entre o modelo atual e o anterior for maior que um determinado limiar. Isso significa que o algoritmo encontrou os melhores parâmetros do modelo. Para calcular essa diferença, utiliza-se a razão de verossimilhança no domínio logaritmo, onde para um conjunto de caraterísticas X, entre um modelo $\overline{\theta}$ a testar e um modelo impostor θ é dada por:

$$\Lambda(X) = \log g(X|\overline{\theta}) - \log g(X|\theta)\,. \tag{2.25}$$

O valor de $\Lambda(X)$ é comparado com um limiar de decisão Γ do sistema como forma de atribuir ou não uma pessoa. Caso $\Lambda(X) > \Gamma$, a pessoa é aceita e atribuída, caso $\Lambda(X) < \Gamma$, a pessoa é rejeitada e por isso não é atribuída. A razão de verossimilhança determina o quão melhor a pessoa testada se assemelha ao modelo da pessoa verdadeira quando comparado com modelo impostor (MALHEIRO, 2004).

A verossimilhança entre as características extraídas e o GMM de uma pessoa é calculada por meio de:

$$\log g(X|\overline{\theta}) = \frac{1}{T}\sum_{t=1}^{R} \log g(x_t|\theta)\,. \tag{2.26}$$

sendo X uma sequência de vetores de características e $1/T$ para normalizar a verossimilhança de acordo com o número de vetores característicos extraídos.

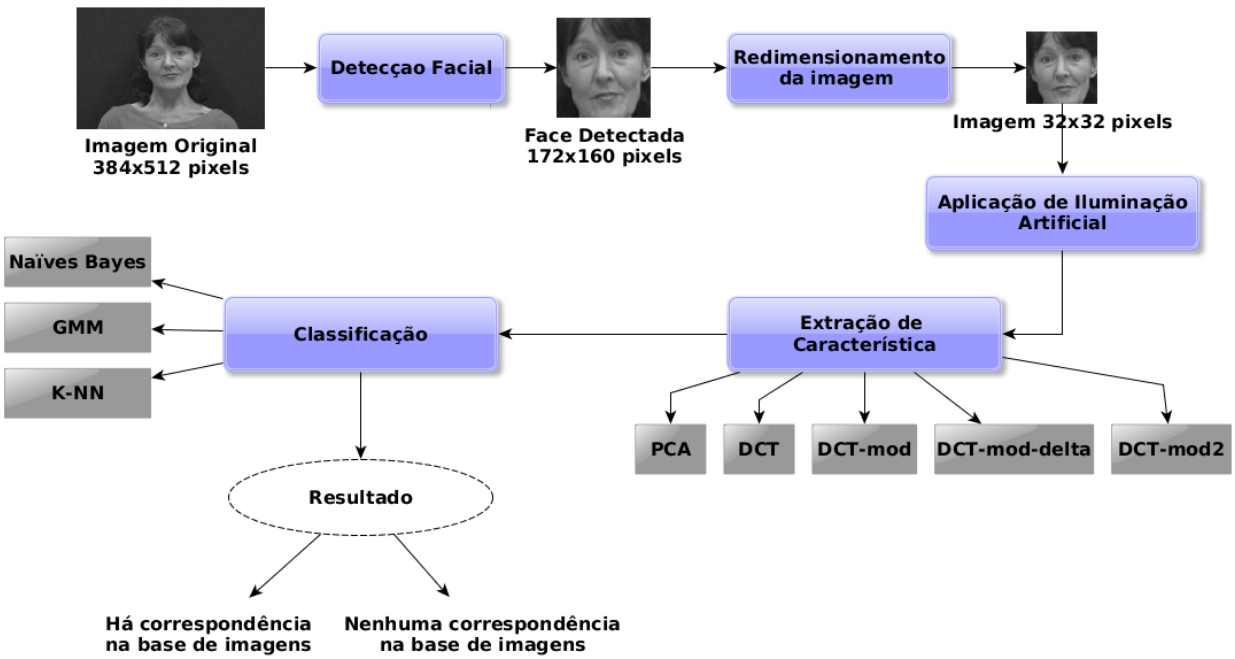
3 METODOLOGIA

Este capítulo visa a apresentar detalhes sobre a base de imagens utilizada, como foi realizada a detecção facial bem como o redimensionamento e a aplicação das iluminações artificiais nas imagens, como foram feitas as extrações de características com as técnicas PCA e DCT, como as faces foram classificadas com os classificadores GMM, Naïve Bayes e K-NN, e por fim, quais os softwares utilizados e suas respectivas versões assim como o hardware usado.

3.1 O RECONHECIMENTO FACIAL

Esta seção tem como objetivo apresentar as etapas do processo de reconhecimento facial realizado neste trabalho. As etapas são divididas da seguinte forma: detecção de faces, redimensionamento das imagens, aplicação de iluminação (somente no conjunto de teste), extração de características e classificação. A Figura 11 ilustra o processo destas etapas que serão detalhadas nas subseções a seguir.

Figura 11 – Diagrama do reconhecimento facial.



Fonte - Elaborado pela autora.

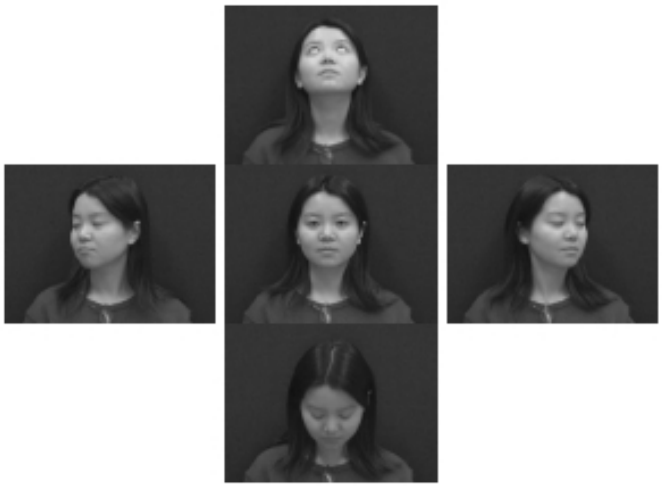
3.1.1 Base de Imagens VidTIMIT

A base de imagens VidTIMIT (SANDERSON; LOVELL, 2009) é composta de vídeos e gravações de áudios correspondendo a 43 pessoas (19 mulheres e 24 homens), gravados em 3 sessões com uma média de tempo de 7 dias entre as sessões 1 e 2, e 6 dias entre as sessões 2 e 3. O vídeo de cada pessoa foi armazenado em uma sequência numerada de imagens JPEG com uma resolução de 384×512 *pixels*, como mostra a Figura 12. Além dos vídeos com a face em posição frontal, também foram gravados sequências de imagens da rotação da cabeça, como mostra a Figura 13.

Figura 12 – Amostras base de imagens VidTIMIT. A primeira, a segunda e a terceira coluna representa as imagens feitas nas sessões 1, 2 e 3, respectivamente.

Fonte - Base de Imagens VidTIMIT (SANDERSON; LOVELL, 2009).

Figura 13 – Amostras base de imagens VidTIMIT. Sequência de rotação da cabeça.

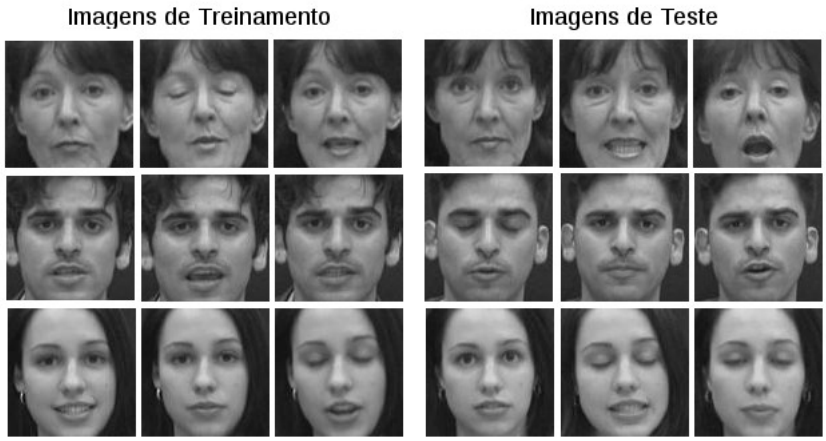


Fonte - Base de Imagens VidTIMIT (SANDERSON; LOVELL, 2009).

3.1.2 Detecção Facial e Redimensionamento das Imagens

Após a aquisição da imagem original 384×512 *pixels*, a etapa de detecção utiliza um algoritmo que busca por uma região de interesse (face do indivíduo). O algoritmo utilizado para detecção da face foi o Viola-Jones (VIOLA; JONES, 2001), sendo um dos mais utilizados na literatura e podendo ser treinado para detectar qualquer objeto. O detector Viola-Jones não detecta a região de interesse sempre com a mesma dimensão, devido a isso, após a detecção todas as imagens foram redimensionadas para 32×32 *pixels*. A Figura 14 apresenta algumas amostras com as faces detectadas e redimensionadas que foram utilizadas neste trabalho. O reconhecimento facial foi realizado somente com imagens frontais sem e com alguma expressão facial. Para cada pessoa 336 imagens foram utilizadas para treinamento e 80 para teste, assim, totalizando 14448 imagens para treinamento e 3440 imagens para teste, todas distintas como mostra a Figura 14.

Figura 14 – Faces detectadas pelo Viola-Jones e redimensionadas para 32×32 *pixels*.



Fonte - Elaborado pela autora.

3.1.3 Aplicação de Iluminação Artificial nas Imagens

Sabendo que a variação de iluminação na imagem é o principal fator para a redução da acurácia (SANDERSON, 2008; VAIDEHI et al., 2010; SHERMINA, 2011), uma mudança de iluminação foi introduzida nas imagens de testes. A iluminação foi aplicada na parte esquerda do rosto, simulando mais iluminação no lado esquerda da face. A simulação da mudança de iluminação foi realizada de acordo com os experimentos de Sanderson (SANDERSON, 2008), que para simular mais iluminação no lado esquerdo da face e menos do lado direito, uma nova janela face $v(y,x)$ é criada pela transformação $w(y,x)$:

$$v(y,x) = w(y,x) + mx + \delta\,, \tag{3.1}$$

sendo

$$m = \frac{-\delta}{(N_x - 1)/2}\,, \tag{3.2}$$

e δ o fator de iluminação delta .

Algoritmo 1: APLICAR ILUMINAÇÃO ARTIFICIAL NAS IMAGENS

Entrada: Imagem w, **inteiro** N_y, **inteiro** N_x

% N_x e N_y dimensão da imagem

Saída: Imagem v

início

  $\delta = 50$

  $m = -\delta/((N_x - 1)/2)$

  para y = *1* ***até*** N_y **faça**

    para x = *1* ***até*** N_x **faça**

      $v(y,x) = w(y,x) + mx + \delta$

    fim

  fim

  retorna Imagem v

fim

As mudanças de iluminação nas imagens foram realizadas com δ = 0, 10, 20, 30, 40, 50, 60 e 70. Algumas amostras dessas mudanças de iluminação nas faces são apresentadas na Figura 15. É possível notar que as amostras contêm mudanças de iluminação artificial, não cobrindo todos os efeitos possíveis da vida real, mas sendo útil para fornecer resultados significativos.

Figura 15 – Mudança de Iluminação. Primeira imagem: $\delta = 0$ (sem mudança de iluminação), segunda imagem: $\delta = 30$, terceira imagem: $\delta = 50$, quarta imagem: $\delta = 70$ e quinta imagem: $\delta = 90$.

Fonte - Elaborado pela autora.

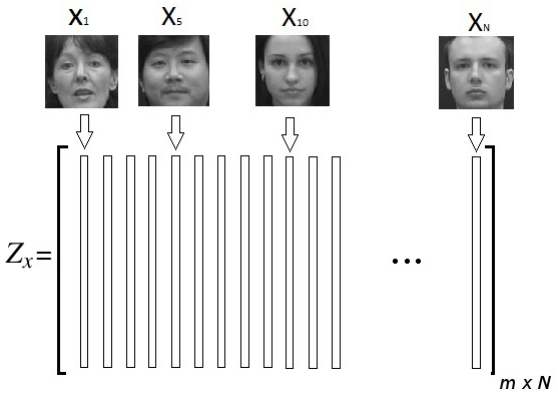
3.1.4 Extração de Características

Como mencionado anteriormente, a extração de características foi realizada através das técnicas: Análise de Componentes Principais e Transformadas Discreta do Cosseno e suas variações DCT-mod, DCT-mod-delta e DCT-mod2. As duas seções a seguir apresentam como foram realizadas as extrações de características.

3.1.4.1 Reconhecimento Facial com PCA

Considerando um conjunto de N imagens de faces de $l \times c$ *pixels* e concatenando cada imagem desse conjunto, é possível agrupar cada vetor dessas faces em uma matriz Z_x que será composta por $l \times c \times N$ elementos. Desta forma, cada coluna da matriz Z_x representa uma face e cada linha os *pixels* das faces. Na Figura 16, é apresentado o espaço de faces deste trabalho, com m linhas por N colunas, sendo $m = l \times c$ e N o total de faces.

Figura 16 – Representação do espaço de faces Z_x.



Elaborado pela autora.

Fase de Treinamento

Depois do espaço de faces Z_x construído, calculamos o vetor média pela Equação 3.3.

$$\Psi = \frac{1}{N} \sum_{i=1}^{N} x_i \,. \tag{3.3}$$

A Equação 3.3 representa um vetor médio, conhecido na literatura de reconhecimento facial como face média, e, tem por objetivo eliminar informações redundantes na face. A Figura

17 apresenta a face média do conjunto de treinamento deste trabalho, representando tudo aquilo que é comum a todas as faces do conjunto de treinamento.

Figura 17 – Face média.

Fonte – Elaborado pela autora.

Uma vez obtido o vetor médio, o mesmo atuará como elemento de diferenciação sobre cada face do conjunto de treinamento conforme a Equação 3.4. O vetor de diferenças Φ resultante gera uma matriz A que contém todas as variações de uma determinada face x em relação à face média Ψ.

$$\Phi_i = x_i - \Psi\,, \tag{3.4}$$

$$A = [\Phi_1, \Phi_2, ..., \Phi_i]\,. \tag{3.5}$$

Neste trabalho, a matriz A assume uma dimensão muito grande devido a quantidade de imagens para cada indivíduo e a dimensionalidade da imagem. Por exemplo, se selecionarmos 336 imagens de 32×32 *pixels* para cada indivíduo (336×43 = 14448), a dimensão da matriz A será 1024×14448.

No próximo passo deve-se calcular a matriz de covariância C como forma de definir o subespaço da imagem.

$$C = AA^T\,. \tag{3.6}$$

Tomando como base o exemplo referente a dimensão da matriz A, 1024×14448, conclui-se que a matriz de covariância possuirá uma dimensão de 1024×1024, o que faz com que

os cálculos de seus autovetores sejam computacionalmente viáveis. Os autovetores e autovalores da matriz de covariância são calculados da seguinte forma:

$$A^T A v_i = \lambda v_i \,. \tag{3.7}$$

Multiplicando ambos os lados por A, tem-se:

$$A A^T A v_i = \lambda A v_i \,. \tag{3.8}$$

Desta forma, pode-se observar que Av_i são os autovetores de $C = AA^T$ associados aos 1024 autovalores da matriz para este exemplo. Se apenas os autovetores associados aos maiores autovalores são considerados, a variância total do padrão não muda muito e a dimensionalidade é m sendo $m \ll 1024$.

A partir deste momento cada imagem de treinamento pode ser projetada no espaço de faces. Assim, o descritor PCA pode ser obtido através da combinação linear de autovetores com os vetores originais das imagens, como mostra a Equação 3.9:

$$W_n = v_n^T (x - \Psi) \,, \tag{3.9}$$

onde, n = 1, 2, ..., m, v_n são os autovetores, x o vetor de faces de treinamento e Ψ a face média.

Fase de Reconhecimento

Nesta etapa, é necessário colocar a imagem de consulta em um vetor Q que será projetado frente ao espaço de face (combinação linear de autovetores).

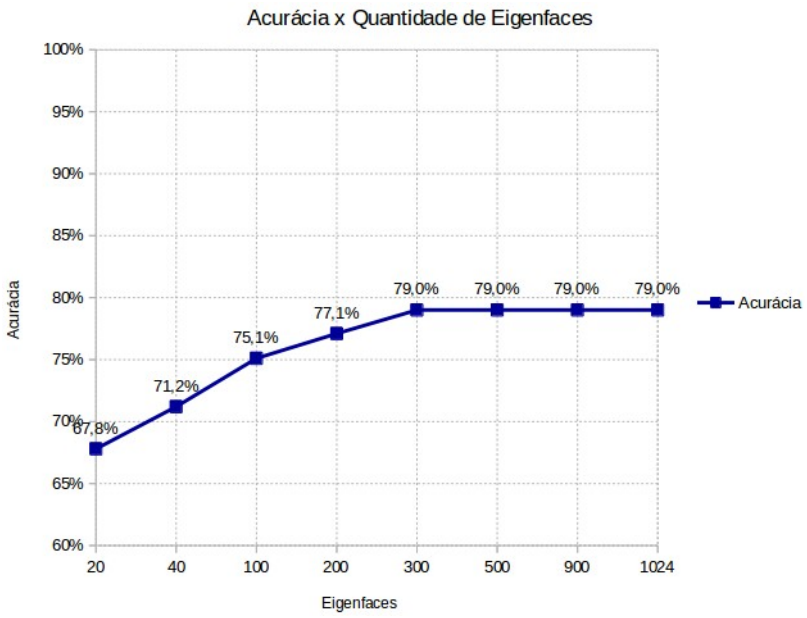
$$W_n = v_n^T (Q - \Psi) \,. \tag{3.10}$$

Por fim, um classificador compara a distância entre o descritor do vetor de consulta com um dos descritores armazenados na base de imagens para a identificação das faces.

Eigenfaces

As *eigenfaces* são os autovetores, que como visto anteriormente, são alcançados quando se aplica a técnica de extração de características PCA. Como mencionado anteriormente, as informações mais importantes estão sempre em m autovetores, sendo $m \ll 1024$, onde 1024 é o total dos autovetores do conjunto de treinamento deste trabalho. Devido a isso, realizou-se alguns experimentos para encontrar a quantidade de *eigenfaces* que resulta na melhor acurácia. Os resultados são mostrados no gráfico da Figura 18.

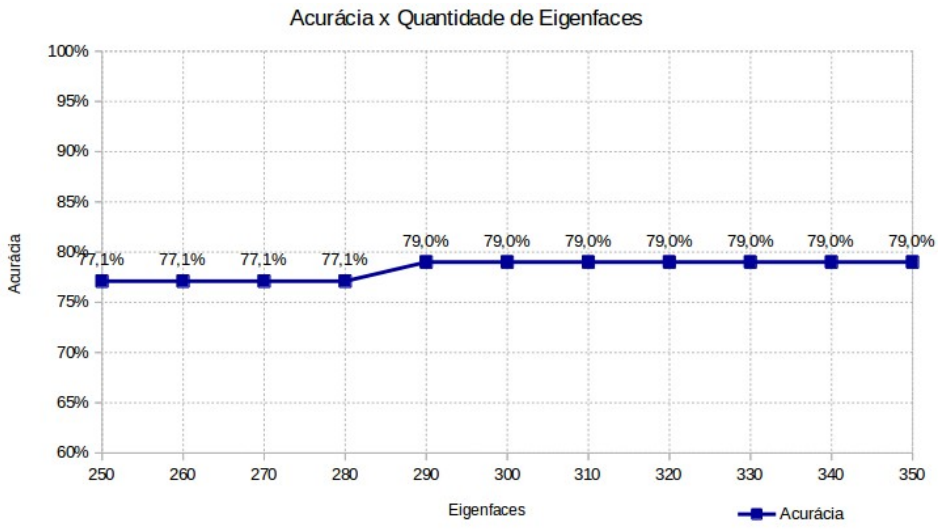
Figura 18 – Acurácia em relação a quantidade de *eigenfaces*.



Fonte - Elaborado pela autora.

Como pode ser visto na Figura 18 a maior acurácia (79%) foi obtida a partir de 290 *eigenfaces*, utilizando a medida de correlação (Equação 2.13) e fazendo os 43×43 indivíduos. Com o objetivo de encontrar uma melhor acurácia foi realizado uma varredura minuciosa de 10 em 10 *eigenfaces* entre 250 e 350 *eigenfaces*, como pode ser observado no gráfico da Figura 19. Contudo, não foi encontrada uma acurácia melhor que 79%. A fim de comparar os resultados deste trabalho com os resultados de Sanderson (SANDERSON, 2008), no qual realizou com somente 40 *eigenfaces*, realizamos o reconhecimento facial com 40 e 300 *eigenfaces*.

Figura 19 – Acurácia em relação a quantidade de *eigenfaces*.



Fonte - Elaborado pela autora.

3.1.4.2 Reconhecimento Facial com DCT

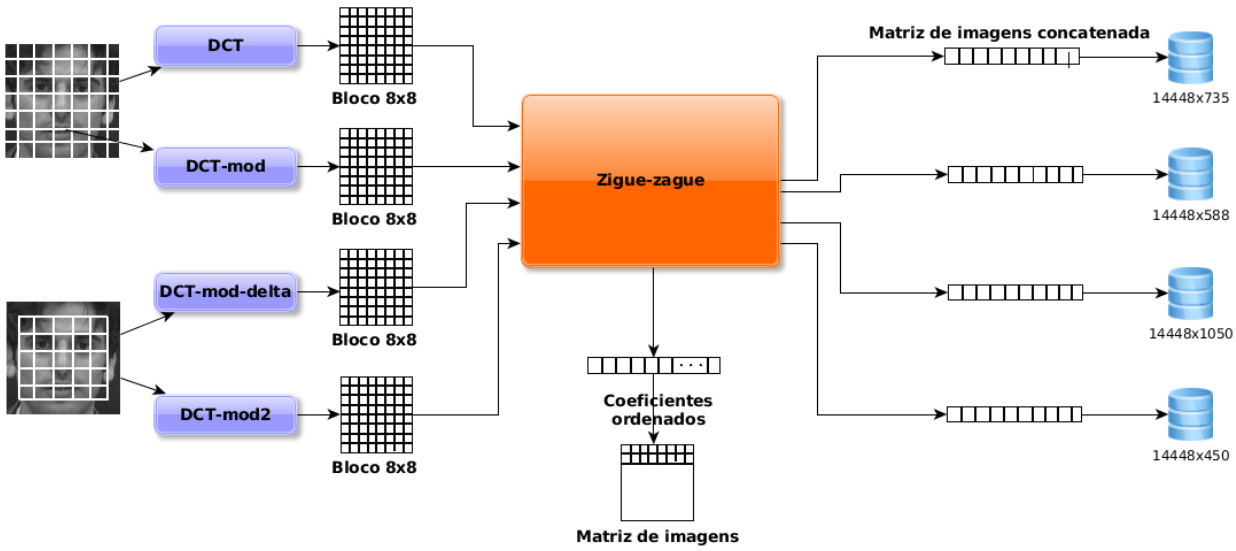
A extração de características através do método DCT foi realizada em blocos 8×8 nas imagens. Esses blocos se sobrepõe em 50% cada um conforme Figura 8. A DCT e a DCT-mod aplicadas a cada uma das imagens com 32×32 *pixels* e blocos se sobrepondo 50% geram um total de 49 blocos (Seção 2.2.2.1) com 64 coeficientes DCT, dos quais somente os 15 primeiros coeficientes foram escolhidos para DCT e 12 para DCT-mod. Assim, cada imagem é representada por uma matriz 49×15 para DCT e 49×12 para a DCT-mod. Para formar o conjunto de treinamento cada matriz foi concatenada para obtemos o vetor de características que representa cada imagem após a DCT e DCT-mod. Cada vetor de características terá uma dimensão de 1×735 para DCT e 1×588 para DCT-mod. Sabendo que para compor o conjunto de treinamento foram selecionadas 336 imagens para cada um dos 43 indivíduos, o conjunto de dados para treinamento será composto por uma matriz 14448×735 para DCT e 14448×588 para DCT-mod. O conjunto de dados para teste será composto por 3440×735 para DCT e 3440×588 para DCT-mod.

A quantidade de blocos para cada imagem após a aplicação da DCT-mod-delta e DCT-mod2 é de 25 blocos, conforme explicado na seção 2.2.2.1. Para cada bloco gerado pela DCT-mod-delta foram escolhidos 30 coeficientes resultantes das operações de diferença entre os coeficientes dos blocos (A e B) e (C e D) mais 12 coeficientes do DCT-mod, assim, cada imagem após a aplicação da DCT-mod-delta é representada por uma matriz de 25×42. Já

para a DCT-mod2 foram escolhidos 18 coeficientes, resultando em uma matriz 25×18 para cada imagem. Os conjuntos de dados de treinamento para a DCT-mod-delta e DCT-mod2 terão dimensão de 14448×1050 e 14448×450, respectivamente, e os conjuntos de teste 3440×1050 e 3440×450, respectivamente.

A Figura 20 apresenta um diagrama da construção da base de treinamento para o reconhecimento facial. Inicialmente, cada imagem é divida em blocos 8×8, em cada bloco é aplicada uma das DCT's, após a aplicação de uma das DCT's os coeficientes são ordenados através do padrão zigue-zague, nos quais irão compor uma matriz que representa todos os coeficientes de cada bloco 8×8 da imagem. Em seguida, essa matriz é concatenada, gerando o vetor de características para cada imagem. Por fim, cada vetor de característica é armazenado para compor a base de treinamento. Esse mesmo procedimento é realizado para a construção da base de teste. Após a construção das bases de treinamento e de teste, um classificador é utilizado para verificar se há alguma correspondência entre as imagens.

Figura 20 – Diagrama da construção da base de treinamento.



Fonte - Elaborado pela autora.

3.1.4.3 Classificação das Faces

Para classificar as faces foram utilizados os classificadores Naïve Bayes, Modelo de Misturas Gaussianas (GMM) e K-Vizinhos Mais Próximo (KNN). Sanderson implementou o classificador GMM utilizando o algoritmo K-means seguidos de 10 iterações do algoritmo

EM com 8 gaussianas e utilizando a matriz de covariância diagonal. No classificador GMM deste trabalho a inicialização da média, matriz de covariância e pesos foram aleatórios, seguido de 2500 iterações do algoritmo EM, com 8 gaussianas e matriz de covariância completa. Para o classificador K-NN foram utilizadas as funções euclidiana, cosseno e a correlação com 2 vizinhos (K = 2). E para o classificador Naïve Bayes utilizou-se a função de densidade normal.

3.2 HARDWARE E SOFTWARE UTILIZADOS

Foi utilizada a ferramenta computacional MATLAB como linguagem de programação base para as implementações de todos os algoritmos dos experimentos citados. Esta seleção foi baseada na disponibilidade de recursos para cálculos matemáticos baseados em matrizes e por permitir um desenvolvimento ágil de protótipos por meio de módulos específicos (*toolboxes*).

Para execução dessas impletações foi utilizado um computador pessoal com processador Intel Core i5 2,30 GHz, 8GB de memória RAM, 500GB de disco rígido e sistema operacional Ubuntu 14.04 64 bits.

4 RESULTADOS

Este capítulo apresenta uma análise comparativa dos métodos de extração de características bem como dos classificadores descritos anteriormente em imagens com grandes variações de iluminação.

Como mencionado na seção anterior, foi realizado um reconhecimento facial de 43 pessoas com face completa em imagens com dimensão 32×32 *pixels*. Para todos os experimentos o método de validação cruzada utilizado foi o *holdout*, com 336 imagens para treinamento e 80 imagens para teste. A técnica empregada para avaliar o desempenho dos experimentos foi através da construção de matrizes de confusão, analisando a acurácia e o índice *kappa*. Na Seção 4.1, cada subseção apresenta os resultados dos métodos de extração de característica associado a um classificador.

4.1 RESULTADOS POR CLASSIFICADOR

Cada conjunto de características foi analisado com os classificadores GMM, Naïve Bayes e KNN, cujos parâmetros utilizados em cada classificador são mostrados na Tabela 1. As acurácias e os erros foram obtidos por meio das Equações 4.1 e 4.2, respectivamente.

Tabela 1 – Parâmetros selecionados para os classificadores.

CLASSIFICADORES	PARÂMETROS
GMM	2500 iterações do EM, 8 gaussianas e matriz de covariância completa
Naïve Bayes	distribuição normal
KNN	K = 2, medidas: correlação, distância euclidiana, e cosseno

$$ACC = \frac{VP + VN}{VP + VN + FP + FN} \times 100\%, \tag{4.1}$$

VP: Verdadeiros Positivos VN: Verdadeiros Negativos

FP: Falsos Positivos FN: Falsos Negativos

$$ERRO = 100\% - ACC\,. \tag{4.2}$$

4.1.1 Classificador Modelo de Misturas Gaussianas (GMM)

A Tabela 2 a seguir apresenta as acurácias dos métodos de extração de características em cada uma das variações de iluminação com o classificador modelo de misturas gaussianas utilizando 8 gaussianas.

Tabela 2 – Acurácia com Classificador GMM - 8 Gaussianas

MÉTODOS	**ACURÁCIAS VARIANDO A ILUMINAÇÃO δ (%)**							
	δ = 0	**δ = 10**	**δ = 20**	**δ = 30**	**δ = 40**	**δ = 50**	**δ = 60**	**δ = 70**
PCA (40 eigenfaces)	71,2	67,6	61,7	45,5	32,5	20,2	10,5	8,5
PCA (300 eigenfaces)	79,0	76,7	69,2	44,9	24,9	9,5	2,7	2,3
DCT	84,2	85,8	83,5	77,6	70,4	55,5	29,5	16,7
DCT-mod	83,8	83,5	83,2	81,7	79,5	79,1	72,3	56,9
DCT-mod-delta	93,4	93,5	93,3	93,4	93,4	92,6	90,7	84,7
DCT-mod2	93,2	93,1	93,1	93,6	93,9	92,8	89,4	71,1

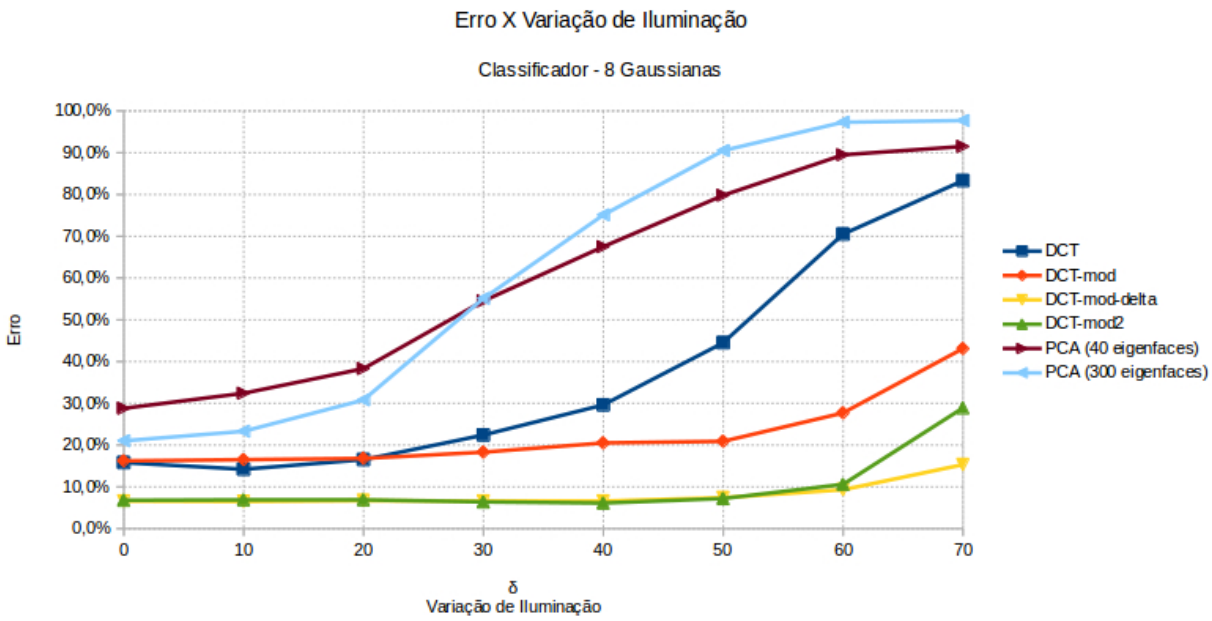
O gráfico da Figura 21 apresenta os erros de cada método de extração de características em relação a cada variação de iluminação δ, proporcionando assim, uma melhor análise do desempenho de cada método à medida que a iluminação sobre a imagem é aplicada.

É possível observar que os erros no método PCA em ambas as quantidades de *eigenfaces* são crescentes a medida que a iluminação é acentuada na imagem, apresentando sempre os maiores erros quando comparado aos métodos DCT. O PCA utilizando 40 eigenfaces apresentou erros de 28,8% e 91,5% nas variações de iluminação δ = 0 e δ = 70, respectivamente. Com 300 *eigenfaces* o erro reduziu para 21% em δ = 0 mas aumentou em δ = 70 para 97,7%.

Os erros no método DCT também são crescentes à medida que a iluminação na imagem é acrescida, mas menores em relação aos erros do PCA. Os erros com imagens sem nenhuma variação de iluminação (δ = 0) e com variação de iluminação extrema (δ = 70) foram 15,8% e 83,3%, respectivamente.

Os métodos DCT-mod, DCT-mod-delta e DCT-mod2 mostraram-se melhores comparados aos métodos PCA e DCT. O método DCT-mod alcançou erros de 16,2% e 43,1% em δ = 0 e δ = 70, respectivamente, se mantendo estável até δ = 50. O método DCT-mod-delta obteve

Figura 21 – Comparativo entre os métodos de extração de características no classificador GMM



Fonte - Elaborado pela autora.

erros 6,6% em $\delta = 0$ e apenas 15,3% em imagens com variação de iluminação extrema ($\delta = 70$) e, o DCT-mod2 obteve erros 6,8% e 28,9% em $\delta = 0$ e $\delta = 70$, respectivamente. Os métodos DCT-mod-delta e DCT-mod2 se mantiveram muito estáveis até $\delta = 50$, onde alcançaram erros de apenas 7,4% e 7,2%, respectivamente.

Nos resultados de Sanderson, a DCT-mod também apresentou a menor acurácia em relação a DCT-mod-delta e DCT-mod2, mas se manteve estável até $\delta = 70$, diferente do resultado da Figura 21. Sanderson também mostrou que suas melhores acurácias foram nos métodos DCT-mod-delta e DCT-mod2, ambos se mantendo estáveis até $\delta = 70$. Na Figura 21 é possível observar que os métodos DCT-mod-delta e DCT-mod2 também obtiveram as melhores acurácias e se mantiveram estáveis até a variação de iluminação $\delta = 60$. Apesar dos resultados da Figura 21 não serem idênticos aos de Sanderson, é possível reafirmar a sua tese, na qual afirma que suas variações da DCT (DCT-mod, DCT-mod-delta e DCT-mod2) são eficientes em grandes variação da iluminação diferente de outros métodos como PCA e DCT, que sofrem grandes quedas nas acurácias a medida que a iluminação é acentuada na imagem.

4.1.2 Classificador Naïve Bayes

A Tabela 3 a seguir apresenta as acurácias dos métodos de extração de características em cada uma das variações de iluminação com o classificador Naïve Bayes.

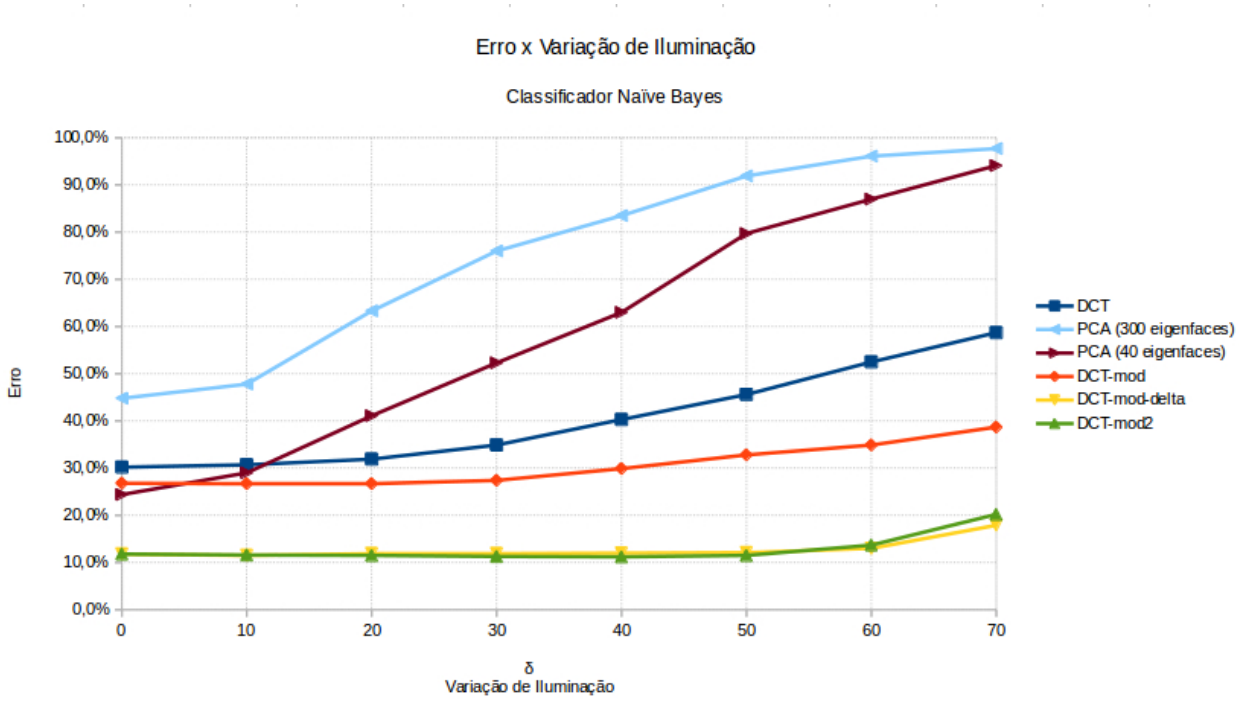
Tabela 3 – Acurácia com Classificador Naïve Bayes.

MÉTODOS	**ACURÁCIAS VARIANDO A ILUMINAÇÃO δ (%)**							
	δ = 0	**δ = 10**	**δ = 20**	**δ = 30**	**δ = 40**	**δ = 50**	**δ = 60**	**δ = 70**
PCA (40 eigenfaces)	75,6	71,0	58,9	47,7	37,0	20,3	13,0	5,9
PCA (300 eigenfaces)	55,2	52,2	36,7	24,0	16,5	8,1	3,9	2,3
DCT	69,8	69,3	68,1	65,1	59,7	54,4	47,7	41,3
DCT-mod	73,2	73,3	73,3	72,6	70,1	67,2	65,1	61,3
DCT-mod-delta	88,2	88,5	88,1	88,1	88,0	87,9	87,0	82,1
DCT-mod2	88,2	88,4	88,5	88,7	88,8	88,5	86,3	79,8

Na Tabela 3 é possível observar que o PCA com 40 *eigenfaces* obteve acurácias melhores em algumas variações de iluminação em relação a Tabela 2, nas variações de iluminação $\delta = 0, 10, 30, 40, 50$ e 60 atingiu 75,6%, 71,0%, 47,7%, 37,0%, 20,3% e 13,0 %, respectivamente. Logo, o PCA 300 *eigenfaces* foi melhor no classificador GMM e, pior de todos os métodos no classificador Naïve Bayes, atingindo 55,2% e 2,3% nas variações de iluminação $\delta = 0$ e 70, respectivamente. Também é possível notar que o método DCT-mod atingiu uma acurácia inferior, 73,2%, na variação de iluminação $\delta = 0$ comparado ao PCA com 40 *eigenfaces* que atingiu 75,6%, mas foi superior aos métodos PCA e DCT nas demais variações de iluminação, como pode ser observado melhor no gráfico da Figura 22.

Também é possível observar por meio do gráfico da Figura 22 que as variações DCT (DCT-mod-delta e DCT -mod2) continuam se mostrando eficientes em grandes variações de iluminação, mas com valores de acurácias inferiores em relação a estes métodos quando associados ao classificador GMM. O método DCT-mod-delta foi o que obteve a melhor acurácia em relação a todos os outros métodos quando associados as classificador Naïve Bayes, alcançando 88,2% sem variação de iluminação ($\delta = 0$) e 82,1% na pior variação de iluminação ($\delta = 70$).

Figura 22 – Comparativo entre os métodos de extração de características no classificador Naïve Bayes.



Fonte - Elaborado pela autora.

4.1.3 Classificador *K*-Vizinhos Mais Próximos (K-NN)

A fim de encontrar uma maior acurácia, os testes com o K-NN foram realizados variando o valor de K e utilizando medidas de dissimilaridade e similaridade. Medida de dissimilaridade mede o quanto dois indivíduos são diferentes, quanto maior for o valor da medida de dissimilaridade menor será a semelhança entre os indivíduos, a medida de dissimilaridade utilizada neste trabalho é a distância euclidiana. Já a medida de similaridade calcula o quanto dois indivíduos são parecidos, assim, quanto maior for a medida de similaridade maior a semelhança entre os indivíduos, as medidas de similaridade utilizadas neste trabalho é a de correlação e a distância do cosseno.

O número de vizinhos K variou de 1 a 5 mas não foi possível encontrar um que forneça uma boa acurácia, pois para cada método de extração de características, principalmente as DCT's, as melhores acurácias não tinham valores de K iguais. Sendo assim, foi escolhido um valor de K intermediário, $K = 2$, baseado nos dois métodos de extração de características que forneceram as melhores acurácias, o DCT-mod-delta e DCT-mod2.

A Tabela 4 a seguir apresenta as acurácias dos métodos de extração de características

em cada uma das variações de iluminação com o classificador K-NN usando a função de distância euclidiana.

Tabela 4 – Acurácia com classificador K-NN (distância euclidiana e $K = 2$).

MÉTODOS	**ACURÁCIAS VARIANDO A ILUMINAÇÃO δ (%)**							
	$\delta = 0$	**$\delta = 10$**	**$\delta = 20$**	**$\delta = 30$**	**$\delta = 40$**	**$\delta = 50$**	**$\delta = 60$**	**$\delta = 70$**
PCA (40 eigenfaces)	92,4	92	91,2	87,3	81,2	67,9	47,5	30,3
PCA (300 eigenfaces)	94,4	92,8	91	87,7	81,9	76,9	58,7	37,3
DCT	87,4	87,7	88,1	86,9	85,5	83,3	76	64,1
DCT-mod	89,2	89,5	89,2	88,7	87,5	86,4	84,7	79,4
DCT-mod-delta	95,6	95,7	95,5	94,9	94	92,9	92,3	90
DCT-mod2	93	93	92,7	92,8	92,4	91,9	91,6	89

Os métodos de extração de características que obtiveram as melhores acurácias com o classificador K-NN utilizando a função de distância euclidiana foram a DCT-mod-delta e DCT-mod2, atingindo 95,6% e 93%, respectivamente, sem nenhuma alteração da iluminação ($\delta = 0$) e 90% e 89%, respectivamente, com variação extrema na iluminação ($\delta = 70$). Na Figura 23 é possível observar uma grande queda da acurácia para o método PCA (40 eingefaces e 300 eigenfaces), chegando a atingir erros de 69,7% e 62,7%, respectivamente, com $\delta = 70$.

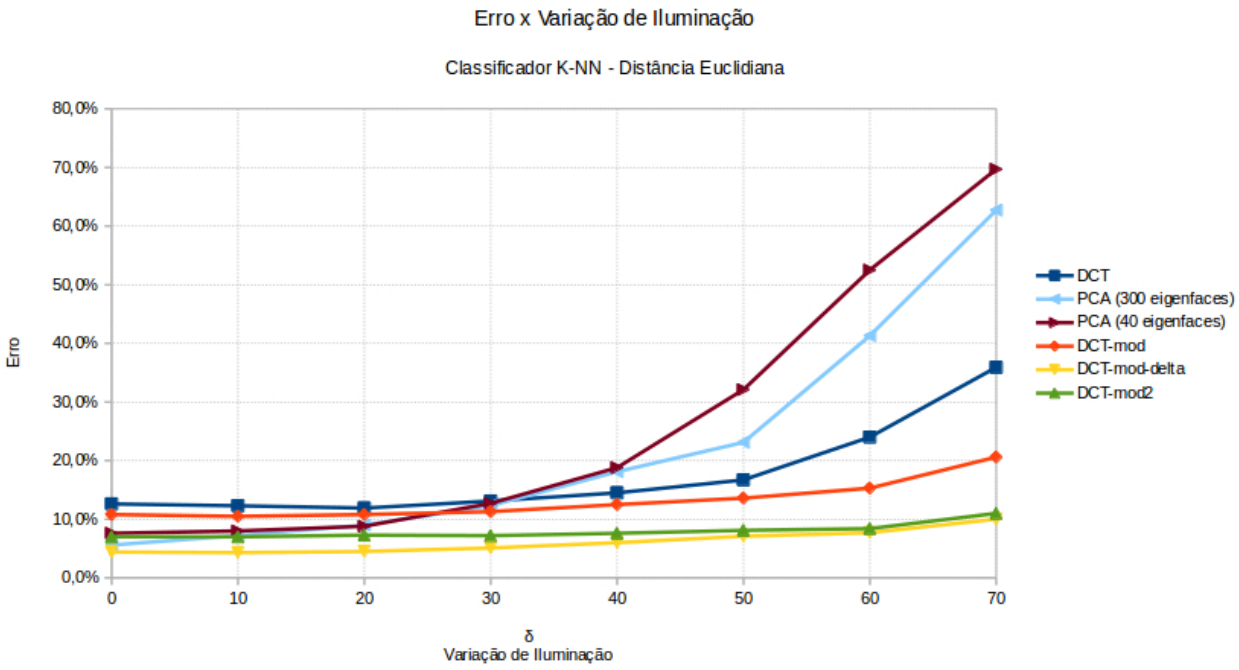
A Tabela 5 a seguir apresenta as acurácias dos métodos de extração de características em cada uma das variações de iluminação com o classificador K-NN usando a medida de correlação.

Tabela 5 – Acurácia com classificador K-NN (Medida de correlação e $K = 2$).

MÉTODOS	**ACURÁCIAS VARIANDO A ILUMINAÇÃO δ (%)**							
	$\delta = 0$	**$\delta = 10$**	**$\delta = 20$**	**$\delta = 30$**	**$\delta = 40$**	**$\delta = 50$**	**$\delta = 60$**	**$\delta = 70$**
PCA (40 eigenfaces)	90,1	89,1	85,6	81,6	70,7	51,5	35,9	17,7
PCA (300 eigenfaces)	90,5	90,2	88,8	84,5	77,9	66,5	47,5	30,3
DCT	88,3	88,5	89,5	89,5	88,4	84,5	77,9	69
DCT-mod	90,8	90,8	90,7	90,9	89,9	88,1	87,9	88,3
DCT-mod-delta	97	97,2	96,9	96,4	95,5	94,7	94,2	92,6
DCT-mod2	92,8	92,4	92,7	92,2	92	91,7	91,3	86,8

Novamente, as melhores acurácias foram obtidas nos métodos DCT-mod-delta e DCT-mod2, 97% e 92,8%, respectivamente, sem nenhuma variação de iluminação $\delta = 0$, e 92,6% e 86,8%, respectivamente, com variação de iluminação $\delta = 70$. Na Figura 24 é possível

Figura 23 – Comparativo entre os métodos de extração de características no classificador K-NN com distância euclidiana e K = 2.



Fonte - Elaborado pela autora.

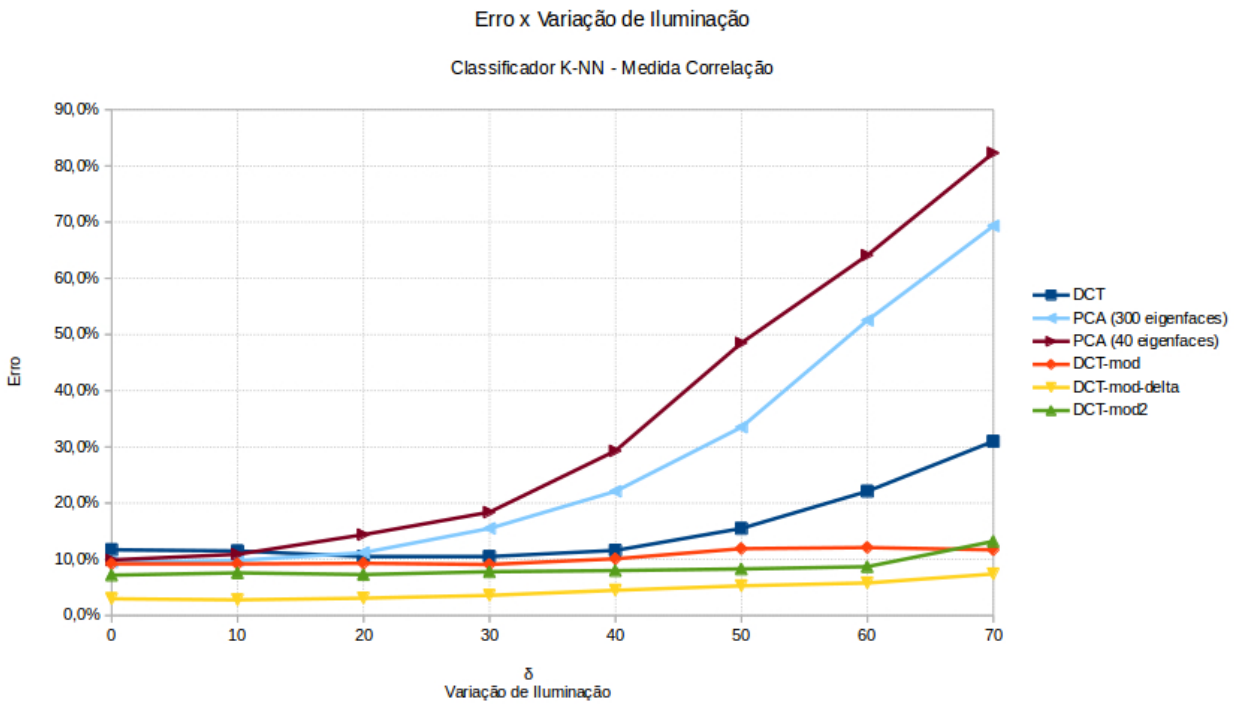
observar que o método PCA na medida de correlação ainda não se mostrou eficiente a medida que a iluminação é aplicada na imagem. É possível também observar que a DCT-mod com essa função de medida obteve a terceira melhor acurácia e se manteve praticamente constante até a variação de iluminação $\delta = 70$, como se manteve constante nos resultados de Sanderson.

A Tabela 6 a seguir apresenta as acurácias dos métodos de extração de características em cada uma das variações de iluminação com o classificador K-NN usando a função de distância do cosseno.

Tabela 6 – Acurácia com classificador K-NN (distância cosseno e $K = 2$).

MÉTODOS	**ACURÁCIAS VARIANDO A ILUMINAÇÃO δ (%)**							
	$\delta = 0$	**$\delta = 10$**	**$\delta = 20$**	**$\delta = 30$**	**$\delta = 40$**	**$\delta = 50$**	**$\delta = 60$**	**$\delta = 70$**
PCA (40 eigenfaces)	90,8	89,4	86,3	82,5	73,1	54,7	38,4	17,4
PCA (300 eigenfaces)	90,6	90,3	88,6	84,5	78	66,5	48,2	30,2
DCT	88,9	89	89,5	89,6	88,8	84,8	77,9	68,7
DCT-mod	90,9	91,2	91,1	91,4	89,9	88,3	88,3	88,5
DCT-mod-delta	96,7	96,8	96,7	96,1	95,1	94,6	94	92,6
DCT-mod2	93,3	93,2	93,3	93,1	92,8	92,3	91,9	87,8

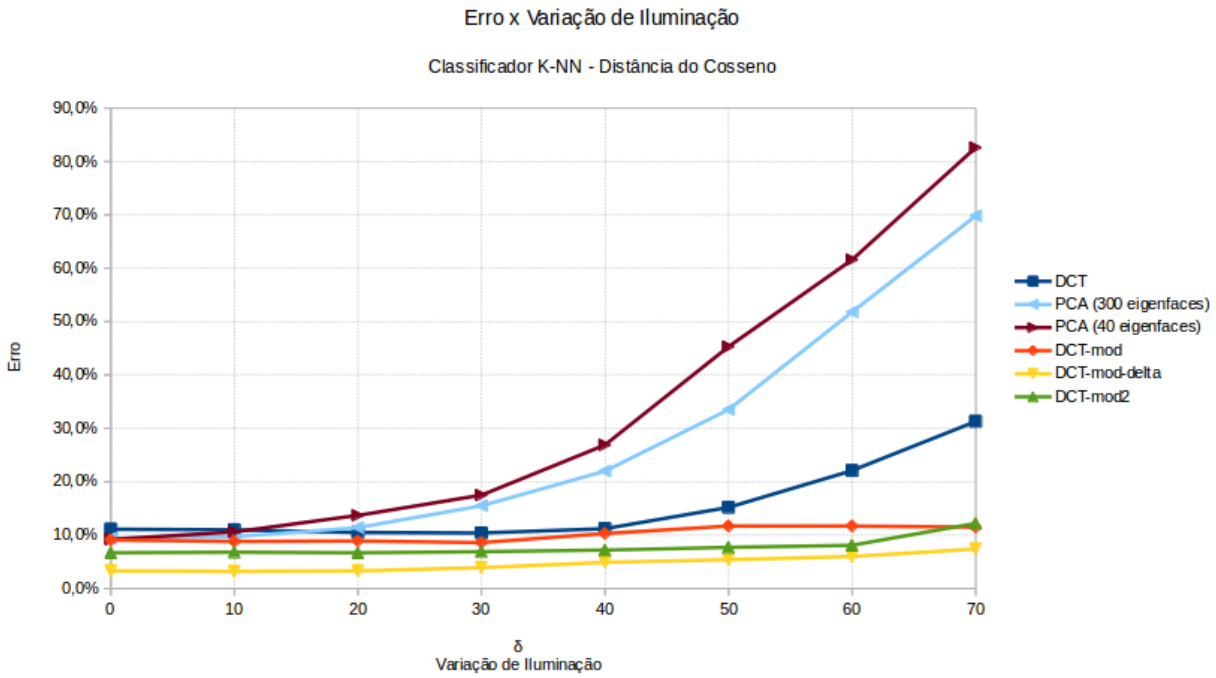
Figura 24 – Comparativo entre os métodos de extração de características no classificador K-NN com a medida de correlação e K = 2.



Fonte - Elaborado pela autora.

Utilizando a função distância cosseno, também foi possível obter bons valores de acurácias para os métodos DCT-mod-delta e DCT-mod2, 96,7% e 93,3%, respectivamente, para imagens sem nenhuma variação de iluminação ($\delta = 0$) e 92,6 e 87,8%, respectivamente, em imagens com variação de iluminação extrema ($\delta = 70$). Observando a Figura 25 a DCT-mod ainda continua se mantendo constante com uma das melhores acurácias, ultrapassando a DCT-mod2 em $\delta = 70$, atingindo um erro de 11,5%.

Figura 25 – Comparativo entre os métodos de extração de características no classificador K-NN com distância cosseno e K = 2.



Fonte - Elaborado pela autora.

4.2 O ÍNDICE *KAPPA* NA AVALIAÇÃO DO DESEMPENHO DOS CLASSIFICADORES

A fim de realizar uma análise comparativa entre os classificadores, foram escolhidos os dois métodos de extração de características que obtiveram as melhores acurácias com imagens sem nenhuma variação de iluminação ($\delta = 0$) e com variação de iluminação extrema ($\delta = 70$). A avaliação do desempenho dos classificadores foi feita através do valor do índice *kappa*.

O índice *kappa* é uma das variáveis que podem ser quantificadas após construir a matriz de confusão, sendo um índice que retrata o grau de concordância dos dados, gerando assim, um aspecto de confiabilidade e precisão dos dados classificados (PERROCA; GAIDZINSKI, 2003). O resultado obtido pelo índice *kappa* varia entre 0 a 1, sendo que quanto mais próximo de 1, melhor a qualidade dos dados classificados. Vários são os índices para agrupar esses dados quantitativos para qualitativos, entre eles, pode ser destacado o de Fonseca (2000), conforme a Tabela 7.

Tabela 7 – Agrupamento qualitativo do índice *kappa*.

Índice *kappa*	**Desempenho**
<0	Péssimo
$0 < \kappa \leq 0{,}2$	Ruim
$0{,}2 < \kappa \leq 0{,}4$	Razoável
$0{,}4 < \kappa \leq 0{,}6$	Bom
$0{,}6 < \kappa \leq 0{,}8$	Muito Bom
$0{,}8 < \kappa \leq 1{,}0$	Excelente

O índice *kappa* (κ) é calculado a partir da seguinte fórmula:

$$\kappa = \frac{N\sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r}(x_{i+} * x_{+i})}{N^2 - \sum_{i=1}^{r}(x_{i+} * x_{+i})}, \tag{4.3}$$

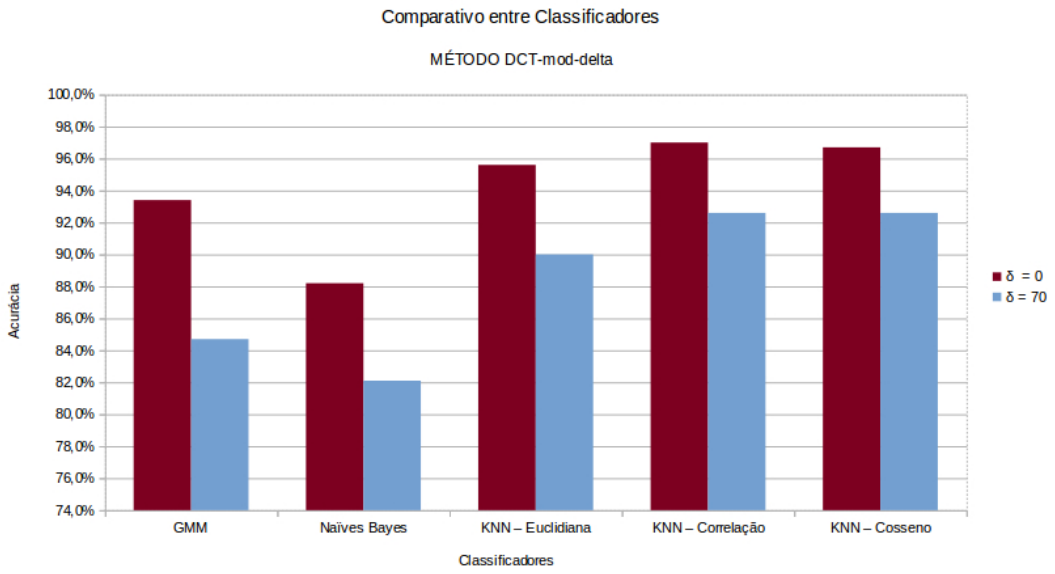
em que N é o número de observações, r o número de linhas da matriz de confusão, x_{ii} os elementos da diagonal principal da matriz de confusão, x_{i+} e x_{+i} o somatório dos elementos da linha i e coluna i, respectivamente, da matriz de confusão.

Na Figura 26, podemos observar que em ambas as variações de iluminação ($\delta = 0$ e $\delta = 70$) as melhores classificações foram com o classificador KNN com a medida de correlação e a distância do cosseno. Podemos comprovar isto com os valores do índice *kappa* da Tabela 8, onde o classificador K-NN na medida de correlação e distância do cosseno obtiveram um índice 0,97 em $\delta = 0$ e 0,92 em $\delta = 70$, muito próximo de 1, assim, apresentando qualidade de classificação excelente conforme a Tabela 7. Comparando os valores obtidos, Tabela 8, com o grau de concordância da Tabela 7, observa-se que os resultados para o índice *kappa* foram para todos excelentes.

Tabela 8 – Índice *kappa* dos Classificadores referente a Figura 26.

CLASSIFICADORES	**ÍNDICE KAPPA**	
	δ **= 0**	δ **= 70**
GMM	0,93	0,84
Naïve Bayes	0,88	0,82
K-NN - Euclidiana	0,95	0,90
K-NN - Correlação	0,97	0,92
K-NN - Cosseno	0,97	0,92

Figura 26 – Comparativo entre os classificadores com o método de extração de características DCT-mod-delta.



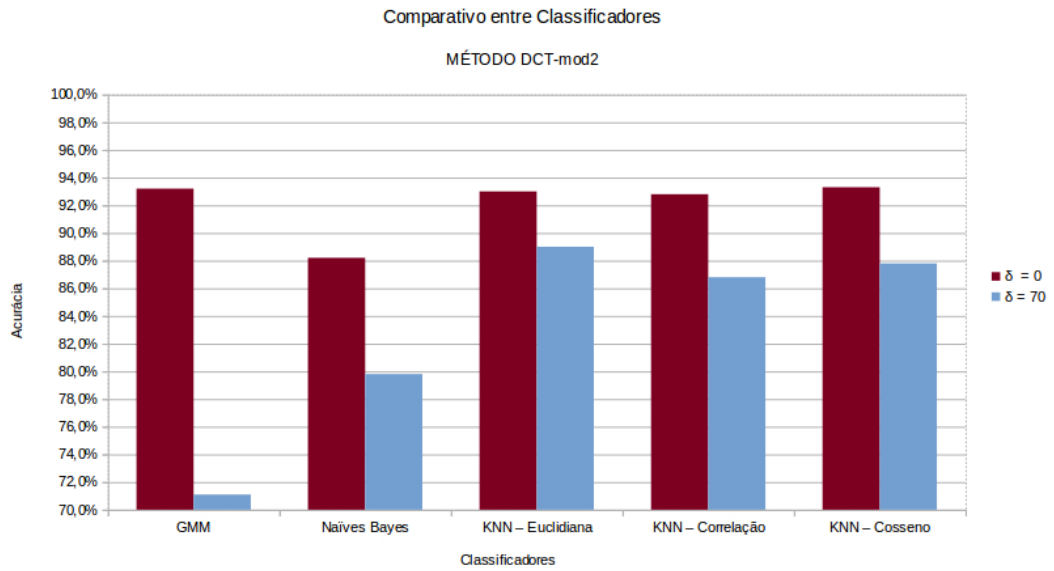
Fonte - Elaborado pela autora.

Analisando a Figura 27, podemos observar que os piores classificadores em relação ao índice *kappa* foram o GMM e Naïve Bayes, principalmente em $\delta = 70$. Mas, quando analisados na Tabela 9, pode-se concluir que os mesmos apresentam qualidade de classificação excelente em $\delta = 0$ e muito bom em $\delta = 70$.

Tabela 9 – Índice *kappa* dos Classificadores referente a Figura 27.

CLASSIFICADORES	**ÍNDICE *KAPPA***	
	δ **= 0**	δ **= 70**
GMM	0,93	0,70
Naïve Bayes	0,88	0,79
K-NN - Euclidiana	0,93	0,89
K-NN - Correlação	0,93	0,86
K-NN - Cosseno	0,93	0,87

Figura 27 – Comparativo entre os classificadores com o método de extração de características DCT-mod2.



Fonte - Elaborado pela autora.

Para o método DCT-mod2 podemos concluir pelas Figura 27 e Tabela 9 que o classificador que obteve o melhor desempenho foi o K-NN com função distância euclidiana, com índice *kappa* 0,93 para $\delta = 0$ e 0,89 para $\delta = 70$, assim, qualificando como um classificador com desempenho excelente.

5 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho implementou um reconhecimento facial robusto a imagens com grandes variações de iluminação. Para extrair as características das faces, bem como reduzir a dimensionalidade dos vetores características, foram utilizados os métodos PCA, DCT e variações da DCT propostas por Sanderson, DCT-mod, DCT-mod-delta e DCT-mod2. Para a classificação das faces foram usados os classificadores GMM com 8 gaussianas, Naïve Bayes e K-NN com as funções de medidas euclidiana, correlação e cosseno. O reconhecimento facial foi realizado usando imagens monocromáticas frontais com as faces completas contendo parte do fundo e não normalizadas. Por meio das análises dos resultados as principais conclusões deste trabalho são as seguintes:

a) É possível observar que as técnicas DCT e PCA em todos os classificadores não se mostraram eficientes mesmo variando a quantidade de *eigenfaces* no PCA, provando não serem apropriadas para o reconhecimento facial em imagens com grandes variações de iluminação.
b) Os métodos DCT-mod-delta e DCT-mod2 foram os que atingiram as melhores acurácias em todos os classificadores mesmo variando a iluminação. Nos resultados de Sanderson, os métodos DCT-mod-delta e DCT-mod2 também foram os que atingiram as melhores acurácias, com o DCT-mod2 sendo o melhor. Diferente, neste trabalho o DCT-mod-delta foi o melhor método, principalmente quando combinado com o classificador K-NN, obtendo acurácia sempre $\geq 90\%$ na pior variação de iluminação ($\delta = 70$) e em todas as funções de medidas (euclidiana, correlação e cosseno). A acurácia obtida por Sanderson com variação de iluminação extrema ($\delta = 70$) foi sempre $> 97\%$, mas, diferente deste trabalho, Sanderson realizou uma verificação facial utilizando imagens normalizadas e cortadas, contendo somente parte da testa, sobrancelhas, olhos e nariz.
c) É possível notar a importância da escolha dos parâmetros no desempenho de algoritmos de classificação mais sofisticados como o GMM. Também é possível notar que um simples algoritmo baseado em instâncias K-vizinhos mais próximos é competitivo e apresenta resultados comprovadamente superiores aos algoritmos de classificação mais sofisticados e que possuem uma quantidade maior de parâmetros.
d) O classificador Naïve Bayes obteve uma boa eficiência computacional mas não foi robusto o suficiente para obter os melhores resultados quando comparado aos resultados dos classificadores GMM e K-NN, isto porque o mesmo depende muito da distribuição das

informações geradas pelos métodos de extração de características, pois o classificador elabora a sua função de decisão a partir da suposição de que os dados possuem uma distribuição normal, então, quanto mais próximos os dados forem de uma distribuição normal, melhor será o desempenho do classificador. Assim, podemos concluir que os métodos de extração utilizados não geraram informações próximas de uma distribuição normal, reduzindo assim a acurácia através do mesmo.

e) O classificador K-NN foi o que obteve a melhor acurácia para todos os métodos de extração de características. Diferente do classificador Naïve Bayes, o classificador K-NN obteve os melhores resultados porque a sua função de decisão não faz suposição através da distribuição dos dados, ao calcular a distância entre os pontos de dados cada atributo terá o mesmo peso, tornado a decisão mais flexível. Mas ao contrário no Naïve Bayes, não possui uma boa eficiência computacional em conjuntos de dados de dimensão elevada.

f) As melhores acurácias para todas as variações de iluminação δ foram obtidas por meio da combinação do método de extração de característica DCT-mod-delta com o classificador K-NN utilizando a medida de correlação. Alcançando 97% em imagens sem variações de iluminação ($\delta = 0$) e 92,6% em imagens com grandes variações de iluminação ($\delta = 70$).

5.1 TRABALHOS FUTUROS

Um dos fatores que podem influenciar na queda da acurácia é a não normalização das imagens, que no caso deste trabalho as mesmas não estão normalizadas. A normalização de imagens, ou alinhamento de imagens é uma etapa importante de pré-processamento, consiste na retirada da variação da posição, rotação e escala entre as imagens das faces. Maioria dos processos de normalização de imagens faciais se baseiam na posição dos olhos, para isso, é fundamental uma detecção precisa para os próximos passos. Com os olhos detectados, inicia-se o alinhamento dos olhos, que consiste da eliminação da inclinação do segmento de reta que une os dois olhos, se houver, e com as coordenadas dos olhos calcula-se a inclinação e aplica a rotação para alinhá-los. O próximo passo é normalizar a escala, deixando todas as imagens com a mesma distância entre os olhos, essa etapa é realizada por meio do redimensionamento da imagem de acordo com um fator de escala. Visando melhorias nos valores das acurácias, um processo de normalização nas imagens utilizadas deste trabalho estão sendo feitas por um aluno por meio de métodos de gradientes.

Visto que o outro grande fator da redução da acurácia nos reconhecimentos faciais

são as variações de iluminação nas imagens, e que estas variações de iluminação são as grandes causadoras das dispersões dos dados dentro da classe, propõe-se combinar os melhores métodos (DCT-mod-delta e DCT-mod2) ao método *Fisherfaces*, que tem como finalidade maximizar a relação de dispersão entre as classes com a dispersão dentro da classe.

REFERÊNCIAS

BELHUMEUR, P. N.; HESPANHA, J. P.; KRIEGMAN, D. J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 19, n. 7, p. 711–720, 1997.

BUCIU, I.; GACSADI, A. Biometrics systems and technologies: A survey. **International Journal of Computers Communications & Control**, v. 11, n. 3, p. 315–330, 2016.

CAMPOS, T. E. **Técnicas de Seleção de Características com Aplicações em Reconhecimento de Faces**. Dissertação (Mestrado) — Universidade de São Paulo, São Paulo, 2001.

CERQUEIRA, P. H. R. **Um estudo sobre reconhecimento de padrões: um aprendizado supervisionado com classificador bayesiano**. Dissertação (Mestrado) — Universidade de São Paulo, Piracicaba, 2010.

CONTRERAS, R. J. **Técnicas de Seleção de Características aplicadas a Modelos Neuro-Fuzzy Hierárquicos BSP**. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2002.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern classification**. [S.l.]: John Wiley & Sons, 2012.

EBRAHIMPOUR, H.; KOUZANI, A. Face recognition using bagging knn. In: **International Conference on Signal Processing and Communication Systems (ICSPCS'2007) Australia, Gold Coast**. [S.l.: s.n.], 2007. p. 17–19.

ER, M. J.; CHEN, W.; WU, S. High-speed face recognition based on discrete cosine transform and rbf neural networks. **IEEE Transactions on Neural Networks**, IEEE, v. 16, n. 3, p. 679–691, 2005.

ER, M. J.; WU, S.; LU, J.; TOH, H. L. Face recognition with radial basis function (rbf) neural networks. **IEEE transactions on neural networks**, IEEE, v. 13, n. 3, p. 697–710, 2002.

GOMATHI, E.; BASKARAN, K. An efficient method for face recognition based on fusion of global and local feature extraction. **IJSCE International Journal of Soft Computing and Engineering**, IJSCE, v. 4, n. 4, p. 56–60, 2014.

GONZALEZ, R. C.; WOODS, R. E. Digital image processing. **Nueva Jersey**, 2008.

JAIN, A. K.; DUIN, R. P. W.; MAO, J. Statistical pattern recognition: A review. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 22, n. 1, p. 4–37, 2000.

JING, X.-Y.; ZHANG, D. A face and palmprint recognition approach based on discriminant dct feature extraction. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, IEEE, v. 34, n. 6, p. 2405–2415, 2004.

JOLLIFFE, I. **Principal component analysis**. [S.l.]: Wiley Online Library, 2002.

KODANDARAM, R.; MALLIKARJUN, S.; KRISHNAMUTHAN, M.; SIVAN, R. Face recognition using truncated transform domain feature extraction. **Int. Arab J. Inf. Technol.**, v. 12, n. 3, p. 211–219, 2015.

LEE, K.; CHUNG, Y.; BYUN, H. Svm-based face verification with feature set of small size. **Electronics Letters**, The Institution of Engineering & Technology, v. 38, n. 15, p. 1, 2002.

LUCCA, F. J. **Implementação Modular da Técnica de Compreesão e Desacompressão JPEG para Imagens**. Dissertação (Mestrado) — Universidade de São Paulo, São Paulo, 1994.

MALHEIRO, R. **Sistemas de Classificação Automática em Géneros Musicais**. Dissertação (Mestrado) — Engenharia Informática, Universidade de Coimbra, 2004.

MARTINS, D. C. J. **Redução de Dimensionalidade Utilizando Entropia Condicional Média Aplicada a Problemas de Bioinformática e de Processamento de Imagens**. Dissertação (Mestrado) — Universidade de São Paulo, São Paulo, 2004.

MATOS, F. M. S. **Reconhecimento Facial Utilizando a Transformada Cosseno Discreta**. Dissertação (Mestrado) — Universidade Federal de Paraíba, João Pessoa, 2008.

OTHMAN, H.; ABOULNASR, T. A separable low complexity 2d hmm with application to face recognition. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 25, n. 10, p. 1229–1238, 2003.

PEDRINI, H.; SCHWARTZ, W. R. **Análise de imagens digitais: princípios, algoritmos e aplicações**. [S.l.]: Thomson Learning, 2008.

PERLOVSKY, L. I. Conundrum of combinatorial complexity. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 20, n. 6, p. 666–670, 1998.

SANDERSON, C. **Biometric person recognition: Face, speech and fusion**. [S.l.: s.n.], 2008. v. 4.

SANDERSON, C.; LOVELL, B. C. Multi-region probabilistic histograms for robust and scalable identity inference. In: SPRINGER. **International Conference on Biometrics**. [S.l.], 2009. p. 199–208.

SANDERSON, C.; PALIWAL, K. K. Robust face-based identity verification. In: CITESEER. **Proc. Microelectronic Engineering Research Conf**. [S.l.], 2001.

SHERMINA, J. Illumination invariant face recognition using discrete cosine transform and principal component analysis. In: IEEE. **Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on**. [S.l.], 2011. p. 826–830.

SILVA, S. S. **Segmentação de Imagens Utilizando Combinação de Modelos de Misturas Gaussianas**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, Recife, 2014.

SOONG, F. K.; ROSENBERG, A. E. On the use of instantaneous and transitional spectral information in speaker recognition. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, IEEE, v. 36, n. 6, p. 871–879, 1988.

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

VAIDEHI, V.; BABU, N. N.; AVINASH, H.; VIMAL, M.; SUMITRA, A.; BALMURALIDHAR, P.; CHANDRA, G. Face recognition using discrete cosine transform and fisher linear discriminant. In: IEEE. **Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on**. [S.l.], 2010. p. 1157–1160.

WEBB, A. R. **Statistical pattern recognition**. [S.l.]: John Wiley & Sons, 2011.

ZHANG, H.; WEN, T.; ZHENG, Y.; XU, D.; WANG, D.; NGUYEN, T. M.; WU, Q. J. Two fast and robust modified gaussian mixture models incorporating local spatial information for image segmentation. **Journal of Signal Processing Systems**, Springer, v. 81, n. 1, p. 45–58, 2015.

ZHAO, W.; CHELLAPPA, R.; PHILLIPS, P. J.; ROSENFELD, A. Face recognition: A literature survey. **ACM computing surveys (CSUR)**, ACM, v. 35, n. 4, p. 399–458, 2003.

ZHAO, W.; KRISHNASWAMY, A.; CHELLAPPA, R.; SWETS, D. L.; WENG, J. Discriminant analysis of principal components for face recognition. In: **Face Recognition**. [S.l.]: Springer, 1998. p. 73–85.

APÊNDICES

APÊNDICE A – Análise de Componentes Principais

Segundo Jolliffe (JOLLIFFE, 2002), a ideia central da Análise de Componentes Principais (PCA, do inglês *Principal Component Analysis*) é reduzir a dimensionalidade de um conjunto de dados que consiste de um grande número de variáveis inter-relacionadas, enquanto mantém a variação presente do conjunto de dados.

Considerando o conjunto de dados $x = \{x_1, x_2, ..., x_p\} \subset \mathbb{R}^d$, a PCA projeta os elementos de x em novas direções ortogonais $z_1, z_2, ..., z_d$, onde a primeira componente tem a maior variância dentre os dados projetados, a segunda componente a segunda maior variância, e assim, sucessivamente. Dessa maneira, a PCA permite manusear um conjunto de dados originais de forma mais fácil em um espaço mais simples de se observar, mantendo a maior parte da sua variabilidade e, facilitando assim, trabalhar com um número de variáveis muito grande, pois, não é uma tarefa simples e nem muito útil.

A redução de dimensionalidade de um conjunto de dados através da PCA é realizada encontrando suas componentes principais. E o primeiro passo é obter uma transformação linear α_1^T que possua variância máxima ao ser aplicada ao vetor x, composto por p variáveis aleatórias. Ou seja:

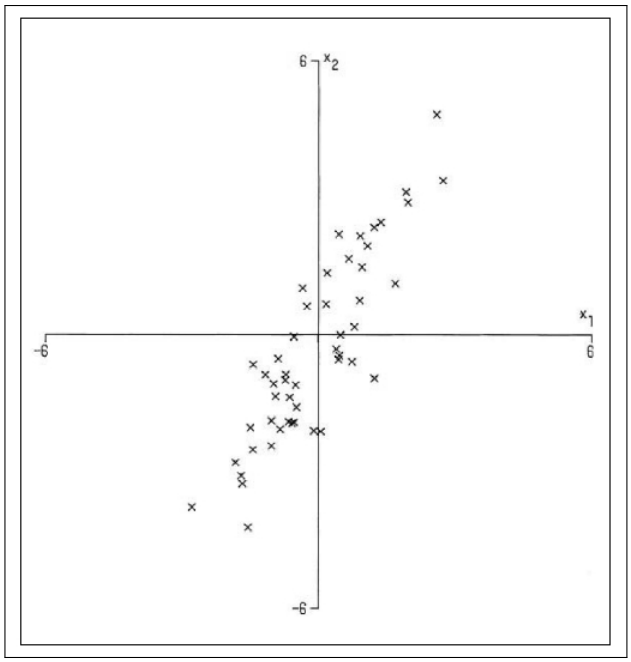
$$\alpha_1^T x = \alpha_{11} x_1 + \alpha_{12} x_2 + ... + \alpha_{1p} x_p = \sum_{j=1}^{p} \alpha_1 x_j \,. \tag{A.1}$$

Na Equação A.1, a transformação linear α_1^T realiza uma combinação linear ponderada dos elementos de x, sendo que α_1 precisa ser escolhido de tal maneira que a variância da resultante seja igual a variação máxima de x. Posteriormente, busca-se por outra função $\alpha_2 x$ que seja não correlacionada com $\alpha_1^T x$ e que, quando aplicada aos elementos do vetor x, crie um nova variável aleatória que seja não correlacionada com $\alpha_1^T x$, apresentando a segunda maior variância possível. Esse processo é repetido para $\alpha_3 x$, $\alpha_4 x$, ..., $\alpha_i x$, onde $\alpha_i x$ é a i-ésima componente principal. O máximo de componentes principais que pode ser encontrado é p (a quantidade de variáveis aleatórias), sendo que a maior parte da variância das variáveis contidas em x pode ser representada por uma quantidade de m componentes principais, sendo $m \ll p$ (JOLLIFFE, 2002).

Na Figura 28, é apresentado um conjunto de dados bidimensionais, x_1 e x_2 ($p = 2$), contendo 50 exemplos. É possível observar o quanto as duas variáveis x_1 e x_2 são altamente correlacionadas. Existe uma considerável variância dos dados em ambas as direções x_1 e x_2,

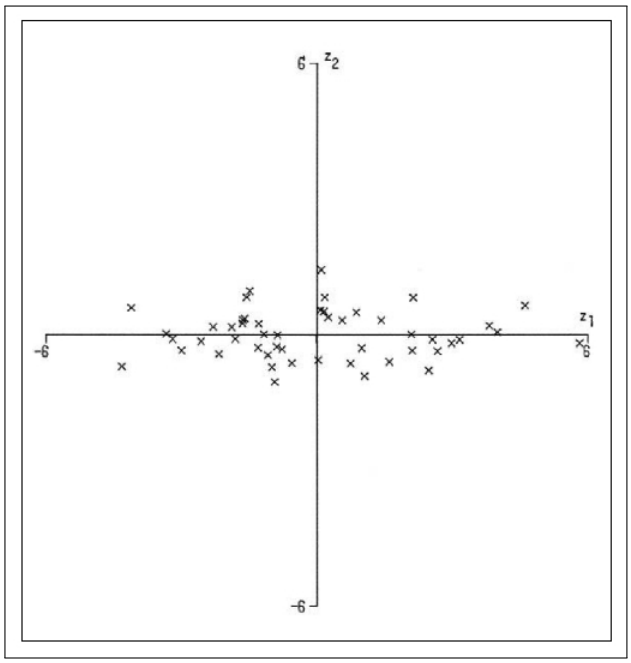
embora um pouco mais no sentido de x_2 do que x_1. Porém, se esses dados forem transformados para um espaço contendo somente as suas componentes principais z_1 e z_2, podemos ver que a componente z_1 apresenta uma variância maior que qualquer uma das variáveis originais, visto que essa direção é a que possui a maior variância de todo o conjunto. Na Figura 29, podemos observar o conjunto de dados sendo representado no espaço das componentes principais (JOLLIFFE, 2002).

Figura 28 – Conjunto de dados contendo duas variáveis originais (x_1 e x_2).



Fonte – Figura retirada de (JOLLIFFE, 2002).

Figura 29 – Conjunto de dados representado no espaço das componentes principais.



Fonte – Figura retirada de (JOLLIFFE, 2002).

Definidas as componentes principais, agora é necessário saber como encontrá-las. Para encontra as componentes principais de um vetor x contendo p variáveis aleatórias, calcula-se a matriz de covariância dessas variáveis. Na matriz de covariância Σ seu (i, j)-ésimo elemento expressa a covariância entre as variáveis i e j do vetor x. Quando $i = j$, este elemento é a variância da i-ésima variável.

Determinada a matriz de covariância Σ, a primeira componente principal $z_1 = \alpha_1^T x$ é determinada pelo autovetor α_1 de Σ associado ao seu maior autovalor λ_1, a segunda componente principal $z_1 = \alpha_2^T x$, pelo autovetor α_2 associado ao segundo maior autovalor λ_2 e, assim, sucessivamente. Assim sendo, para encontrarmos as componentes principais de um conjunto de dados, é necessário primeiro buscar pela primeira componente $\alpha_1^T x$. Para isso é preciso encontrar um vetor α_1 que maximize a sua variância, isto é:

$$max \quad \alpha_1^T \Sigma \alpha_1 \,, \tag{A.2}$$

sujeito a

$$\alpha_1^T \alpha_1 = 1 \,. \tag{A.3}$$

Jolliffe (JOLLIFFE, 2002) apresenta uma abordagem para a solução desse problema de otimização usando a técnica dos multiplicadores de Lagrange. Assim, o problema passar a ser de maximizar:

$$\alpha_1^T \Sigma \alpha_1 - \lambda_1 (\alpha_1^T \alpha_1 - 1),$$

onde λ_1 é um multiplicador da Lagrange.

Fazendo a diferenciação em relação a α_1 e igualando a zero, obtemos:

$$\Sigma \alpha_1 - \lambda_1 \alpha_1 = 0. \tag{A.4}$$

Se isolarmos α_1 da Equação A.4, temos a seguinte equação:

$$(\Sigma - \lambda_1 I_p)\alpha_1 = 0, \tag{A.5}$$

onde I_p é a matriz identidade $p \times p$, λ_1 é um autovalor de Σ e α_1, o seu autovetor correspondente.

Para decidir qual dos p autovetores proporciona a $\alpha_1^T x$ a maior variância possível, é preciso considerar que λ_1 deve ser o maior possível, levando em consideração que:

$$\alpha_1^T \Sigma \alpha_1 = \alpha_1^T \lambda_1 \alpha_1 = \lambda_1 \alpha_1^T \alpha_1 = \lambda_1,$$

Assim, α_1 é o autovetor correspondente ao maior autovalor de Σ e a $\text{Var}(\alpha_1^T x) = \alpha_1^T \Sigma \alpha_1 = \lambda_1$ é esse maior autovalor.

A i-ésima componente principal de x é $\alpha_i^T x$ e $\text{Var}(\alpha_i^T x) = \lambda_i$ onde i-ésimo maior autovalor de Σ e α_i é o autovetor correspondente.

A segunda componente principal, $\alpha_2^T x$ maximiza:

$$\alpha_2^T \Sigma \alpha_2, \tag{A.6}$$

sujeito a ser não correlacionada com $\alpha_1^T x$, ou equivalentemente sujeito a:

$$\mathrm{Cov}(\alpha_1^T x, \alpha_2^T x) = 0\,. \tag{A.7}$$

Sendo que $\mathrm{Cov}(x,y)$ denota a covariância entre as variáveis x e y, a Equação A.7 pode ser reescrita como mostra a Equação A.8:

$$\mathrm{Cov}(\alpha_1^T x, \alpha_2^T x) = \alpha_1^T \Sigma \alpha_2 = \alpha_2^T \Sigma \alpha_1 = \alpha_2^T \lambda_1 \alpha_1 = \lambda_1 \alpha_2^T \alpha_1 = \lambda_1 \alpha_1^T \alpha_2\,. \tag{A.8}$$

Assim, qualquer uma das Equações A.9, A.10, A.11 ou A.12 pode ser usada para informar que não existe correlação entre $\alpha_1^T x$ e $\alpha_2^T x$:

$$\alpha_1^T \Sigma \alpha_2 = 0\,, \tag{A.9}$$

$$\alpha_2^T \Sigma \alpha_1 = 0\,, \tag{A.10}$$

$$\alpha_1^T \alpha_2 = 0\,, \tag{A.11}$$

$$\alpha_2^T \alpha_1 = 0\,, \tag{A.12}$$

Para encontrar o segundo maior autovalor e seu autovetor correspondente, usa-se novamente o multiplicador de Lagrange para maximizar a equação:

$$\alpha_2^T \Sigma \alpha_2 - \lambda_2(\alpha_2^T \alpha_2 - 1) - \phi\, \alpha_2^T \alpha_1\,, \tag{A.13}$$

onde λ_2 e ϕ são os multiplicadores de Lagrange.

Diferenciando a Equação A.13 em relação a α_2, temos a equação:

$$\Sigma \alpha_2 - \lambda_2 \alpha_2 - \phi\, \alpha_1 = 0\,. \tag{A.14}$$

Multiplicando na esquerda por α_1^T, obtemos a equação:

$$\alpha_1^T \Sigma \alpha_2 - \lambda_2 \alpha_1^T \alpha_2 - \phi \alpha_1^T \alpha_1 = 0\,. \tag{A.15}$$

Sabendo que os dois primeiros termos são zero e que $\alpha_1^T \alpha_1 = 1$, então concluímos que $\phi = 0$. Desse modo, $\Sigma\alpha_2 - \lambda_2\alpha_2 = 0$, ou equivalentemente $(\Sigma\lambda_2 I_p)\alpha_2 = 0$, assim, λ_2 é novamente um autovalor de Σ sendo α_2 seu autovetor correspondente.

Novamente, $\lambda_2 = \alpha_2^T \Sigma \alpha_2$, assim λ_2 deve ser o maior possível. Assumindo que Σ não apresenta autovalores repetidos λ_2 e não pode ser igual a λ_1. Se isso fosse possível, indicaria que $\alpha_2 = \alpha_1$, e isso desobedeceria a restrição $\alpha_1^T \alpha_2 = 0$. Assim, λ_2 é o segundo maior autovalor e α_2 o seu autovetor correspondente.

Jolliffe (JOLLIFFE, 2002) apresentou desta forma como encontrar o primeiro e segundo autorvetores, provando que os coeficientes $\alpha_3, \alpha_4, ..., \alpha_p$ são os autovetores de Σ correspondendo aos autovetores $\lambda_3, \lambda_4, ..., \lambda_p$.