FERNANDO LUIS GUTIÉRREZ LÓPEZ

Método não intrusivo de detecção de fraudes em ataques de suplantação de identidade por reconhecimento facial

São Paulo

2023

FERNANDO LUIS GUTIÉRREZ LÓPEZ

Método não intrusivo de detecção de fraudes em ataques de suplantação de identidade por reconhecimento facial

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Ciências.

São Paulo

2023

FERNANDO LUIS GUTIÉRREZ LÓPEZ

Método não intrusivo de detecção de fraudes em ataques de suplantação de identidade por reconhecimento facial

Versão Corrigida

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Ciências.

Área de Concentração:
Engenharia de Computação

Orientadora:
Profª. Dra. Graça Bressan

São Paulo
2023

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 24 de Julho de 2023

Assinatura do autor: 

Assinatura do orientador: 

Catalogação-na-publicação

LOPEZ, FERNANDO
Método não intrusivo de detecção de fraudes em ataques de suplantação de identidade por reconhecimento facial / F. LOPEZ -- versão corr. -- São Paulo, 2023.
99 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.Liveness 2.Deep Learning 3.Machine Learning I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II.t.

Dedicatória

O resultado desse trabalho é dedicado a meus pais por todo o amor, compreensão e dedicação durante a minha vida toda. Por ser minha inspiração em cada passo.

Los amo.

AGRADECIMENTOS

Primeiramente gostaria de agradecer a minha orientadora Graça pela confiança, apoio e ajuda em cada momento durante todos esses anos. Pela guia, conselhos e colaboração nesse trabalho e na minha formação como melhor profissional. Grato

Ao Brasil pela acolhida, e especialmente a Universidade de São Paulo pela oportunidade de continuar meus estudos e fornecer todos os recursos para a conclusão desse trabalho durante o período todo.

Agradeço muito ao Laboratório de Arquitetura e Redes de Computadores (LARC) onde criei uma nova família de amigos e professores, onde vivi momentos incríveis pessoal e profissionalmente e tive sempre o apoio de cada um de seus colaboradores.

À Fundação de Apoio à Universidade de São Paulo (FUSP) e à Fundação para o desenvolvimento tecnológico da Engenharia (FDTE) pelo apoio financeiro.

Agradecer aos grandes colaboradores desse trabalho Claudia de Armas e José Carlos Gutiérrez por todo o apoio, conselhos, ideias e o mais valioso, o seu tempo.

Finalmente gostaria de agradecer a todos os professores, família e amigos que de uma forma ou outra estiveram presentes durante toda minha vida me encaminhando até este resultado.

Obrigado a todos.

RESUMO

Com os novos avanços tecnológicos de hardware e software e a sua implementação na área da biometria facial, a segurança é um fator crítico a garantir. Além das dificuldades que podem ter os algoritmos de reconhecimento facial como a oclusão do rosto, idade e similaridade das pessoas, também têm que lidar com ataques de suplantação de identidade. Indivíduos mal-intencionados tentam burlar os sistemas fazendo uso de máscaras, imagens e vídeos de outras pessoas com o objetivo de roubar sua identidade. Visando resolver esses problemas, o objetivo deste trabalho é propor um método não intrusivo para evitar ataques de suplantação de identidade em sistemas de autenticação por reconhecimento facial.

Este trabalho faz uma pesquisa atual dos principais conceitos relacionados a prova de vida e algoritmos *anti-spoofing*, instrumentos de ataques e os métodos mais revolucionários dos últimos anos. Também é feito um levantamento dos principais bancos de dados de livre acesso para treinamentos e testes. São analisados os trabalhos com os resultados mais relevantes na área baseado em métodos não intrusivos e que não contemplam *hardware* externo ao sistema. Além disso é realizada a proposta de um método que prevê atingir resultados satisfatórios a partir dos trabalhos relacionados.

Finalmente foi selecionado o uso de algoritmos *anti-spoofing* baseados em *software* e foram escolhidos os bancos de dados Replay Attack e SiW. Foi definido o método proposto que contempla três etapas: o pré-processamento das imagens baseado nos canais de croma; o treinamento da rede através de uma arquitetura *Deep Learning* siamesas e uma função *Triplet Loss;* e por fim, a classificação da imagem real ou falsa mediante o *Support Vector Machine*. Para finalizar foi definida a métrica HTER para validar o sistema, assim como os trabalhos relacionados que serão utilizados para compará-las.

Palavras – chave: Anti-spoofing, Deep Learning, SVM, PAI, PAD, método não intrusivo.

ABSTRACT

With new technological advances in hardware and software and their implementation in facial biometrics, security is a critical factor to guarantee. However, in addition to the difficulties that recognition algorithms can have, such as face occlusion, age and similarity of people, they also have to deal with identity supplanting attacks. Moreover, malicious individuals try to circumvent systems by using other people's masks, images, and videos to steal their identities. Aiming to solve these problems, this work proposes a non-intrusive method to avoid identity impersonation attacks in facial recognition authentication systems.

This work makes current research on the main concepts related to proof of life and anti-spoofing algorithms, attack instruments and the most revolutionary methods of the last years. A survey of the main freely accessible databases for training and testing is also carried out. Works with the most relevant results in the area based on non-intrusive methods that do not include hardware external to the system are analyzed. In addition, a proposal is made for a method that aims to achieve satisfactory results from the related works.

Finally, software-based anti-spoofing algorithms were selected, and the Replay Attack and SiW databases were chosen. Next, the proposed method was defined, which includes three steps: pre-processing images based on chroma channels; network training through a Siamese Deep Learning architecture and a Triplet Loss function; and finally, the classification of the real or fake image using the Support Vector Machine. Finally, the HTER metric was defined to validate the system and the related works that will be used to compare them.

Keywords: Anti-spoofing, Deep Learning, SVM, PAI, PAD, non-intrusive method

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACER	Average Classification Error Rate
APCER	Attack Presentation Classification Error Rate
AUC	Area Under Curve
BPCER	Bona Fide Presentation Classification Error Rate
BSIF	Binarized Statistical Image Features
CER	Crossover Error Rate
CoALBP	Co-occurrence of Adjacent Local Binary Patterns
CNN	Convolutional Neural Network
DoG	Difference of Gaussians
DTN	Deep Tree Network
EER	Equal Error Rate
FAR	False Acceptance Rate
FP	False Positives
FPR	False Positive Rate
FN	False Negatives
FNR	False Negative Rate
FRR	False Rejection Rate
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
HTER	Half Total Error Rate
IA	Inteligência Artificial
LBP	Local Binary Patterns
LDA	Linear Discriminant Analysis
LPQ	Local Phase Quantization
MLP	Multi-layer Perceptron
PAD	Presentation Attack Detection

PAI	Presentation Attack Instrument
PCA	Principal Component Analysis
ReLu	Rectified Linear Unit
ResNet	Residual Neural Network
RNA	Rede Neural Artificial
ROC	Receiver Operating Characteristic
RPPG	Remote Photoplethysmography
SID	Scale-Invariant Descriptor
SNN	Siamese Neural Network
SOTA	State-of-the-Art
SVM	Support Vector Machines
ZSFA	Zero-Shot Face Anti-spoof

SUMÁRIO

Capítulo 1

1 Introdução

Há cada vez mais ataques a sistemas de segurança que usam os dados biométricos como mecanismo de proteção. O uso de características físicas ou comportamentais de cada indivíduo tornou-se difícil de imitar e muitos setores começaram a usá-lo para proteger as suas informações, autenticar ou identificar usuários (ALSAADI, 2015). Os avanços tecnológicos, tanto de software quanto de hardware, estão abrindo novas portas para a implementação de sistemas de controle de acesso a prédios, fronteiras, telefones celulares, computadores e aplicativos. Devido à essa propagação, mais e mais tentativas de ataques são identificadas, a fim de suplantar a identidade da pessoa.

Atualmente, o reconhecimento facial é considerado um dos meios de identificação biométrica mais utilizados; resposta rápida, implementação relativamente fácil e interação não física da pessoa, atrai a atenção de muitas empresas e aplicativos (MASI et al., 2018). Sua estreita conexão com os algoritmos de detecção de rosto, na maioria dos casos, incentiva a análise desses recursos faciais em mídias dinâmicas com grande quantidade de pessoas embora a sua confiabilidade diminua em relação aos ambientes controlados, fator positivo na busca de suspeitos. A análise e o processamento das etapas de identificação e autenticação em grandes volumes de dados e que exigem uma resposta em tempo real, só podem ser atendidos por sistemas de computadores, atualmente é descartada a possibilidade de ser monitorada por um ser humano e atingir valores significativos (VINAY et al., 2015). A grande maioria dos ataques atuais a esses sistemas ocorrem através do uso de imagens não reais; imagens impressas, vídeos, máscaras que conseguem violar e roubar a identidade do indivíduo (RAMACHANDRA; BUSCH, 2017), algo que o olho humano consegue perceber. Mas a necessidade de analisar milhões de dados no menor tempo possível coloca nas mãos da inteligência artificial a tarefa de deteção para impedir que tais tentativas ocorram.

O conceito de prova de vida está se tornando essencial quando se trata de sistemas seguros que implementam reconhecimento facial (KIM; SUH; HAN, 2015). A deteção da vida é simplesmente identificar se determinado indivíduo é quem diz, por muito simples que pareça, tornou-se um desafio entre os desenvolvedores para evitar a falsificação. Esse conceito foi dividido na literatura em dois grandes grupos, os meios intrusivos de verificação (ALI; DERAVI; HOQUE, 2012); (TANG et al., 2018) e os não intrusivos (KOLLREIDER; FRONTHALER; BIGUN, 2009); (ALOTAIBI; MAHMOOD, 2017); (KIM et al., 2012)**.**

O primeiro grupo mencionado também conhecido como de resposta sob desafios (*challenge response)* refere-se ao uso de determinadas ações, como piscar ou o movimento dos olhos, para serem realizadas a fim de verificar a identidade. Um segundo grupo não intrusivo é mais orientado para a detecção de outras características como profundidade, textura, frequência e desfoque em imagens e vídeos. Imagens impressas ou capturadas da tela de dispositivos, na melhor das hipóteses, diferem bastante das características acima mencionados em relação aos rostos reais (BOULKENAFET; KOMULAINEN; HADID, 2015a). Mas a imensa variedade de amostras que podem ser obtidas em cenários com diferentes condições de luminosidade, qualidade de impressão, resolução de tela e dispositivos, reduziu significativamente a probabilidade de encontrar grandes diferenças que facilitem a sua correta identificação. Portanto, um dos principais desafios tem sido criar bancos de dados com essa grande variedade de ambientes.

Atualmente, não existem muitos bancos de dados disponíveis; ou estão focados em um ataque específico, ou a quantidade de dados não é representativa para executar o aprendizado de máquina ou são privados (BOULKENAFET; KOMULAINEN; HADID, 2015a) (AKBULUT et al., 2017); (LIU; JOURABLOO; LIU, 2018). Ambientes muito controlados é outra das características que garantem melhores resultados, mas restringe o campo de uso. Em muitas ocasiões, as imagens desses repositórios de treinamento não têm a qualidade adequada: brilho baixo ou excessivo, desfoque ou baixa resolução são alguns exemplos que, durante o processo

de aprendizado, inserem valores falsos (SUBASIC et al., 2005); (UNNIKRISHNAN; ESHACK, 2016).

Analisar grandes volumes de dados, extrair informações e identificar padrões sem intervenção humana é atualmente uma tarefa atribuída a mecanismos de aprendizado automatizado, como *Machine Learning* e *Deep Learning*. É necessária uma grande quantidade de dados para treinamento, calibração e teste a fim de fazer uma previsão com base no conhecimento já adquirido. Por tais razões, com o desenvolvimento dos algoritmos de aprendizado de máquina, o conceito de prova de vida, pode ser desenvolvido até conseguir melhores resultados.

1.1 Problema

O uso incremental da biometria facial estimulou o aparecimento de novos mecanismos de ataque para violar esses sistemas. O reconhecimento facial foi expandido para aplicações diferentes, como a verificação de documentos de identidade, acesso a fronteiras e instituições, aplicativos bancários e a resposta dos invasores na tentativa de burlar esses sistemas de verificação não foi lenta.

Na Rússia, Grigory Bakunov desenvolveu um algoritmo que cria uma maquiagem especial para enganar um software de reconhecimento. No entanto, ele decidiu não levar seu produto ao mercado depois de perceber a facilidade com que poderia ser usado por criminosos como explica Zavyalova (ZAVYALOVA, 2017). No final de 2017, a empresa vietnamita Bkav Corporation (BKAV'S CORPORATION, 2017) usou com sucesso uma máscara para invadir o recurso de reconhecimento facial Face ID do iPhone X da Apple. Quase ao mesmo tempo, pesquisadores de uma empresa alemã revelaram um método que permitia ignorar a autenticação facial do Windows 10 Hello, imprimindo uma imagem facial infravermelha. A empresa de cibersegurança Forcepoint alerta que o uso de impressões 3D através de uma fotografia pode fazer uma 'máscara' do rosto e, assim, violar o sistema de autenticação facial. Um relatório desta empresa destaca que, em 2016, especialistas em visão computacional e segurança da Universidade da Carolina do Norte (EUA) enganaram

sistemas de reconhecimento facial usando fotos e vídeos digitais públicos, disponíveis em redes sociais e motores de busca. Os sistemas de deteção de vida resolvem alguns problemas muito sérios, por exemplo, o mecanismo de prova de vida de Facebook permitiu eliminar 5,4 bilhões de contas falsas apenas em 2019 (FUNG; GARCIA, 2019).

O foco deste trabalho são as tentativas de ataques por roubo de identidade e não outros fatores que influenciam negativamente no reconhecimento, como oclusão facial, distância, idade da pessoa, similaridade, entre outros. Portanto, nosso principal problema está em verificar que a imagem capturada para identificar uma pessoa, não seja tomada de uma fotografia impressa ou tela de computador.

1.2 Objetivo

O objetivo deste trabalho é propor um método não intrusivo para evitar ataques de suplantação de identidade em sistemas de autenticação por reconhecimento facial.

Como objetivos específicos:

- Análise dos principais bancos de dados disponíveis para treinamento e teste de sistemas de suplantação de identidade facial.
- Implementação dos métodos necessários para normalização dos dados de treinamento e teste.
- Análise das principais técnicas utilizadas na extração e classificação de características para selecionar as mais adequadas.
- Implementar o método proposto em um cenário controlado para fazer os testes e avaliar o sistema.
- Realizar uma análise dos resultados alcançados e comparar com os trabalhos relacionados.

1.3 Justificativa e Motivação

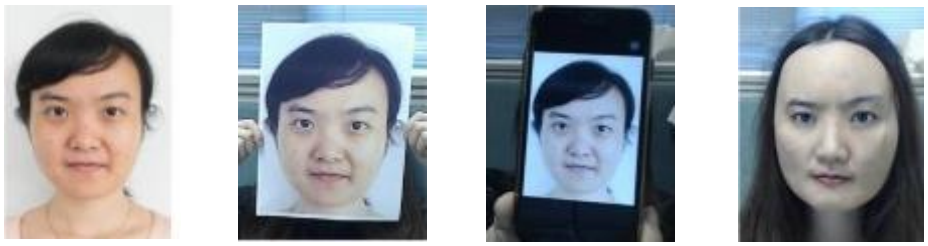
O aumento das pesquisas na área de computação e os avanços de sistemas de hardware e software tem propiciado que técnicas e métodos sejam criados para aumentar a segurança em sistemas computacionais. Atualmente existem conceitos e algoritmos que permitem criar estruturas mais complexas para solucionar os problemas que serão abordados nesta pesquisa.

O conceito de prova de vida tem sido amplamente estudado com o surgimento de mecanismos de autenticação e identificação de pessoas através de características biométricas. Os ataques para violar esses sistemas estão se desenvolvendo juntamente com o próprio sistema e é cada vez mais importante evitar que eles ocorram. O reconhecimento facial é o meio mais natural de identificação biométrica e não requer contato com a pessoa, de modo que sua aplicação está ganhando espaço entre os outros mecanismos biométricos.

Os algoritmos de reconhecimento facial são dedicados a encontrar características semelhantes em uma pessoa e realizar a sua identificação ou autenticação em um determinado sistema (BRUCE; YOUNG, 1986). Saber que a pessoa que está executando este procedimento é uma pessoa real, ou uma fotografia dela, um vídeo ou uma máscara 3D está fora do alcance deles. Os avanços tecnológicos têm propiciado grandes melhorias na impressão e resolução de imagens, qualidade de vídeo e impressões 3D, sendo que começaram permitiram violar sistemas que apenas contemplavam sua segurança através do reconhecimento facial (ERDOGMUS; MARCEL, 2014). Os invasores podem encontrar facilmente fotos de usuários legítimos visitando sites de redes sociais como Facebook, Instagram ou sites pessoais ou profissionais. Eles podem usar essas fotos mediante as abordagens apresentadas na Figura 1-1 para atacar o sistema de reconhecimento facial. Desde então, foi necessário começar a implementar soluções que pudessem detectar ataques desse tipo. A prova de vida é aplicada dentro desses sistemas a partir de vários mecanismos, entre eles os desafios que exigem ao usuário realizar algum tipo de movimento, ação ou observar comportamentos causados por estímulos

(FRISCHHOLZ; WERNER, 2003); (MARSICO et al., 2012); (ALI; DERAVI; HOQUE, 2013); (SMITH; WILIEM; LOVELL, 2015) . Nos últimos anos, essas técnicas foram perdendo aceitação pelo fato do usuário ter que colaborar com o sistema. Isso deu lugar a mecanismos não intrusivos capazes de, sem a interação do usuário, poder avaliar e por fim obter resultados significativos.

Figura 1-1: Diferentes abordagens para enganar aos sistemas de reconhecimento facial.



(a) *(b)* *(c)* *(d)*

(a) Imagem real
(b) Imagem impressa
(c) Imagem na tela
(d) Máscara 3D.
Fonte: (YUEN, 2019)

Na prova de vida, mecanismos não intrusivos lideram as investigações dos últimos anos. Foi demonstrado que esse tipo de esquema possui alta precisão e um custo relativamente baixo. Além disso, esses esquemas não requerem cooperação do usuário e também evitam a necessidade de hardware especializado Mecanismos de detecção de movimento ocular ou piscar de olhos (PAN et al., 2007), oferecem bons resultados na detecção de fraude por imagens impressas mas podem não detetar a fraude no caso de uso de vídeos e máscaras com abertura nos olhos como mostrado na Figura 1-2.

Figura 1-2: Imagem impressa com furos na área dos olhos para violar sistemas de verificação por meio de detecção de movimento ocular.

Fonte: (RAMACHANDRA; BUSCH, 2017).

Outra abordagem bem utilizada foi a análise de textura nas imagens: reflexo, sombra, pigmentos de imagens reais diferem em seus valores das impressões, capturas de tela e máscaras 3D. O uso de *Local Binary Patterns* (LBP) é o método mais usado de análise de microtextura e pode dar uma diferença qualitativa nos padrões de textura existentes nas imagens (MÄÄTTÄ; HADID; PIETIKÄINEN, 2011); (CHINGOVSKA; ANJOS; MARCEL, 2012). O uso de LBP junto com outros descritores fornece, na maioria das vezes, os melhores resultados, no entanto, podem produzir texturas de papel e impressões muito variadas, e os sistemas baseados na análise de textura devem ser robustos a diferentes padrões de textura que exigem um conjunto muito diversificado de dados. Assim como a análise de textura, outra abordagem é o uso de análise de frequência usando a transformada de Fourier (LI et al., 2004) em que os autores sugerem que fotografias fraudulentas têm menos componentes de alta frequência do que as reais. A fusão entre análise de textura e frequência contribuiu para a melhoria dos classificadores antifalsificação (KIM et al., 2012a), mas da mesma forma a necessidade de dados diversos, dispositivos de captura e os ambientes dificulta a identificação em muitos casos.

Os bancos de dados atuais e de acesso gratuito usados para treinamento e teste concentram-se principalmente em algum tipo de ataque específico: de imagens impressas, fotos tomadas nas telas de dispositivos, vídeos ou máscaras 3D. Além disso, de bancos de dados com cenários controlados de luminosidade, com câmeras de boa qualidade e com distância fixa pode-se obter melhores resultados do que bancos de dados com muita variedade de cenários e dispositivos de captura.

Características como faces de perfil ou faces tortas e baixa qualidade da imagem podem ser encontradas nesses bancos de dados sendo que podem influenciar com valores indesejados durante o treinamento.

Recentemente, arquiteturas como *Deep Learning* surgiram como boas alternativas para resolver problemas complexos e alcançaram melhores resultados em muitas tarefas devido ao seu grande poder de abstração e robustez, trabalhando com características de alto nível e autodidatas (CANZIANI; CULURCIELLO; PASZKE, 2017). Dentro de prova de vida, vários trabalhos de sucesso empregam estas arquiteturas (YANG; LEI; LI, 2014); (ALOTAIBI; MAHMOOD, 2017); (BOTELHO DE SOUZA et al., 2018); (LIU; JOURABLOO; LIU, 2018), e serão utilizados como ponto de referência para a criação da proposta final deste trabalho. No capítulo 3, nos trabalhos relacionados será explicado de forma mais detalhada os alcances e resultados das pesquisas relacionadas.

1.4 Método

Visando desenvolver o objetivo definido foi planejada a seguinte metodologia:

- Fazer uma pesquisa exploratória sobre os conceitos de prova de vida, reconhecimento da face, inteligência artificial, *machine learning, deep learning*, entre outros relevantes nesta pesquisa.
- Estudo dos principais bancos de dados de acesso livre dedicados ao treinamento e testes de sistemas *anti-spoofing.*
- Definir o mecanismo *anti-spoofing* a utilizar.
- Estudar os trabalhos relacionados para identificar o estado da arte e encontrar desafios, problemas e lacunas.
- Definir a arquitetura a implementar através do estudo dos trabalhos relacionados focado no mecanismo *anti-spoofing* selecionado.
- Serão utilizados filtros implementados no Python que vai selecionar as imagens do banco de dados para treinamentos, testes. Além disso será feita uma parametrização dos dados.

- Implementar o método proposto.
- Avaliar o método fazendo a comparação com os resultados alcançados nos trabalhos relacionados. Será avaliada a métricas HTER.

1.5 Estrutura do trabalho

O trabalho de pesquisa está estruturado em vários capítulos e seções para facilitar a sua leitura e compreensão.

Nesse primeiro capítulo foi feita uma introdução ao tema, e a discussão do problema, objetivos e a justificativa e motivação do trabalho. No capítulo definiu-se o contexto da pesquisa, relevância, atualidade, assim como a importância de estudar e pesquisar sobre o tema que será tratado.

O segundo capítulo tem como objetivo fundamentar teoricamente os conceitos que serão utilizados neste trabalho, assim como contextualizar outros que são relacionados ao tema. Alguns desses conceitos são: biometria, reconhecimento facial, *Machine Learning, Deep Learning,* prova de vida etc.

O terceiro capítulo descreve os trabalhos relacionados sendo analisados de forma detalhada e fazendo a extração de dados relevantes para serem comparados com os resultados desta pesquisa.

O quarto capítulo define o sistema proposto a partir da análise dos trabalhos relacionados.

O quinto capítulo implementa o método proposto assim como a realização do treinamento com imagens e por fim, a validação do método em relação aos resultados dos trabalhos relacionados selecionados.

O sexto e último capítulo vai conter as principais conclusões e considerações, assim como as recomendações para trabalhos futuros.

Capítulo 2

2 Fundamentação teórica

Neste capítulo, são revisados alguns dos principais conceitos para a compreensão do trabalho. Inicialmente, há uma introdução aos conceitos relacionados ao reconhecimento facial, como parte dos mecanismos biométricos de autenticação e identificação. Além disso, a prova de vida é aprofundada como mecanismos *anti-spoofing* focados nesse ramo específico da biometria. Por fim, serão introduzidos os conceitos de aprendizado de máquina, bem como as principais arquiteturas e definições envolvidas em sistemas *anti-spoofing* na área de reconhecimento facial. Um resumo das características dos principais bancos de dados utilizados nesse contexto é abordado no final deste capítulo assim como as principais métricas de desempenho.

2.1 Biometria

A biometria oferece uma solução prática e poderosa para aplicativos que exigem autenticação. A palavra "biometria", do grego bios (vida) e métron (medida), significaria literalmente "medir vida". Em princípio, é o caminho natural para o homem reconhecer seus pares por características físicas e intransferíveis que os individualizam: sua voz, seus traços, seus movimentos (JAIN, ANIL K AND FLYNN, PATRICK AND ROSS, 2007). Com a evolução tecnológica, o uso de indicadores biométricos apresenta um grande desenvolvimento devido ao surgimento de componentes eletrônicos e digitais vinculados a softwares e processadores. Um conceito mais específico pode ser que a biometria é um sistema automatizado de reconhecimento humano baseado nas características físicas e no comportamento das pessoas (JAIN; ROSS; PANKANTI, 2006).

Existem dois tipos de medições de categorias biométricas, por comportamentos e medidas fisiológicas que podem ser morfológicas ou biológicas (BIOMETRICS INSTITUTE, 2019). Dentro dessas categorias, as medidas fisiológicas geralmente oferecem o benefício de permanecerem mais estáveis durante a vida de uma pessoa,

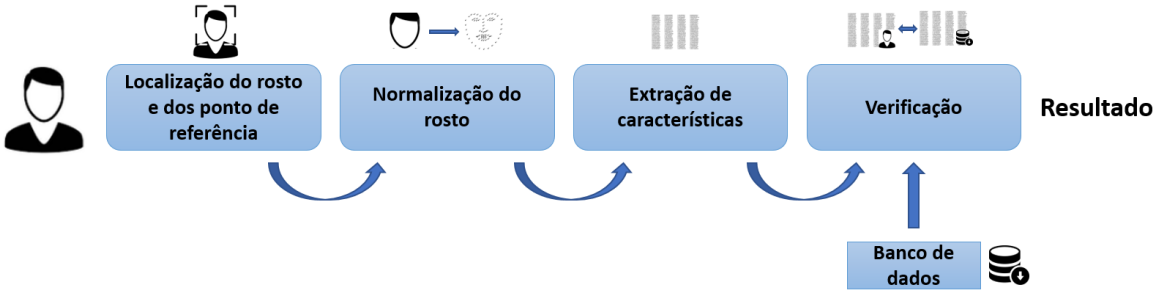
por não sofrerem os efeitos do estresse, em contraste com a identificação por medida do comportamento. Cada uma dessas categorias é verificada através de algoritmos de software e dispositivos de captura que realizam autenticação ou identificação em um determinado sistema. A autenticação biométrica responde à pergunta "Você é quem realmente afirma ser", onde é feita uma comparação com dados armazenados anteriormente dessa pessoa e, portanto, sua identidade é verificada. A identificação biométrica responde a "Quem você é", onde os dados biométricos da pessoa são capturados e comparados com o mesmo tipo de dados de outras pessoas em um banco de dados. As imagens faciais são os meios mais naturais usados pelos seres humanos para reconhecer as pessoas e não sendo necessário entrar em contato com o dispositivo de captura, leva a inúmeras vantagens de segurança, incluindo higiene.

2.2 Reconhecimento Facial

O reconhecimento facial é um método para identificar ou verificar a identidade de um indivíduo usando seu rosto. Além de natural e não intrusiva, a vantagem mais importante do rosto é que ele pode ser capturado remotamente e secretamente. Tornou-se cada vez mais importante devido aos rápidos avanços nos dispositivos de captura de imagens (câmeras de vigilância, câmeras em celulares), à disponibilidade de grandes quantidades de imagens faciais na Web e às crescentes demandas por maior segurança.

Na figura 2-1, quatro etapas são definidas no fluxo de trabalho de reconhecimento facial: detecção de rosto, normalização de rosto, extração de recursos e verificação, ((STAN Z; ANIL K, 2011).

Figura 2-1: Fluxo de trabalho de reconhecimento facial.



Fonte: Autor

A detecção de rosto neste fluxo de trabalho inclui a estimativa da pose do rosto, bem como a localização dos pontos de referência que estabelecem os pontos de referência entre olhos, nariz, boca, contorno facial (ZHU; RAMANAN, 2012). O objetivo é normalizar o rosto e reduzir informações inúteis, interferentes e redundantes, como fundo, cabelo, iluminação, variedade de poses etc. (QIAN; DENG; HU, 2019),(YI et al., 2015) A extração das características faciais após a normalização é o processo de extração das características dos componentes faciais, como olhos, nariz, boca, da imagem do rosto humano (SUFYANU et al., 2016). Na verificação, os recursos extraídos da face de entrada são comparados com uma ou muitas das faces cadastradas no banco de dados. O comparador gera "sim" ou "não" para verificação 1:1; para a identificação 1:N, a saída é a identidade da face de entrada quando a correspondência superior é encontrada com confiança suficiente ou é desconhecida quando a pontuação da correspondência está abaixo de um limite (SUN et al., 2014). O principal desafio nesta etapa do reconhecimento facial é encontrar uma métrica de similaridade adequada para comparar as características faciais.

Em situações restritas, por exemplo, onde iluminação, postura, distância, desgaste facial e expressão facial podem ser controlados, o reconhecimento facial automático pode superar o desempenho do reconhecimento humano (STAN Z; ANIL K, 2011). No entanto, o reconhecimento automático de rosto ainda enfrenta muitos desafios quando as imagens de rosto são adquiridas em ambientes irrestritos.

2.3 Prova de vida

Em biometria, prova de vida é a capacidade de um sistema de inteligência artificial de determinar se está interagindo com um ser humano fisicamente presente e não outro artefato (PAN; WU; SUN, 2008a). Também pode ser definido como a medição e análise de características anatômicas ou reações involuntárias ou voluntárias, para determinar se esta capturando uma amostra biométrica de um sujeito vivo (ISO/IEC 30107-1, 2016a). Segundo (DENNING, 2001) "O que torna a biometria bem-sucedida não é o segredo, mas sim a capacidade de determinar ‘vida’. A execução do ataque de apresentação varia de acordo com a modalidade biométrica; isto é, se a técnica biométrica usa impressões digitais, face, íris, voz ou biometria da digitação, etc.

A detecção de vida tem sido um tópico de pesquisa muito ativo nas comunidades de reconhecimento de impressões digitais e íris nos últimos anos. Mas no reconhecimento facial, as abordagens são muito limitadas para lidar com esse problema (CHAKRABORTY; DAS, 2014). O uso do reconhecimento facial estava aumentando e rapidamente esses sistemas se concentraram na detecção de ataques de suplantação sob reconhecimento facial. A detecção de vida impede que robôs e maus atores usem fotos roubadas, vídeos falsos, máscaras ou outras falsificações para criar ou acessar contas online. Os principais ataques detectados nos últimos anos são realizados por diferentes mídias, como imagens impressas ou tiradas de uma tela de monitor, vídeos ou máscaras 3D (RAMACHANDRA; BUSCH, 2017).

2.4 Instrumentos de ataque de apresentação

De acordo com ISO/IEC 30107-1 (ISO/IEC 30107-1, 2016), a característica ou objeto biométrico usado em um ataque de apresentação é chamado de *Presentation Attack Instrument* (PAI). Entre os diferentes tipos de PAI, os mais utilizados para violar os sistemas de reconhecimento facial são os classificados como instrumentos artificiais; impressões digitais de uma imagem, capturas de tela de imagens e vídeos ou máscaras 3D. Nas impressões digitais, o rosto do usuário é impresso em papel e apresentado na frente da câmera para verificação ou identificação. O ataque

fotográfico é a abordagem mais barata e mais fácil, já que a imagem facial de uma pessoa geralmente está facilmente disponível ao público, por exemplo, baixada da Web, capturada sem permissão ou sem conhecimento da existência de uma câmera (PAN; WU; SUN, 2008a). O impostor pode girar, deslocar e dobrar a foto na frente da câmera como uma pessoa viva para enganar o sistema de autenticação como apresentado na figura 2-2. Outra abordagem dentro das imagens é o uso daquelas capturadas de telas de telefone, computadores etc. Do mesmo modo que o método anterior é usado para tentar suplantar a identidade de outra pessoa, como mostrado na figura 2-3. A representação de vídeo é outra grande ameaça aos sistemas de reconhecimento facial. Ao contrário das imagens, é um mecanismo não estático contra roubo de identidade em sistemas que baseiam sua verificação movendo os olhos, lábios, etc. No ataque de reprodução de vídeo, um vídeo gravado de um indivíduo é mostrado na frente da câmera do sistema de autenticação para violá-lo.

Figura 2-2: Suplantação de identidade por meio de imagens.

Fonte: (PAN; WU; SUN, 2008b)

Figura 2-3: Suplantação de identidade utilizando foto na tela do celular.

Fonte: (ANN-KATHRIN SCHMITT, 2019)

Infelizmente, métodos que dependem da suposição de uma superfície plana para um rosto falso se tornam inúteis em caso de ataques de máscara facial 3D. O reconhecimento facial 3D é conhecido por atingir uma alta taxa de reconhecimento e por ser altamente seguro, especialmente contra ataques de roubo de identidade. Para o cenário de máscara facial 3D, o atacante usa uma máscara 3D para enganar o sistema de reconhecimento facial como apresentado na figura 2-4. Comparadas aos ataques de apresentação de faces 2D, as máscaras 3D contêm informações estruturais e de profundidade semelhantes às faces reais (LI et al., 2019).

Figura 2-4: Suplantação de identidade mediante máscara 3D.

Fonte: (BREWSTER, 2018)

Cada um desses instrumentos, sob certas condições, conseguiu e ainda consegue enganar os sistemas de segurança através do reconhecimento facial. Como resultado, algoritmos foram desenvolvidos com o objetivo de detectar e evitar essas tentativas de roubo de identidade.

2.5 Detecção de Ataque de Apresentação

Na literatura, a *Presentation Attack Detection* (PAD) é conhecida como uma contramedida ou uma técnica *anti-spoofing*. Pode ser definida como uma determinação automatizada de um ataque de apresentação (ISO/IEC 30107-1, 2016b). O termo PAD tem sido usado em muitas ocasiões como um mecanismo para a detecção de vida.

O uso de algoritmos de detecção de ataques de apresentação foi classificado em dois grupos, algoritmos baseados em hardware e software (RAMACHANDRA; BUSCH, 2017). Sob a abordagem de hardware, estão os mecanismos que usam adicionalmente um sensor de reconhecimento facial ou algum outro componente externo para executar a verificação. Essa abordagem também inclui sistemas baseados em desafios e respostas (mecanismos intrusivos) que solicitam algum tipo de ação a ser executada pelo usuário (ALI; DERAVI; HOQUE, 2013).

A abordagem baseada em software é possível através de algoritmos capazes de detectar a vida ou não, sem a necessidade de hardware especializado e sem a cooperação do usuário (mecanismos não intrusivos). Em (RAMACHANDRA; BUSCH, 2017), a abordagem baseada em software é dividida em duas partes, métodos dinâmicos e métodos estáticos. Os métodos dinâmicos trabalham na reprodução de vídeos e realizam a análise principalmente analisando movimentos, textura ou a fusão de ambos. A abordagem baseada em movimentos baseia sua funcionalidade em movimentos executados inconscientemente pela cabeça (MARSICO et al., 2012), os olhos (PAN et al., 2007), e a boca (KOLLREIDER et al., 2007). A análise de textura utiliza os algoritmos LBP de três planos ortogonais para determinar as alterações de textura durante a reprodução do vídeo (PEREIRA et al., 2012). E, finalmente, os sistemas híbridos baseados nos dois anteriores têm o objetivo de obter maior precisão na classificação final entre um vídeo real e um falso (YAN et al., 2012).

O método estático refere-se ao processamento de uma imagem sem informações adicionais, e seu uso pode ser estendido a vídeos com análise quadro a quadro. Na literatura são encontrados três tipos fundamentais de algoritmos; contemplando a análise de textura (CHINGOVSKA; ANJOS; MARCEL, 2012), frequência (LI et al., 2004) e sistemas híbridos entre os anteriores (KIM et al., 2012b). Descritores como LBP (OJALA; PIETIKAINEN; MAENPAA, 2002), *Local Phase Quantization* (LPQ) (OJANSIVU; HEIKKILÄ, 2008), *Co-Occurrence of Adjacent Local Binary Patterns* (CoALBP) (NOSAKA; OHKAWA; FUKUI, 2011) são alguns exemplos utilizados para a extração de características na análise de textura e frequência nas imagens. O processamento dessas características obtidas através dos descritores é

interpretado por algoritmos de *Machine Learning* que aprenderão com os dados obtidos para fazer uma previsão do que é real ou não. Na próxima seção, é realizado um estudo dos principais conceitos nessa área e sua aplicação nos sistemas PAD.

2.6 Aprendizado de Máquina

O aprendizado de máquina (*Machine Learning*) é um método de análise de dados que automatiza a construção de modelos analíticos (MITCHELL, 1999). É um ramo da Inteligência Artificial (IA) baseado na ideia de que os sistemas podem aprender com os dados, identificar padrões e tomar decisões com o mínimo de intervenção humana. Como afirmado anteriormente, é um campo da ciência da computação que, de acordo com Arthur Samuel (SAMUEL, 1959) dá aos computadores a capacidade de aprender sem serem explicitamente programados. Sua principal característica, no entanto, é não precisar ter as rotinas implantadas a mão: o próprio sistema tem a habilidade de aprender com a análise de dados e executar tarefas com uma precisão cada vez maior.

2.6.1 Algoritmos supervisionados

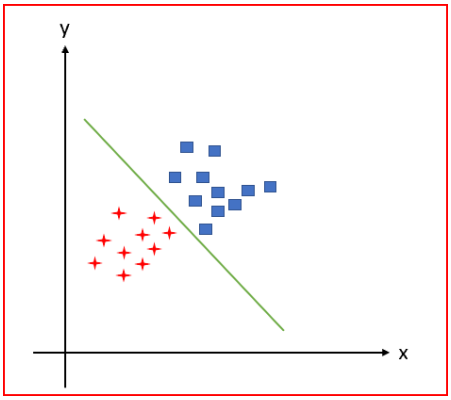
O tipo de aprendizado supervisionado tenta resolver os problemas de regressão e classificação e, para isso, utiliza algoritmos como o *Support Vector Machines* (SVM) e redes neurais. Estes são utilizados em termos de classificação tanto no reconhecimento facial quanto na prova de vida, e alcançaram resultados satisfatórios nos últimos anos (KIM et al., 2012)(KHAN et al., 2019)(MAULIK; CHAKRABORTY, 2017).

2.6.1.1 Support Vector Machine

No aprendizado de máquina, o SVM é um modelo de aprendizado supervisionado com algoritmos associados que analisam dados e reconhecem padrões, usado para análise da classificação e regressão (NOBLE, 2006). O SVM realiza uma separação de um conjunto de objetos com diferentes classes, ou seja, utiliza o conceito de planos de decisão como se apresenta na (Figura 2-5). Dado um conjunto de exemplos de treinamento, cada um rotulado como pertencente a uma das

duas categorias, um algoritmo de treinamento constrói um modelo que atribui novos exemplos em uma categoria ou outra. Em essência, um SVM é uma entidade matemática, um algoritmo (ou receita) para maximizar uma função matemática específica em relação a uma determinada coleção de dados (NOBLE, 2006).

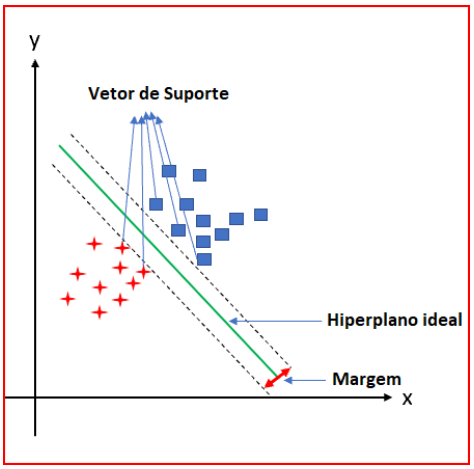
Figura 2-5: Plano de decisão no SVM.



Fonte: (KHANDELWAL, 2018)

Os vetores de suporte são os pontos de dados no conjunto de dados mais próximo do hiperplano como na Figura 2-6. A eliminação de vetores de suporte alterará o hiperplano que separa duas classes. Os vetores de suporte são elementos críticos do conjunto de dados, já que o SVM é baseado neles. A máquina de vetores de suporte tem dois objetivos principais: encontrar um hiperplano (linha) que separa linearmente os pontos de dados em duas classes; e maximizar a margem entre os vetores de suporte das duas classes (JAKKULA, 2006).

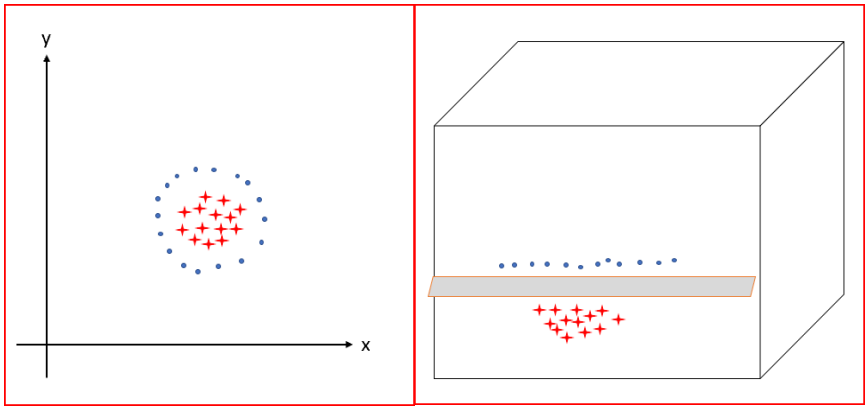
Figura 2-6: Representação do vetor de suporte.



Fonte: (KHANDELWAL, 2018)

O kernel SVM é uma função que utiliza um espaço de entrada de baixa dimensão e o transforma em um espaço de dimensão superior, ou seja, converte um problema não separável em um problema separável (JAKKULA, 2006). É especialmente útil em problemas de separação não linear (Figura 2-7-a). Simplificando, ele realiza algumas transformações de dados extremamente complexas e descobre o processo para separar os dados com base nas etiquetas ou saídas definidas (Figura 2-7-b).

Figura 2-7: Transformação de espaço não linear para espaço linear com função de Kernel



(a) (b)

a) Espaço dimensional não linear

b) Transformação de espaço não linear com função de Kernel

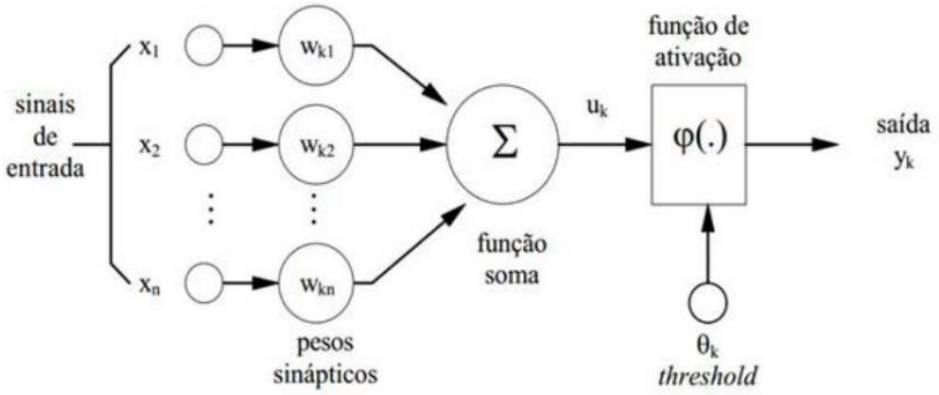
Fonte: (KHANDELWAL, 2018)

2.7 Redes Neurais

Uma Rede Neural Artificial (RNA) é um conjunto de algoritmos matemáticos que processam informações e encontram relações não lineares entre o conjunto de dados e cuja unidade básica de processamento é inspirada na célula fundamental do sistema nervoso humano: o neurônio (HOSKINS; HIMMELBLAU, 1988). A capacidade do cérebro humano de pensar, lembrar, relacionar fatos e resolver problemas inspirou muitos cientistas a tentar modelar seu funcionamento. Assim, as RNAs tentam imitar certas características do cérebro humano, como a capacidade de memorizar e associar fatos. É importante ter em mente que, todos os problemas que não podem ser expressos através de um algoritmo, o homem é capaz de resolvê-los recorrendo a algo chamado: experiência. Portanto, as redes neurais são modelos artificiais e simplificados do cérebro humano, capaz de adquirir conhecimento através da experiência (R.C. LACHER; NARAYAN; CYBENKO, 1999).

Como no caso do neurônio biológico, o neurônio artificial recebe estímulos que podem vir de fora ou da conexão com outros neurônios (HOSKINS; HIMMELBLAU, 1988) (Figura 2-8). As entradas recebidas pelo neurônio são modificadas por um vetor w de pesos sinápticos, cujo papel é simular a sinapse realizada pelos neurônios biológicos (ANDERSON, 1972). O parâmetro θ, é o *threshold* de um neurônio (UDYAVAR, 2017), os diferentes valores recebidos pelo neurônio modificado pelos pesos sinápticos, são adicionados para produzir o que é chamado de função soma, que determina se o neurônio é ativado ou permanece inativo. A ativação ou não, depende do que é chamado de função de ativação (KARLIK; OLGAC, 2011), a saída y_k do neurônio é gerada pela avaliação da função soma na função de ativação e isso pode ser propagado para outros neurônios ou resultar na saída final da rede.

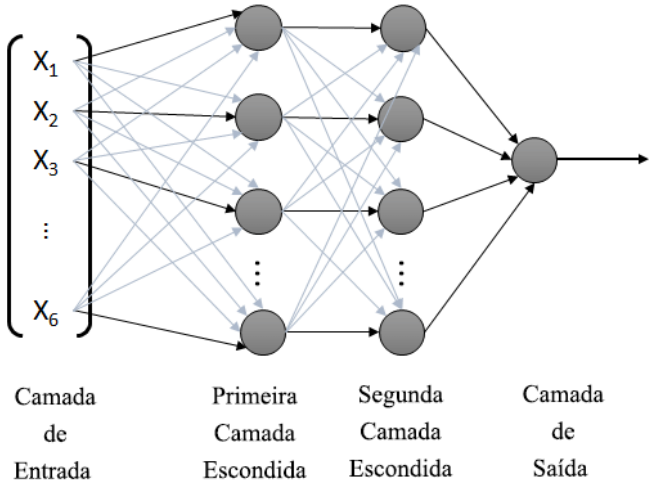
Figura 2-8: Processo de ativação dos neurônios.



Fonte: (HAYKIN, 2001)

O neurônio artificial sozinho possui baixa capacidade de processamento e seu nível de aplicabilidade é baixo. O verdadeiro potencial dos neurônios está na interconexão entre eles, como é o caso dos neurônios biológicos no cérebro humano. Dadas essas premissas, diferentes pesquisadores propuseram uma variedade de estruturas de conectividade de neurônios artificiais, dando origem ao que é conhecido como redes neurais artificiais (JAIN; MAO; MOHIUDDIN, 1996a). A distribuição dos neurônios dentro de uma rede neural artificial é feita através da formação de níveis ou camadas como se apresenta na (Figura 2-9). A camada de entrada é a camada que recebe as variáveis de entrada na rede, ou seja, onde as informações externas são processadas. As camadas ocultas são as camadas intermediárias da rede que não têm contato com o exterior e sua conexão variada e quantidade determinarão o tipo de rede (JAIN; MAO; MOHIUDDIN, 1996b). A rede em quanto ao número de camadas pode ser definida como redes monocamada ou multicamada (JAIN; MAO; MOHIUDDIN, 1996b). A camada de saída é a responsável por transferir as informações processadas nas camadas anteriores para o exterior.

Figura 2-9: Representação das camadas de distribuição das redes neurais artificiais.



Fonte: (VIDAL et al., 2015)

A partir do conceito de redes neurais e produto de avanços computacionais e disponibilidade de grandes volumes de dados, surge *Deep Learning*. Esse novo subconjunto do *Machine Learning* representa a vanguarda da inteligência artificial e tornou-se o passo mais próximo em direção à inteligência artificial real. O uso expandiu-se para áreas diferentes, sendo que o reconhecimento facial e a prova de vida foram beneficiados com o seu surgimento.

2.8 Aprendizado Profundo

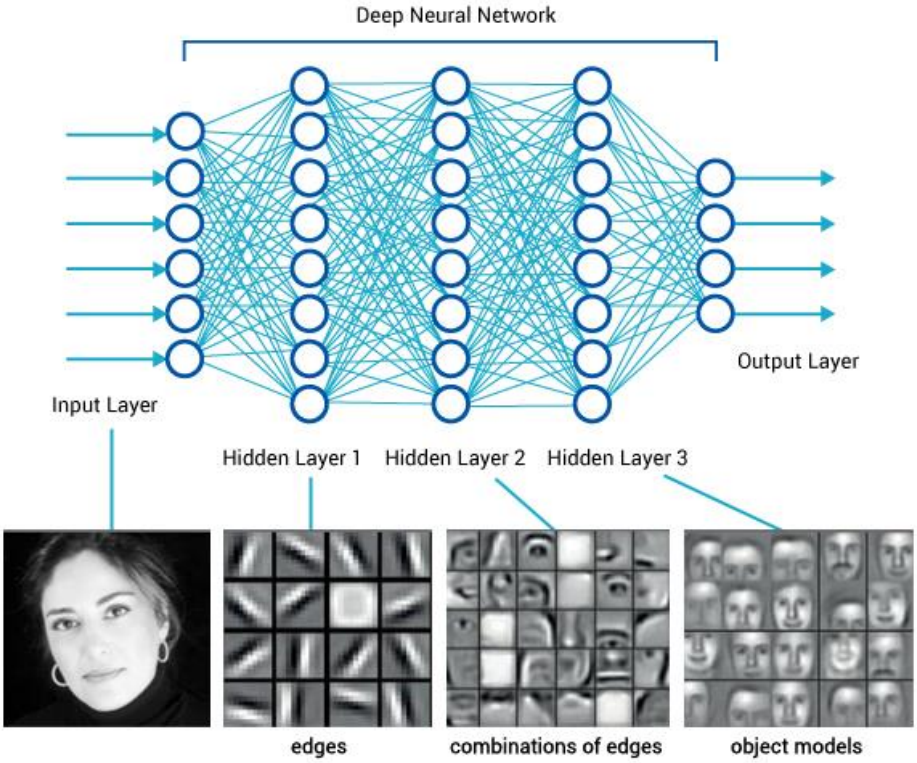
O Aprendizado Profundo (*Deep Learning*) basicamente é um conjunto de algoritmos de aprendizado de máquina baseados em técnicas de redes neurais artificiais que extraem e processam em cada camada informações úteis para uma próxima camada (YANN LECUN, 2015). Dessa maneira, esses tipos de algoritmos são capazes de obter um aprendizado completo das informações de entrada e podem tirar conclusões relevantes nos seus resultados. Os modelos computacionais do *Deep Learning* imitam os recursos arquitetônicos do sistema nervoso, permitindo que dentro do sistema global existam redes de unidades de processo especializadas na detecção de certos recursos ocultos nos dados (YANN LECUN, 2015).

O número de camadas de processamento pelas quais os dados devem passar é o que inspirou a etiqueta de profundidade *"deep"*. O surgimento desse conceito não seria possível sem o nascimento das *Graphics Processing Unit* (GPUs), devido à natureza altamente paralelizável desses problemas, o uso de GPUs permite um aumento no desempenho em várias ordens de magnitude (ZHANG et al., 2017). Para alcançar um nível aceitável de precisão, os programas de *Deep Learning* exigem acesso a imensas quantidades de dados de treinamento que só foram possíveis na era do *Big Data* (MCAFEE et al., 2012). Conforme a rede recebe informações, ela vai tomar decisões corretas em uma porcentagem maior de ocorrências e, portanto, o algoritmo evolui devido à alimentação de milhões de exemplos, alcançando uma melhoria ao longo do tempo.

O *Deep Learning* realiza o processo de *Machine Learning* usando uma rede neural artificial que consiste em vários níveis hierárquicos. No nível inicial da hierarquia, a rede aprende algo simples e envia essas informações para o próximo nível. O próximo nível pega essas informações simples, as combina, compõe informações um pouco mais complexas e as passa para o próximo nível, e assim por diante.

O nível inicial de uma rede de *Deep Learning* pode usar as diferenças entre as áreas claras e escuras de uma imagem de um rosto para saber onde estão as bordas da imagem. O nível inicial passa essas informações para o segundo nível, que combina as arestas criando formas simples, como uma linha diagonal ou um ângulo reto. O terceiro nível combina formas simples e obtém objetos mais complexos, como ovais ou retângulos. O próximo nível poderia combinar os ovais e retângulos, formando olhos, nariz ou boca. O processo continua até que seja alcançado o nível superior na hierarquia, onde finalmente a rede aprende a identificar a pessoa. Um exemplo simples de uma rede neural profunda de três camadas ocultas é mostrado na Figura 2-11.

Figura 2-10: Rede neural profunda de 3 camadas.

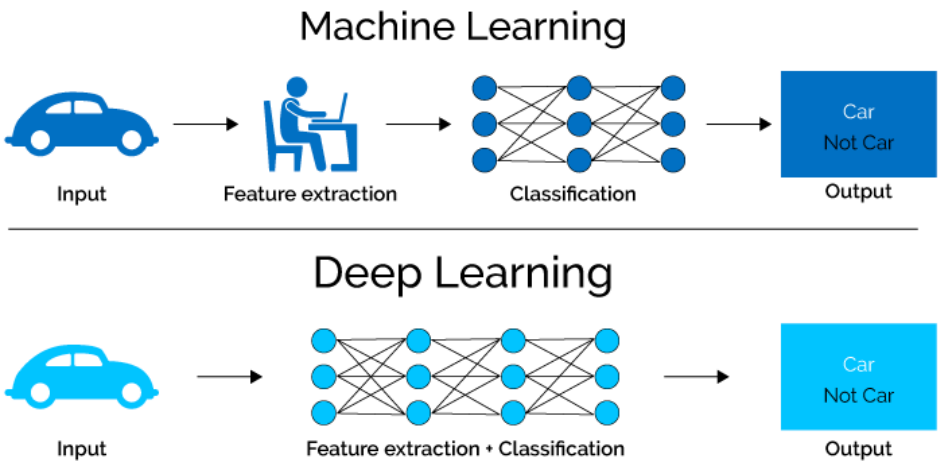


Fonte: (COLLET, 2017)

Embora os termos às vezes sejam usados como sinônimos, *Deep Learning* e *Machine Learning* não são os mesmos, o primeiro sendo um tipo específico do segundo, ou seja, *Deep Learning* é *Machine Learning*, mas existem técnicas de *Machine Learning* que não são *Deep Learning*.

Um fluxo de trabalho de aprendizado de máquina começa com os recursos relevantes extraídos manualmente das imagens. Os recursos são usados para criar um modelo que classifique os objetos na imagem. Com um fluxo de trabalho de aprendizado profundo, os recursos relevantes são extraídos automaticamente das imagens como na Figura 2-12.

Figura 2-11: *Machine Learning vs Deep Learning*



Fonte: (GILL, 2018)

2.8.1 Arquiteturas de Aprendizado Profundo

Existem muitas arquiteturas de aprendizado profundo voltadas para trabalhos com processamento e classificação de imagens. Tudo isso é baseado na RNA, mas possui otimizações específicas que as tornam boas para determinados casos de uso. A seguir, é feita uma breve abordagem para alguns dos mais utilizados nesta área.

Convolutional Neural Network **(CNN):** as redes neurais convolucionais são a escolha popular de redes neurais para diferentes tarefas de visão computacional, como reconhecimento de imagem (RAWAT; WANG, 2017). O nome "convolução" é derivado de uma operação matemática que envolve a convolução de diferentes funções. Seu principal uso é no reconhecimento visual onde se obtém bons resultados. Sua principal desvantagem está no tamanho e na qualidade dos dados inseridos nele para treinamento.

Residual Neural Network **(ResNet):** são muito úteis para solucionar problemas de classificação de imagem e reconhecimento visual (WU; SHEN; VAN DEN HENGEL, 2019). À medida que as tarefas se tornam mais complexas, o treinamento da rede neural começa a ser muito mais difícil, pois são necessárias

camadas profundas adicionais para calcular e melhorar a precisão do modelo. Sua principal desvantagem está na profundidade em que os erros podem ser difíceis de detectar e não podem se propagar rápida e corretamente. Ao mesmo tempo, se as camadas forem menos profundas, o aprendizado pode não ser muito eficiente.

***Inception Neural Network* (INN):** Os módulos de inicialização são usados em redes neurais convolucionais para permitir cálculos mais eficientes e redes mais profundas através da dimensionalidade reduzida (SZEGEDY et al., 2016). Os módulos foram projetados para resolver o problema de gastos computacionais, além de super ajuste, entre outros problemas. A solução, em resumo, é usar vários tamanhos de filtro de núcleo na CNN e, em vez de empilhá-los sequencialmente, ordenando que operem no mesmo nível. Isso permite que o modelo aproveite a extração de recursos de vários níveis. Por exemplo, extrai características gerais (5x5) e locais (1x1) ao mesmo tempo.

***Siamese Neural Network* (SNN):** Recentemente, as arquiteturas de redes neurais siamesas foram desenvolvidas para aprender o conceito de similaridade e dissimilaridade entre imagens de entrada e foram usadas de maneira eficiente para o reconhecimento de faces e mapeamentos de regiões semelhantes de imagens (MELEKHOV; KANNALA; RAHTU, 2016). O objetivo da rede é obter o descritor de duas imagens, já que o objetivo é o mesmo, não é aconselhável usar duas CNNs diferentes que executam o mesmo trabalho. Portanto, surge a ideia de executar uma única CNN para as duas imagens de entrada que executam essa tarefa que compartilha os mesmos pesos, parâmetros que são usados pelas duas imagens (GUO et al., 2017). Permite encontrar a distância entre duas imagens, ou seja, se duas imagens corresponderem à mesma pessoa, obteremos um pequeno valor de distância, caso contrário, esse valor será grande.

2.9 Bancos de dados

A avaliação dos sistemas PAD é realizada a partir dos treinamentos e testes realizados em diferentes bancos de dados. Portanto, a disponibilidade pública deles e a tecnologia utilizada para a apresentação e aquisição de faces são um fator decisivo

na implementação de novos métodos. A seguir, são apresentados alguns dos bancos de dados mais relevantes com base no número de dados, variedade de ataques, dispositivos de captura etc.

NUAA

É o primeiro banco de dados público disponível para ataques de apresentação. Tem 15 pessoas diferentes, com vídeos em 20fps tirados com uma webcam em 3 seções diferentes com variação de brilho e ambiente. Cada seção contém 500 amostras de cada pessoa e as capturas são feitas frontalmente e sem expressões faciais. O *Presentation Attack Instrument* (PAI) utilizado foram as fotos impressas das imagens capturadas. O banco de dado é composto por 5105 imagens reais e 7509 imagens falsas (TAN et al., 2010).

Replay Video Attack Database

No banco de dados são utilizadas 50 pessoas para capturar amostras de vídeos reais e falsos. Os vídeos são gravados em dois cenários diferentes e em duas sequências de vídeo com resolução 320 x 240 (QVGA) a 25 fps por 15 segundos. São tiradas duas fotos de cada um dos participantes com câmeras de 12.1 e 3.1 megapixels, respectivamente, para representar as tentativas de ataque. Além das impressões, são capturados vídeos em vários dispositivos como ataques de falsificação. São usados neste banco de dados um total de 200 vídeos reais e 1000 vídeos falsos (CHINGOVSKA; ANJOS; MARCEL, 2012).

CASIA FASD

No banco de dados são usadas três câmeras diferentes para captura de baixa, média e alta qualidade nas resoluções 640 x 480, 480 x 640 e 1920 x 1080 respetivamente. As amostras são coletadas de 50 participantes em condições naturais. As imagens de alta qualidade são impressas como amostras falsas e os vídeos são reproduzidos na tela de um iPad que, devido à sua resolução de 1280 x

720, reduz a qualidade dos vídeos de alta resolução. São capturados um total de 150 vídeos reais e 450 falsos (ZHANG et al., 2012).

MSU-USSA

O banco de dados tem um total de 1140 pessoas. Do total, 1000 amostras são extraídas da Web, com uma variedade de condições de iluminação e ambientes. As outras 140 são distribuídos entre outros bancos de dados com uma resolução média total de 705 x 865. Totalizando, o banco de dados contém 6840 imagens capturadas de diferentes câmeras com resoluções 1280 × 960 e 3264 × 2448 como PAI, sendo projetadas na tela de diferentes dispositivos. Além disso, é criada uma quantidade de 2280 imagens impressas para serem usada também como PAI. São contemplados diferentes expressões faciais e ângulos de posição (PATEL; HAN; JAIN, 2016).

Oulu-NPU

No banco de dados os vídeos são capturados em duas sessões com uma diferença de duas semanas e em um ambiente homogêneo com diferentes alterações na luminância. Foram analisadas 40 pessoas diferentes e capturado um total de 1980 vídeos reais e 3960 vídeos falsos. Os PAIs de impressão e reprodução de vídeo são usados em telas diferentes (BOULKENAFET et al., 2017).

SiW

O banco de dados contém 165 pessoas com vídeos gravados a 30fps com 8 vídeos reais e 20 vídeos falsos, totalizando 4620 vídeos. Os vídeos são gravados com câmeras de resolução 1920 x 1080 e são feitas 2 impressões e 4 reproduções de vídeo como métodos de ataques de apresentação. Várias resoluções de impressão são implementadas em baixa e alta qualidade 5184 × 3456 e consideram expressões faciais e ângulos não frontais da face entre -90º e 90º ((LIU; JOURABLOO; LIU, 2018).

SiW - M

O banco de dados inclui novos tipos de ataques que não foram considerados anteriormente. Nesse caso, foram coletados 968 vídeos de 13 tipos de ataques de falsificação. Os vídeos são gravados em 1080 e 720 pixels. Foram utilizados 660 vídeos de 493 pessoas como vídeos reais em 3 seções e são contemplados diferentes ângulos de posicionamento da face, luminosidade e expressões. Vídeos e impressões falsas são utilizadas como PAIs (LIU et al., 2019).

2.10 Métricas de desempenho

Para avaliar o desempenho dos sistemas PAD são usadas uma série de métricas de classificação binária. Nos trabalhos a seguir são utilizadas para validar os sistemas propostos. A seleção adequada das métricas de avaliação é uma chave importante para discriminar e obter o classificador ideal.

Os sistemas de classificação binária têm dois tipos de erros, um é *False Positives* (FP), que geralmente é relatado como *False Positive Rate* (FPR) e o outro é *False Negatives* (FN) ou *False Negative Rate* (FNR). FPR corresponde à razão FP e o número total de amostras negativas e a FNR corresponde à relação entre a FN e o número total de amostras positivas (CHINGOVSKA et al., 2019). Como os aspectos positivos e negativos estão associados à ação de aceitação e rejeição pelo sistema de verificação, uma prática comum é substituir o FPR e o FNR pela *False Acceptance Rate* (FAR) e pela *False Rejection Rate* (FRR) respectivamente (CHINGOVSKA; ANJOS; MARCEL, 2014).

Half Total Error Rate (HTER) é o erro médio calculado a partir do FPR e FNR. Após a padronização da ISO (ISO/IEC JTC 1/SC 37, 2016), o HTER foi chamado de *Average Classification Error Rate* (ACER), o FPR passou a ser chamado de *Attack Presentation Classification Error Rate* (APCER), que indica a proporção de ataques de apresentação classificados incorretamente como ataques de boa-fé e FNR como *Bona Fide Presentation Classification Error Rate* (BPCER) indica a proporção de ataques de boa-fé classificados incorretamente como ataques de apresentação. Em

Chingovska (CHINGOVSKA et al., 2019) é mostrada uma tabela com alguns dos termos usados para avaliar a taxa de erro e alguns dos seus sinônimos que podem ser observados na literatura. O valor mais baixo da taxa de erro significa melhor desempenho do sistema.

São analisadas também mais duas métricas nesses trabalhos, uma delas é *Equal Error Rate* (EER) ou *Crossover Error Rate* (CER). São usadas para predeterminar os valores limite para FAR e FRR. Quando as taxas FAR e FRR são iguais é chamado de EER, quanto menor esse valor, maior a precisão do sistema biométrico (CHINGOVSKA et al., 2019). *Area Under Curve* (AUC) é uma medida de desempenho para o problema de classificação em várias configurações de limite. *Receiver Operating Characteristic* (ROC) é uma curva de probabilidade e AUC representa o grau ou a medida da separabilidade. Quanto maior é AUC, melhor é o modelo para prever 0 como 0 e 1 como 1. Por analogia, quanto maior a AUC, melhor será o modelo (CHINGOVSKA et al., 2019)

Capítulo 3

3 Trabalhos relacionados

Neste capítulo, é realizado um estudo dos principais trabalhos da última década relacionados aos sistemas PAD. São contemplados os principais trabalhos baseados em software, com resultados relevantes e contribuições inovadoras nessa área. Também são apresentadas as pesquisas relacionadas à análise de textura em diferentes espaços de cores, ao uso de diferentes descritores na extração de características, aos mecanismos de *Machine Learning* e *Deep Learning* na extração de características e classificação, bem como a mistura deles para obter melhores resultados.

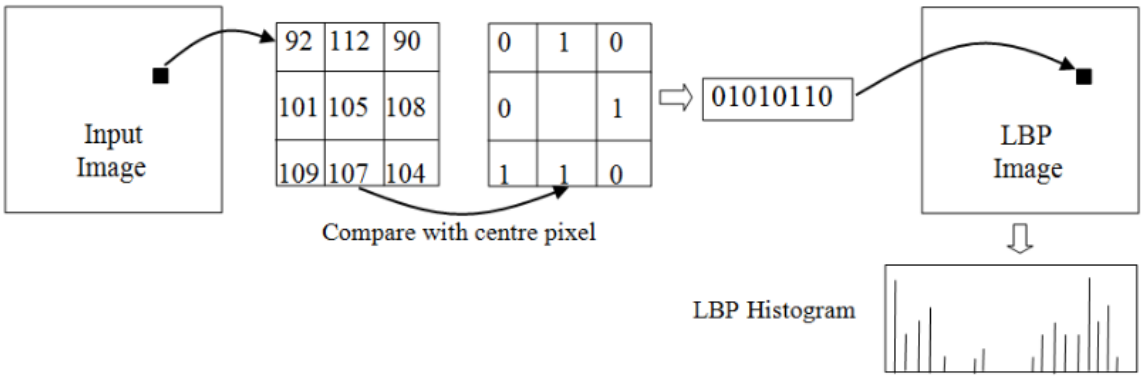
3.1 Análise baseada em textura

As imagens impressas ou tomadas da tela do computador, assim como as máscaras, são normalmente bem identificadas pelo olho humano, mas para sistemas que precisam detectar essas alterações sem a intervenção humana, fica realmente difícil encontrar essas diferenças. Mediante diferentes características em termos de desfoque, efeitos de iluminação, pigmentação, frequência que podem ser observadas em uma imagem, muitos trabalhos se concentram na análise de textura com o objetivo de extrair características distintivas entre imagens reais e falsas.

Uma das principais pesquisas nesse campo foi realizada por Li et al., (LI et al., 2004) que realizou um método baseado nos espectros de Fourier para determinar a variação dos componentes de frequência entre as imagens. Em (BAI et al., 2010), é proposto um método baseado na análise de uma única imagem e na obtenção da refletância a partir de sua microtextura. Tan et al. (TAN et al., 2010) consegue extrair as características da superfície de imagens de rostos vivos e fotografias, usando o modelo de refletância lambertiana e o filtro de diferença gaussiana (DoG) em termos de amostras latentes.

(MÄÄTTÄ; HADID; PIETIKÄINEN, 2011) realiza uma nova proposta com base na análise de textura para detecção de vida. Essa proposta surge da ideia de que as impressões de uma imagem normalmente apresentam defeitos, como baixa qualidade, reflexão da luz e sombras, características que diferem entre imagens impressas e rostos humanos reais. No trabalho, eles fazem uma proposta para a extração de características para obter os padrões de microtextura usando múltiplas escalas - LBP. Esse descritor codifica a relação de cada pixel na imagem com a intensidade da cor dos pixels vizinhos, onde cada pixel com um valor maior que o analisado recebe o valor 1, e no outro caso o valor 0. A partir desse resultado, é obtido um valor binário representativo para cada um dos pixels, o que criará um histograma que será o descritor de textura dessa imagem (AHONEN; HADID; PIETIKAINEN, 2006), como apresentado na (Figura 3-1).

Figura 3-1: Descritor LBP



Fonte: (MUKUNDAN, 2014)

Na arquitetura proposta, (MÄÄTTÄ; HADID; PIETIKÄINEN, 2011) realizam uma transformação da imagem em escalas de cinza e detecta o rosto na imagem para normalizá-la com um corte de 64 x 64 pixels. São usadas três escalas do descritor LBP para obter os padrões de microtextura. Cada uma dessas escalas retorna seu próprio histograma, que é concatenado para obter o histograma final que descreve a textura da imagem. Para classificação, eles usam um SVM não linear que determinará

se a imagem de entrada é real ou falsa. Com essa arquitetura, o trabalho alcança um EER de 2,9% para o descritor LBP treinado e testado no banco de dados NUAA.

(CHINGOVSKA; ANJOS; MARCEL, 2012) propõem uma análise semelhante à proposta anterior, realizando também uma análise de textura com o uso de LBP. Uma das suas principais contribuições foi a criação de um banco de dados para treinamento e teste chamado *Replay - Attack*. Neste trabalho, diferente do anterior, eles não usam várias escalas no uso do descritor LBP. Na arquitetura proposta, também é realizada a transformação em escalas de cinza e a imagem também é normalizada para 64 x 64 pixels. A proposta é dividida em dois caminhos: no primeiro, a extração das características de toda a imagem é realizada com o uso de LBP; no segundo, a imagem é dividida em blocos de 3 X 3; essa última proposta é baseada na ideia de (MÄÄTTÄ; HADID; PIETIKÄINEN, 2011) que mostra como as características da textura podem ser mais visíveis em áreas menores da imagem. Todos os vetores de características obtidos dessas imagens são representados em histogramas, é realizada uma concatenação deles para obter a representação final da imagem. Foram utilizados para a classificação: qui-quadrado (X2) (SATORRA; BENTLER, 2001), *Linear Discriminant Analysis* (LDA) (BALAKRISHNAMA; GANAPATHIRAJU, 1998) e SVM, por fim, é realizada para cada classificador uma análise comparativa dos resultados. Os resultados de cada um são apresentados com três bancos de dados: *Replay – Attack* (a proposta do trabalho), NUAA e CASIA - FASD (ZHANG et al., 2012). De acordo com a arquitetura e implementação dos multicanais LBP com o classificador SVM, (CHINGOVSKA; ANJOS; MARCEL, 2012) atingem um valor de 4,23% do HTER com NUAA.

3.2 Análise de textura pela cor

A baixa resolução de uma imagem ou a falta de parâmetros para realizar a análise de textura, dificultavam achar diferenças significativas na classificação entre uma imagem real e uma imagem falsa. A partir disso, os pesquisadores começaram a se adentrar no tópico da análise de textura, mas desta vez analisando as propriedades cromáticas que poderiam ser extraídas de uma imagem.

(BOULKENAFET; KOMULAINEN; HADID, 2015b), propõem a análise dos espaços de cores para distinguir entre uma face real e uma face falsa. A partir da sua análise de que a luminosidade é mais perceptível que o croma diante do olho humano e que por isso os rostos falsos e reais vão ter muitas similaridades quando as imagens são mostradas em cores, eles propõem a análise de certos componentes cromáticos para determinar algumas diferenças. São analisados três espaços de cores RGB, HSV e YCbCr, e como nos trabalhos mencionados anteriormente, o descritor LBP é usado para extrair as informações de textura de cores. Apesar do RGB ser o espaço de cores mais utilizado para representar imagens, o foco da pesquisa está nos outros dois espaços capazes de ter seus canais de luminância e croma bem definidos. No HSV, o (H) representa o matiz, a (S) a saturação, enquanto o valor (V) corresponde à luminância. Em YCbCr o espaço RGB é separado em luminância (Y), crominância azul (Cb) e crominância vermelha (Cr) (LUKAC; PLATANIOTIS, 2007).

Na arquitetura proposta por (BOULKENAFET; KOMULAINEN; HADID, 2015b), várias imagens são capturadas de vídeos com intervalos de 3 e 4 segundos. Cada imagem RGB é normalizada para 64 x 64 pixels e transformada nos espaços de cores mencionados acima. Cada espaço de cor é dividido em cada um de seus canais de luminância e cromaticidade e é analisado individualmente pelo descritor LBP, a fim de obter um histograma de cada canal. Como nos trabalhos já vistos neste capítulo, esses histogramas são concatenados para obter um histograma geral que representará essa imagem, também neste caso, é usado um SVM linear para a classificação. Na sua proposta, eles conseguem obter 0,4% EER e 2,9% HTER usando o banco de dados *Replay* – Attack e 6,2% EER para o banco de dados CASIA.

Boulkenafet em (BOULKENAFET; KOMULAINEN; HADID, 2016) faz uma extensão do seu trabalho explicado anteriormente sob a mesma filosofia de análise de espaços de cores. Nesse caso, além do descritor LBP, usa CoALBP apresentados por (NOSAKA; OHKAWA; FUKUI, 2011), LPQ apresentado por (OJANSIVU; HEIKKILÄ, 2008), este último especificamente pela suas inúmeras vantagens diante das imagens desfocadas, *Binarized Statistical Image Features* (BSIF) propostos por (KANNALA; RAHTU, 2012) e *Scale-Invariant Descriptor* (SID) com base na transformação de

Fourier e apresentados por (KOKKINOS; YUILLE, 2008). Desta vez é adicionado o banco de dados MSU MFSD para fazer os testes. Com a proposta e com a introdução de novos descritores no seu trabalho, ele consegue obter 0,4% EER e 2,8% HTER usando o banco de dados *Replay - Attack*, 2,1% EER para CASIA e 4,9% EER para MSU MFSD.

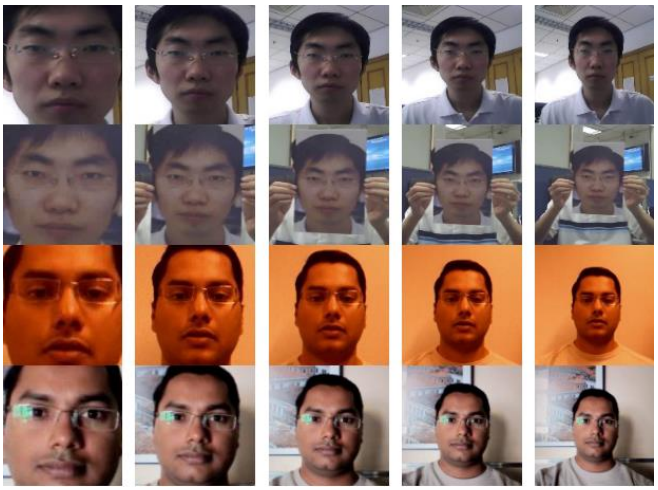
3.3 Aprendizado profundo na extração e classificação

(YANG et al., 2013) definem alguns aspectos que podem produzir erros na extração de características pela textura: imagens desfocadas como resultado da resolução do objeto de captura, resolução das imagens, telas impressas e mudanças na refletância ou nas sombras.

O uso de algoritmos de *Deep Learning* e a capacidade dos GPUs de realizar o seu processamento permitiram extrair outras características das imagens e ficar menos vulnerável aos aspectos mencionados. Uma das principais arquiteturas usadas na área de *Computer Vision* e que mais tarde se encontrou sua aplicação em sistemas *anti-spoofing* é a CNN ((KRIZHEVSKY; SUTSKEVER; HINTON, 2012; LAWRENCE et al., 1997).

De acordo com (YANG; LEI; LI, 2014), o seu trabalho é a primeira tentativa nesse tipo de análise, com base na extração de características usando a CNN. Na primeira parte do seu experimento, divide em cinco escalas de imagem para obter características que possam ser úteis no plano de fundo da imagem (Figura 3-2). Essas características (limites de uma fotografia impressa ou bordas do monitor, reflexos ou bordas desfocadas) podem ser extraídas pela CNN e diferenciais ao analisar imagens reais e falsas. A rede proposta consiste em cinco camadas convolucionais, seguidas por três camadas totalmente conectadas e é aplicada uma função de ativação *Rectified Linear Unit* (ReLU) em cada saída da camada convolucional e nas camadas totalmente conectadas.

Figura 3-2: Imagens em diferentes escalas de cores



Fonte: (YANG; LEI; LI, 2014)

Na proposta, as imagens são normalizadas para 128 x 128 pixels para cada uma das escalas utilizadas. Depois que os recursos são extraídos pela CNN, usa SVM para classificar o sistema. São utilizados dois bancos de dados para o conjunto de treinamento e teste, os bancos *Replay Attack* e CASIA, alcançando neste último um EER de 4,64% e um HTER menor de 5%.

Li em (LI et al., 2016) ajustou a CNN nos conjuntos de dados de representação facial e, em seguida, extraiu recursos e aplicou a *Principal Component Analysis* (PCA) para reduzir a dimensionalidade e reduzir os problemas de *over-fitting* do modelo. No seu trabalho, eles propõem extrair as características das partes mais profundas da CNN (DPCNN por sua sigla em inglês), alcançando uma rede três vezes mais profunda do que a proposta do (YANG; LEI; LI, 2014). Li usa um modelo VGG-face pré-treinado que foi projetado por (PARKHI, 2015) para reconhecimento facial. Finalmente, o SVM é usado para fazer a classificação entre a face real versus a falsa. São usados dois bancos de dados para o treinamento e teste, *Replay Attack* com um EER 2,9% e um HTER de 6,1% e CASIA com um EER de 4,5%.

Y. Atoum et al. (ATOUM et al., 2017), propõem um novo método para a detecção de características e, posteriormente, a sua classificação. Seu método é orientado de duas maneiras: análise de amostras aleatórias em diferentes partes da face e análise de mapas de profundidade que cobrem toda a região da face, a fim de obter, a partir de imagens bidimensionais, a distribuição espacial real. Uma das suas principais motivações para o uso de recortes é aumentar o número de amostras para treinamento e reduzir o tamanho das imagens, considerando que pode ser um fator importante, pois a CNN não vai precisar reduzir o tamanho da imagem e fazer alterações na resolução, o que ajudaria a obter mais dados discriminatórios. Esses dados são extraídos com um descritor LBP nos espaços de cores HSV e YCbCr para obter os diferentes canais de luminância e crominância. Os mapas de profundidade detectam a presença de uma face real (3D) ou uma impressão plana e são baseados nos algoritmos implementados por (JOURABLOO; LIU, 2015, 2016, 2017).

Na abordagem de Y. Atoum et al. (ATOUM et al., 2017), tanto para o método de aplicação de recortes quanto para o de profundidade da imagem, usam a CNN. A rede neural é composta por cinco camadas convolucionais e três totalmente conectadas, com uma função ReLU de ativação. A fusão dos valores finais desses dois métodos resultará na classificação da imagem. Vale ressaltar que, no caso da análise de profundidade após a extração das características da imagem, ela é classificada com um SVM. São usados para avaliar o sistema três bancos de dados: *Replay Attack* atingindo 0,79% EER e 0,72% HTER; MSU-USSA atingindo 0,35% EER e 0,21% HTER; para CASIA-MFSD atingindo 2,67% EER e 2,27% HTER.

(LIU; JOURABLOO; LIU, 2018) propõe um modelo *Deep Learning* baseado em informações espaciais e temporais focado em ataques de apresentação de vídeo. O método espacial está ligado à análise da profundidade de uma imagem, aumentando a diferença entre os pontos mais próximos de uma câmera e seu relacionamento com os outros em uma face real, uma diferença que não é percebida nas imagens impressas ou nos ataques de vídeo, pois os pixels nesses casos mantêm a profundidade. Do ponto de vista espacial, a *Remote Photoplethysmography* (rPPG) é usada para a detecção de sinais vitais, como um pulso cardíaco. Vários trabalhos já

demonstram que esses sinais são detectáveis em vídeos reais e não em vídeos de falsificação. Haan em (DE HAAN; JEANNE, 2013) consegue obter dados rPPG de vídeos RGB com alterações de movimento e brilho. Este artigo também apresenta o banco de dados SiW.

A arquitetura proposta por (LIU; JOURABLOO; LIU, 2018) possui duas redes profundas: uma CNN que avalia cada quadro retirado do vídeo e calcula a profundidade e o mapa de características; e uma RNN para estimar sinais rPPG com supervisão sequencial. As métricas APCER, BPCER, ACER, HTER foram usadas para avaliar a proposta e foram utilizados os bancos de dados CASIA-MFSD, *Replay Attack*, SiW e Oulu. É importante destacar um ACER de 3,58% no banco de dados proposto (SiW).

Uma nova proposta foi feita por Garg et al. (GARG et al., 2020) na qual os autores propuseram uma *Deep Belief Network* - DBN (HUA; GUO; ZHAO, 2015). O DeBNet proposto é usado para extração de características e classificação de rosto em real ou falso. O conjunto de dados de imagens utilizado para treinamento e teste foi NUAA. Nessa abordagem é feito primeiramente um pré-treino não supervisionado e em seguida, com o objetivo de fazer ajustes de pesos, um treinamento supervisionado para rotular os parâmetros na última camada. Foram usadas 3 técnicas para extração de caraterísticas LBPSVM, LBPDBN e DeBNet, e no caso desse último, como é baseado em aprendizagem profunda, participa tanto da extração de características como na classificação. No experimento os autores conseguem um valor de 0,31 HTER, um valor muito baixo em comparação com as outras duas técnicas utilizadas.

A proposta de Mohamed et al. (MOHAMED et al., 2021) também se foca na melhora dos resultados de classificação. Nesse caso uma rede neural convolucional é utilizada para extração de caraterísticas e classificação das imagens em real ou falsas. Uma novidade foi o treinamento com um novo banco de dados CelebA Spoof (ZHANG et al., 2020).Um diferencial interessante na proposta foi o aumento de dados para melhorar o treinamento mediante a inversão horizontal, zoom e cisalhamento sendo usadas imagens com tamanho de 150x150. Um detalhe a ressaltar foi a

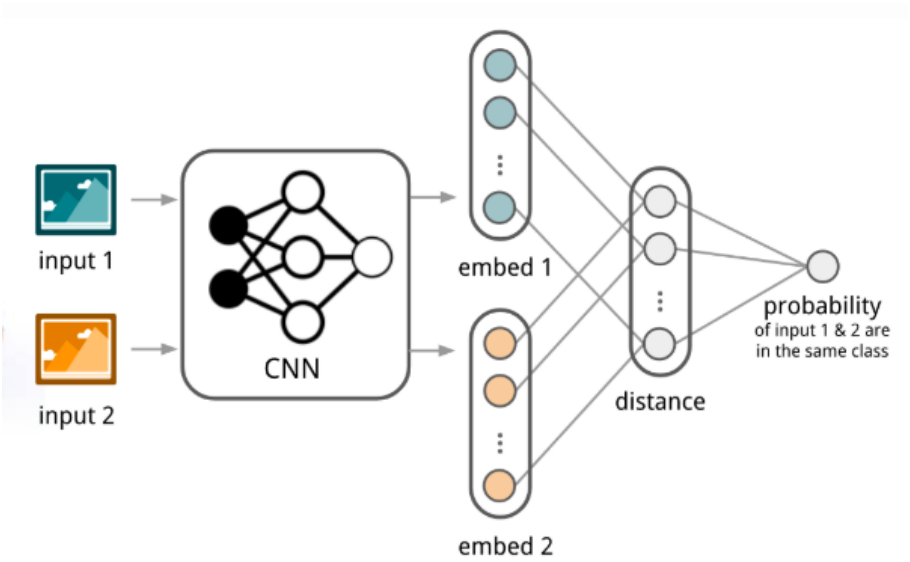
descrição especifica dos recursos, bibliotecas e linguagem para desenvolver o experimento, coisas que ficaram faltando na maioria dos trabalhos analisados. Foi usada uma placa gráfica GPU NVIDIA GTX1060, a linguagem Python e a biblioteca Keras. O conjunto de dados foi dividido em 2000 imagens para treinamento e 200 para testes. Consideramos que por ser um banco de dados que disponibiliza 625.537 imagens de 10.177 indivíduos, deveria ter sido considerado um número maior na mostra para tornar o estudo mais significativo. Os resultados mostram uma precisão de 87% durante o teste, uma precisão de 96,9% durante o treino e 94,7% durante a validação cruzada e um ROC de 0,535.

3.4 Redes Neurais Siamesas

Nas arquiteturas de Deep Learning, as redes siamesas (SNN) apresentadas por Bromley et al. (BROMLEY et al., 1994) propõem uma estrutura interessante para este trabalho. A aceitação das SNN e seus resultados para o problema do *One-shot* na área de visão computacional (KOCH; ZEMEL; SALAKHUTDINOV, 2015) despertou grande interesse na área em encontrar semelhanças entre duas entradas (BERTINETTO et al., 2016) (GUO et al., 2017).

Em geral, as redes siamesas podem ser vistas como duas sub-redes idênticas que compartilham os mesmos pesos e realizam a extração de características de cada uma das entradas, cada saída da rede é um vetor de características que serão comparados com seu par para verificar sua similaridade (BROMLEY et al., 1994) a Figura 3-3 mostra um exemplo de uma SNN. O treinamento de redes siamesas com funções de perda comparativas resultou em melhor desempenho, o que mais tarde levou à função de perda tripla usada no sistema FaceNet pelo Google, que alcançou resultados de ponta em tarefas de reconhecimento do rosto (SCHROFF; KALENICHENKO; PHILBIN, 2015).

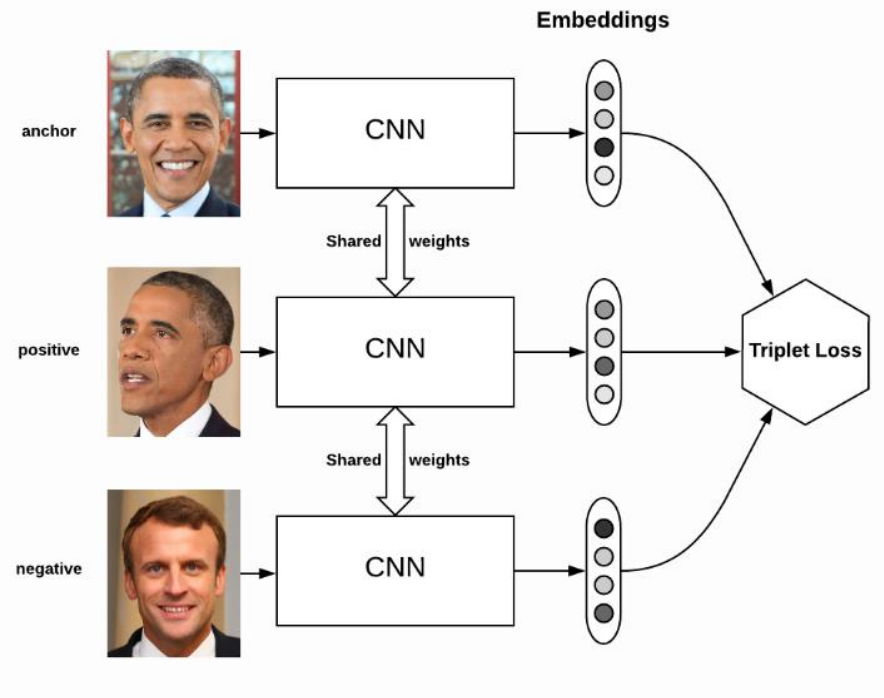
Figura 3-3: Arquitetura SNN



Fonte: (L WENG, 2018)

A ideia de uma função de perda tripla é estabelecer uma imagem âncora e fazer uma comparação com uma da mesma classe ou similar com outra de uma classe diferente, como mostrado na (Figura 3-4). O objetivo da função de perda tripla é reduzir a distância entre as duas imagens da mesma classe e aumentar a distância entre as duas imagens de classes diferentes (SCHROFF; KALENICHENKO; PHILBIN, 2015). Alguns trabalhos já foram apresentados na área de biometria contra ataques de falsificação em áreas como reconhecimento de voz (SRISKANDARAJA; SETHU; AMBIKAIRAJAH, 2018), impressões digitais (PALA; BHANU, 2017) e íris (PALA, FEDERICO AND BHANU, 2017) com o uso de SNN e funções de perda tripla.

Figura 3-4: Definição de função de perda tripla



Fonte: (MOINDROT, 2018)

Hao em (HAO; PEI; ZHAO, 2019) propõe um método anti-spoofing facial usando uma rede siamesa com entrada de dois pares real-real e real-falso. Para cada sub-rede, eles usam uma rede convolucional baseada na arquitetura AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Para encontrar a distância entre as saídas de cada uma das redes, além disso usam a função de perda *Dimensionality Reduction by Learning an Invariant Mapping* (DrLIM) apresentado por Hadsell em (HADSELL; CHOPRA; LECUN, 2006). Seu experimento mostra resultados relevantes na área, realizando testes em bancos de dados como NUAA e *Replay Attack*, com resultados de 1,96% HTER e 0,86% HTER, respectivamente.

3.5 Novos desafios

Nos últimos anos, surgiram outros tipos de PAI que tem sido pouco tratados e estão contornando com sucesso os sistemas baseados em reconhecimento facial. Liu em (LIU et al., 2019) define esses novos tipos de ataques como *Zero-Shot Face Anti-*

spoofing (ZSFA) e são analisados 13 tipos de ataques que incluem impressões e reproduções a (Figura 3-5) mostra exemplos dos novos tipos de ataques que são contemplados. Também é apresentado um novo banco de dados com esses novos tipos de ataque *Spoof in the Wild – Multiple* (SiW-M) como uma extensão do proposto anteriormente por eles em (LIU; JOURABLOO; LIU, 2018).

Figura 3-5: Novos tipos de ataques de apresentação detectados

(a)

(b)

(c)

a) Ataques de máscaras 3D
b) Ataques de maquiagem
c) Ataques parciais

Fonte: (LIU et al., 2019)

A arquitetura proposta nesse caso é uma *Deep Tree Network* (DTN), treinada de maneira não supervisionada para encontrar recursos com a maior variação possível para dividir os dados falsos. O sistema é testado nos bancos de dados CASIA, *Replay Attack*, MSU-MFSD e SiW-M. As métricas usadas para a avaliação são APCER, BPCER, ACER, EER, AUC. Sua proposta atinge 95,9% de AUC em CASIA, *Replay*

Attack, MSU-MFSD, em média, entre diferentes categorias de ataque em comparação com *state-of-the-art* (SOTA) e um EER de 16,1% e ACER de 16,8%.

3.6 Discussão final do capítulo

A tabela 3-1 resume os principais aspectos discutidos no capítulo. A tabela contém a metodologia, o método de extração de recursos, os classificadores e o banco de dados usados nos trabalhos relacionados. Por fim, a última coluna mostra os melhores resultados alcançados tendo em conta o classificador e o banco de dados que aparecem destacados em negrito.

Tabela 3-1 Resumo dos trabalhos relevantes

Autores	Metodologia	Extração de características	Classificador (es)	Banco de dados utilizados	Melhores Resultados
Määttä (MÄÄTTÄ; HADID; PIETIKÄINEN, 2011)	Análise de microtextura	LBP	SVM	NUAA	EER = 2.9%
Chingovska (CHINGOVSKA; ANJOS; MARCEL, 2012)	Análise de microtextura	LBP	**SVM** LDA Chi-square	**NUAA** Replay Attack Casia	HTER = 4.23%
Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2015b)	Análise de espaço de cores (YC_bC_r +HSV)	LBP	SVM	**Casia** Replay Attack	EER = 6.2%
Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2016)	Análise de espaço de cores (YC_bC_r +HSV)	LBP CoALBP LPQ BSIF SID	SVM	**Replay Attack** Casia MSU MFSD	EER = 0.4% HTER = 2.8%
Yang (YANG; LEI; LI, 2014)	Análise baseado em diferentes escalas de imagens	CNN	SVM	**Casia** Replay Attack	EER = 4.64% HTER < 5%
Li (LI et al., 2016)	Análise dos componentes principais (PCA)	CNN (DPCNN)	SVM	**Replay Attack** Casia	EER = 2.9% HTER = 6.1%
Atoum (ATOUM et al., 2017)	Patches e mapas de profundidade (canais HSV+YCbCr)	CNN LBP	SVM	**MSU-USSA** Replay Attack Casia	EER = 0.35% HTER = 0.21%
Liu (LIU; JOURABLOO; LIU, 2018)	Análise de profundidade e rPPG	CNN RNN	CNN RNN	**SiW** Replay Attack Casia Oulu	ACER = 3.58%

Liu (LIU et al., 2019)	Análise de novos ataques (ZSFA)	DTN	DTN	**SiW-M** CASIA Replay Attack MSU-MFSD	ACER = 16.8% EER = 16.1% AUC = 95.9%
Hao (HAO; PEI; ZHAO, 2019)	Arquitetura SNN	CNN	DrLIM (Distância)	**Replay Attack** NUAA	HTER = 0.86%
Garg (GARG et al., 2020)	Arquitetura DeBNet	LBPSVM LBPDBN DeBNet	SVM DeBNet	**NUAA**	HTER = 0,31%
Mohamed (MOHAMED et al., 2021)	CNN	CNN	CNN	**CelebA Spoof**	ROC = 0,535

Fonte: Autor

Alguns autores, com o objetivo de demonstrar escalabilidade do sistema desenvolvido, utilizam o método de validação cruzada usando diferentes banco de dados. Liu (LIU; JOURABLOO; LIU, 2018) realiza o treinamento com um banco de dados e os testes com outro. Outros autores como (BOULKENAFET; KOMULAINEN; HADID, 2016; YANG; LEI; LI, 2014) utilizam mais de um banco de dado para treinamento ou teste. Segundo a análise realizada dos trabalhos existe um aumento do EER e HTER ao combinar diferentes bancos de dados, embora utilizando o mesmo banco de dados para treinamento e teste se consegue melhores resultados.

Ainda assim, o desafio persiste em combinar diversos banco de dados, com o objetivo de testar e validar o método treinado em diferentes condições e ambientes, e tentar diminuir os erros (EER e HTER) nos reconhecimentos. Para isso, Liu (LIU; JOURABLOO; LIU, 2018) realizou uma tabela resumo onde comparou os resultados alcançados por vários trabalhos relacionados ao utilizar diferentes métodos.

Baseados na tabela resumo comparativa realizada por Liu (LIU; JOURABLOO; LIU, 2018) foi criada a tabela 3-2 comparando os resultados dos trabalhos estudados neste capítulo que realizam uma validação cruzada entre diferentes bancos de dados. O primeiro banco de dados de cada caso foi usado para treinamento e os restantes para teste.

Tabela 3-2 Resultados da combinação de dois bancos de dados diferentes

Autores	**Banco de dados**	**Melhores resultados (HTER)**
Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2015b)	Replay Attack CASIA FASD	39.6%
Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2016)	CASIA FASD Replay Attack MSU MFSD	30.3%
Yang (YANG; LEI; LI, 2014)	Replay Attack CASIA FASD	45.5%
Liu (LIU; JOURABLOO; LIU, 2018)	CASIA FASD Replay Attack	27.6%

Fonte: Autor

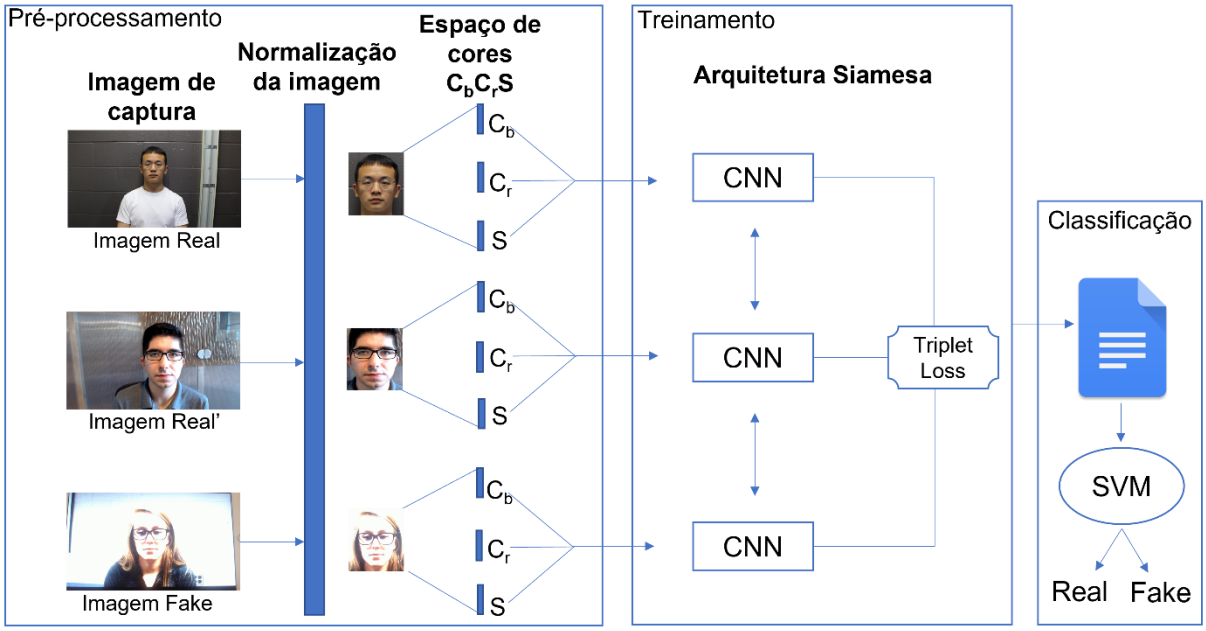
Capítulo 4

4 Método proposto

Neste capítulo é proposto um método para a detecção de ataques de suplantação de identidade por reconhecimento facial. Para identificar uma tentativa de suplantação de identidade por representação facial, nesta proposta não é contemplado o uso de um sensor externo, nem se destina a fazer uso de mecanismos intrusivos, descartando assim a implementação de mecanismos baseados em hardware. Adoptando assim uma abordagem baseada em software com um método estático que em comparação com os métodos dinâmicos têm menores tempo de implementação, velocidade de processamento e menores exigências de poder computacional como foi conceitualizado na seção 2.5.

Por fim, com base nessas conclusões, este trabalho se concentrará em encontrar uma solução por meio de um algoritmo PAD baseado em software sob um método estático, a partir da análise dos trabalhos relacionados do capítulo 3. Na Figura 4-1 se apresenta o método proposto e o ciclo de processamento desde a captura da imagem e seu pre-processamentoo, o treinamento da rede neural siamesa com uma rede convolucional e uma função de perda de trigêmeos, e finalmente a classificação das imagens entre reais e falsas. Nas próximas seções são aprofundadas cada umas das partes como: o processamento das imagens, a arquitetura *Deep Learning*, o mecanismo para a extração de caraterísticas, função de similaridade, o classificador e métricas de avaliação.

Figura 4-1: Método proposto



Fonte: Autor

4.1 Bancos de dados selecionado

Na seção 2.8 foram apresentados os principais bancos de dados para treinamento, teste e as principais caraterísticas deles. Neste trabalho é proposto a utilização dos bancos de dados *Replay Attack* (Figura 4-2) e SiW (Figura 4-3) e aspectos como variedade de meios de captura, de usuários que participam, condições de luminosidade e posturas foram fatores decisivos na sua escolha. O amplo uso do *Replay Attac*k nos trabalhos relacionados apresentados no capítulo 3 e os resultados obtidos com ele apresentados na tabela 3-1, também foram critérios de seleção. Por outro lado, cabe ressaltar que SiW é um banco de dados recente com variedade de luminosidade, postura e alta qualidade de resolução dos PAI e que também mostra bons resultados na análise feita na tabela 3-1.

Figura 4-2: Exemplo de banco de dados *Replay Attack*

Fonte: (CHINGOVSKA; ANJOS; MARCEL, 2012)

Figura 4-3: Exemplos de imagens reais (fila superior) e imagens falsas (fila inferior) no SiW

Fonte: (LIU; JOURABLOO; LIU, 2018)

O uso de dois bancos de dados é uma proposta deste trabalho a partir da análise feita na seção 3.7 em relação a utilização de mais de um para fazer os treinamentos e provas indistintamente. As diferenças das condições como a qualidade dos meios de capturas e dos PAI, condições do ambiente, etnias e posturas contribuirão para que o sistema seja mais robusto e mais abrangente. Outro aspecto a avaliar é se os bancos de dados selecionados têm rotulado cada um dos vídeos o que vai ser positivo na hora do treinamento supervisionado. O ponto fraco nestes bancos de dados é que não contemplam o uso de máscaras.

4.2 Extração e normalização de imagem

A proposta é utilizar 5 quadros de imagens de cada um dos vídeos, tanto reais como falsos e obter diferentes momentos durante o vídeo onde poderá variar a postura, brilho, luminância, reflexo etc. (Figura 4-4). Essas características podem ser

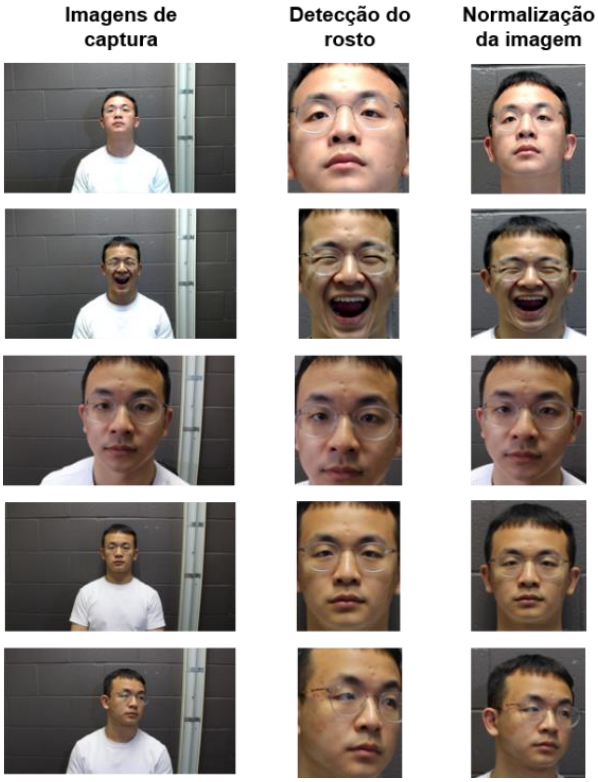
discriminativas na hora de fazer a análise de imagens falsas sobre telas do computador, do celular ou impressões.

Figura 4-4: Captura dos quadros de vídeo em diferentes instâncias de tempo

Fonte: Autor

Cada um dos bancos de dados selecionados fornece os pontos de coordenadas para indicar a localização da face ou a região de interesse na imagem. Isso facilitou não ter que implementar o algoritmo para a detecção do rosto. Este trabalho também visa obter características não só do rosto, senão também da região do fundo como foi analisado no método proposto por (YANG; LEI; LI, 2014). Ele utilizou 5 escalas diferentes, mas neste caso só vai ser implementada a escala que apresentou os melhores resultados como mostra-se na Figura 4-5. Além disso a resolução das imagens será normalizada a 128x128 pixels, sendo que valores menores podem trazer perda de muitas informações relevantes na extração de características. Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2016) e Hao (HAO; PEI; ZHAO, 2019) tem arquiteturas e métodos similares à proposta que se apresenta. Maior resolução pode contribuir a melhores resultados, mas também exigem maior tempo de processamento e recursos computacionais, considerações que não serão contempladas no método proposto no capítulo.

Figura 4-5: Processamento e normalização da imagem



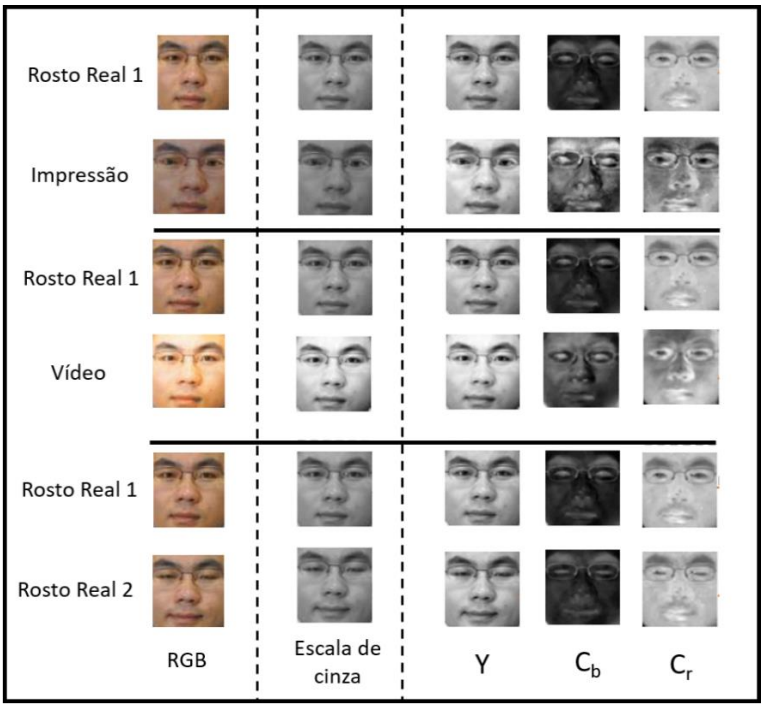
Fonte: Autor

4.3 Espaço de cores e canais

Além da etapa de pré-processamento anterior, também se concebe neste método as propostas das análises de textura nos canais cromáticos de Boulkenafet nos anos (BOULKENAFET; KOMULAINEN; HADID, 2015a) e (BOULKENAFET; KOMULAINEN; HADID, 2016). Esses trabalhos, já analisados no capítulo 3, visualizam as dificuldades que podem trazer só a análise de luminância nas imagens. Na proposta, também focada na detecção de suplantação de identidade, Boulkenafet percebe as diferenças que podem ser observadas em diferentes espaços de cores como: escala de cinza, RGB, YC_bC_r e HSV em diferentes PAI com relação a imagens reais. Essas diferenças também se esperam que sejam notáveis no método proposto deste trabalho, dado que também utilizam um dos bancos de dados que apresentamos na secção 4.1.

Outro aspecto a ter em conta é que o autor trabalha com vários descritores para a extração das características em cada um dos canais, sendo diferente da nossa proposta, que é fazer essa função com ajuda de uma CNN apresentada na seção 4.4.2. A partir da análise e resultados apresentados por Boulkenafet os espaços de cores YC_bC_r e HSV foram os escolhidos para desenvolver a proposta. A partir desses espaços de cores foi criado um que será a entrada de nossa rede. Boulkenafet, segundo os resultados nos experimentos feitos, aponta as maiores diferenças nos canais cromáticos C_b e C_r entre imagens reais e falsas como se apresenta na Figura 4-6.

Figura 4-6: A imagem mostra as diferenças mais significativas entre rosto reais e ataques por impressão e vídeo nos canais C_b e C_r.



Fonte: Autor adaptado de (BOULKENAFET; KOMULAINEN; HADID, 2016)

Nessa proposta se faz uma combinação entre os dois espaços de cores fazendo a exclusão de alguns dos seus canais. No espaço YC_bC_r não é considerado o canal Y que representa a luminosidade devido que não aporta diferenças significativas ao análise entre uma imagem real e uma imagem falsa como mostra a Figura 4-6. No espaço de cores HSV não é considerado o canal V por a semelhança com o canal Y

referente a luminância e desconsiderado o canal H core dentro desse espaço. Essa última escolha considerou não misturar os canais de cores C_b C_r e H por suas semelhanças e ficar apenas com C_b C_r como os canais que representaram as cores separadas. No final é considerada a saturação (S) para conseguir variações dentro das mesmas cores, desse jeito teremos uma imagem do tipo [C_b, C_r, S] como entrada de nossos dados para treinamento.

4.4 Arquitetura Siamesa

Tradicionalmente, uma rede neural aprende a prever várias classes. Isso representa um problema quando precisamos adicionar ou remover novas classes de dados. Nesse caso, precisamos atualizar a rede neural e treiná-la novamente em todo o conjunto de dados. Além disso, redes neurais profundas precisam de um grande volume de dados para serem treinadas. As SNNs, por outro lado, aprendem uma função de similaridade. Assim, podemos treiná-la para verificar se as imagens são iguais e isso permitirá classificar novas classes de dados sem treinar a rede novamente.

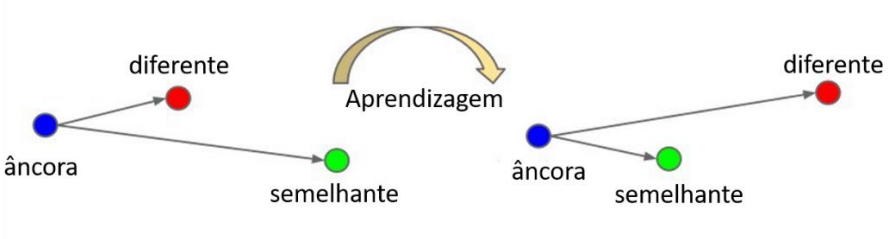
As SNNs são modelos que contêm duas ou mais sub-rede que compartilham seus pesos. Cada sub-rede recebe uma entrada diferente, e o modelo aprende uma função para medir a similaridade entre as entradas. No geral, as redes siamesas não são usadas como classificadores, mas sim como um meio para gerar características em um espaço de similaridade semântica. Após a aquisição das características, classificadores clássicos são utilizados.

Os argumentos mencionados anteriormente motivarão o estudo das SNN como uma possível solução ao problema que se apresenta de suplantação de identidade. Os estudos feitos na área de *Computer Vision* e especificamente os resultados alcançados na área de *Face Recognition* e sistemas PAD analisados no capítulo 3 incentivam o estudo e mostram altas possibilidades de aportes significativos ao método proposto.

4.4.1 Função de Perda de Trigêmeos

No ano (SCHROFF; KALENICHENKO; PHILBIN, 2015) Schroff propõe a função de Perda de Trigêmeos (*Triplet Loss)* apresentada no capítulo 3. Como foi explicado, a ideia era manter uma imagem âncora e fazer a comparação dela com uma de igual classe e outra de diferente classe, com o objetivo de reduzir e ampliar a distância entre elas respectivamente (Figura 4-7).

Figura 4-7: Operação da função *Triplet Loss*



Fonte: Autor adaptado de (SCHROFF; KALENICHENKO; PHILBIN, 2015)

Para calcular a perda de trigêmeos o objetivo é construir trigêmeos âncoras, positivos e negativos. A partir de uma imagem âncora, uma imagem positiva vai ser semelhante à imagem âncora e uma imagem negativa vai ser diferente da imagem âncora. A forma de saber se são imagens iguais ou diferentes será através dos rótulos já definidos nos bancos de dados escolhidos (*Replay Attack* e SiW) como real e falso. As imagens extraídas da mesma classe poderão ser consideradas semelhantes e as imagens de outras classes poderão ser consideradas como diferentes.

A proposta é inserir três imagens na rede divididas nos canais de espaço de cores C_bC_rS. Cada uma das imagens será analisada por uma CNN que compartilha os pesos e a mesma arquitetura para a extração de características. A codificação final da saída de cada uma das redes representa o valor de cada uma das imagens que vai ser calculado pela função *Triplet Loss*. O resultado da função visa aumentar a distância entre imagens diferentes e diminuir a distância entre imagens de igual classe para fazer os reajustes da rede durante o treinamento.

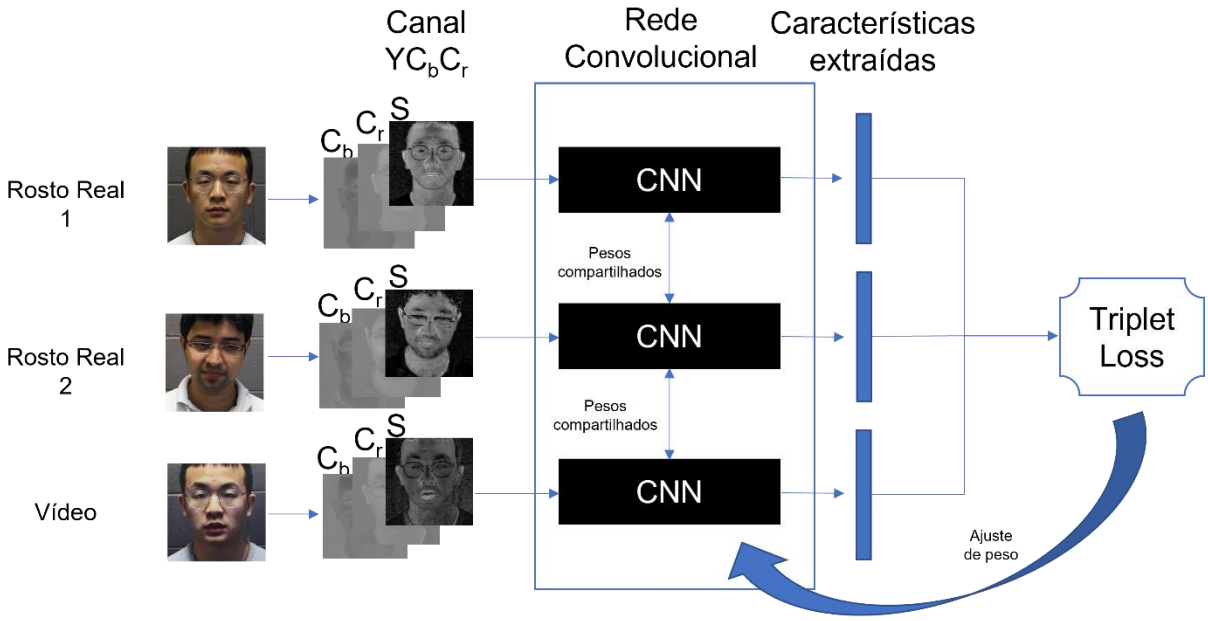
4.4.2 Rede convolucional

Com o aprendizado supervisionado as redes convolucionais em suas camadas tentam imitar o córtex visual do olho humano para identificar características diferentes nas entradas que possibilitam a identificação de objetos. Para isso, a CNN contém várias camadas ocultas especializadas com uma hierarquia: isso significa que as primeiras camadas podem detectar linhas, curvas e se especializar até atingir camadas mais profundas que reconheçam formas complexas, como um rosto ou a silhueta e, no caso deste trabalho, a textura.

A ideia de usar uma rede neural convolucional é seu amplo uso na área de *Computer Vision* para a análise de imagens digitais. Analisando diferentes CNN's para a extração de características, o método que se apresenta propõe utilizar AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) que é uma rede muito estudada e implementada com ótimos resultados em diferentes tipos de classificações. Foi utilizada por (HAO; PEI; ZHAO, 2019), trabalho relacionado já analisado, e que tem também como arquitetura principal uma rede siamesa para resolver o problema de suplantação de identidade facial.

A rede vai ter como entrada as três imagens já definidas referentes aos canais C_b, C_r, S e normalizadas a partir da proposta da secção 4.2. As três redes compartilham suas características e estrutura como se define na arquitetura siamesa. As características de saída de cada uma das redes são utilizadas para o cálculo da função *Triplet Loss*. Os valores do cálculo da função de perda contribuem ao ajuste da rede no momento do treinamento. Na Figura 4-8 é representada a arquitetura baseada no método proposto.

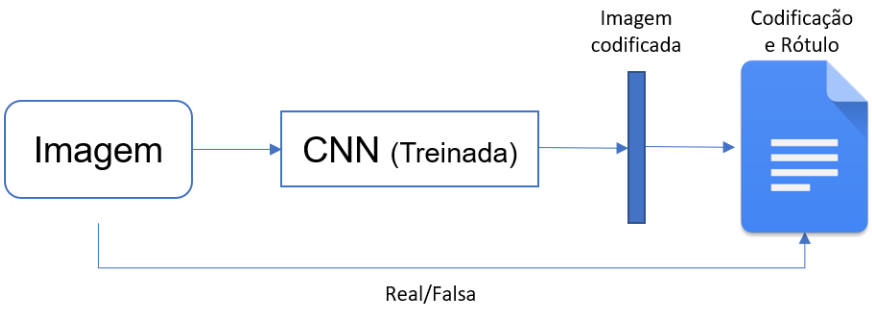
Figura 4-8: Arquitetura siamesa com a função *Triplet Loss* proposta



Fonte: Autor

Depois que temos a rede ajustada a partir dos cálculos feitos pela função de perda é preciso obter a codificação de cada uma das imagens com o modelo já treinado. Durante a primeira etapa, como mostra a Figura 4-8, somente a rede estava modificando os pesos, por isso no momento de saída da imagem ainda não se tem o valor desejado da sua codificação. Uma vez que acabou o processo de aprendizagem da rede todos os pesos já estão ajustados e é momento de conhecer os valores para cada uma das imagens da última camada.

Figura 4-9: Obtenção da codificação da imagem



Fonte: Autor

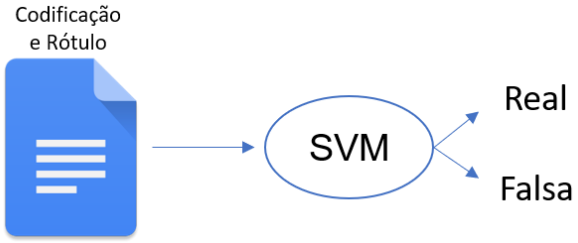
Posteriormente cada um dos valores das imagens serão salvados em um arquivo com o rótulo de imagem (real ou falsa) correspondente como se observa na Figura 4-9. O rótulo é conhecido de cada uma das imagens capturadas dos vídeos dos bancos de dados. Esse arquivo que representa a codificação atribuída pela última camada da rede a uma imagem e seu respectivo rótulo, serão a entrada na etapa de classificação das imagens.

4.5 Classificador SVM

O classificador SVM foi abordado na seção 2.6.2.1 deste trabalho e os trabalhos mais relevantes referentes a sistemas de suplantação que utilizam SVM foram apresentados no capítulo 3. Yang no ano (YANG; LEI; LI, 2014) além de usar uma rede convolucional também fez a classificação do seu método com um SVM e consegue atingir resultados relevantes na área. A tabela 3-1 mostra o principal resultado do seu experimento, que é analisado no próprio capítulo. Trabalhos como os de Tang (TANG, 2013) e do Wu (WU et al., 2018) explicam como podem ser implementados conjuntamente CNN e SVM, e como uma camada linear de SVM pode melhorar a classificação de uma camada totalmente conectado (*fully connected*) com uma função *Softmax*.

A partir da ideia dos trabalhos anteriores o tamanho da convolução e os mapas de características têm grandes influências no desempenho da classificação do CNN e estão sujeitos aos números de neurônios de entrada e saída das camadas totalmente conectadas. Portanto o classificador SVM nesta proposta realizará a classificação substituindo as camadas totalmente conectadas após o classificador CNN ser bem treinado. O classificador a partir do arquivo salvado com a codificação de cada uma das imagens e os rótulos correspondentes será o responsável por fazer a distinção entre o rosto real e o rosto falso como na Figura 4-10.

Figura 4-10: Modelo de classificação das imagens



Fonte: Autor

4.6 Validação do método

Referente à validação do método, no capítulo 3 foi realizado um levantamento das principais métricas usadas nos trabalhos relacionados. Neste trabalho será utilizada a métrica HTER assim como outras métricas como a Acurácia, Predição, *Recall* e F1-Score. A métrica HTER, escolhida anteriormente é muito utilizada nos sistemas de reconhecimento facial e sistemas *anti-spoofing* para comparação e validação do sistema. Nesse capítulo também pode se observar que a maioria dos trabalhos relacionados validam os seus resultados fazendo uso delas, além de outras.

Como foi explicado no capítulo anterior o HTER é o erro médio calculado a partir da FAR e FRR como mostra a equação 1.

$$HTER = \frac{FAR + FRR}{2} \quad [1]$$

FAR corresponde à razão de falsos positivos e o número total de amostras negativas, ou seja, vai determinar a quantidade de imagens que foram classificadas pelo sistema como verdadeiras quando realmente não são. FRR corresponde à relação entre os falsos negativos e o número total de amostras positivas, mais

especificamente, a predição do sistema de imagens reais quando realmente são falsas.

As métricas propostas serão comparadas com alguns dos trabalhos mais importantes analisados no capítulo 3, que pela sua relevância e resultados foram escolhidos como referência nesta pesquisa. O primeiro deles é a proposta de Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2016) que propõe a análise dos espaços de cores e dos canais cromáticos baseado na textura da imagem, método inovador que atingiu bons resultados e que forma parte da proposta deste capítulo no pré-processamento das imagens de entrada. O próximo trabalho selecionado foi de Yang (YANG; LEI; LI, 2014) que é considerado o primeiro que propõe as redes convolucionais para extração de características. Ele obteve bons resultados e também usou um classificador SVM como é considerado na nossa proposta. Por último, o trabalho de Hao (HAO; PEI; ZHAO, 2019), recente na área, considera uma nova arquitetura *Deep Learning* com resultados satisfatórios em reconhecimento facial. A sua proposta é baseada numa SNN com uma rede convolucional que também forma parte do método que será implementado no próximo capítulo.

Capítulo 5

5 Implementação e validação

Nesse capítulo será implementado e validado o método proposto e se apresentará o passo a passo de cada umas das etapas da solução. A partir das métricas selecionadas serão comparados os resultados obtidos com os dos trabalhos escolhidos.

5.1 Processamento de dados

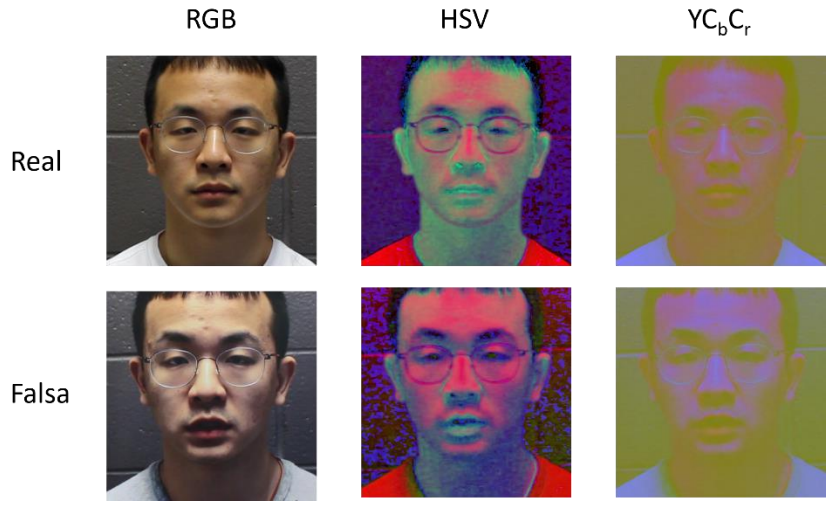
No primeiro momento são geradas as imagens a partir dos vídeos nos bancos de dados. Nossa proposta vai utilizar o banco de dados SiW para fazer o treinamento e validação da rede neural e os dados do banco Replay Attack para o teste. Um total de 5 imagens foram tomadas durante o vídeo para capturar diferentes posturas do rosto assim como mudanças nas condições de luminosidade do ambiente. Um total de 10.132 imagens foram extraídas de 165 pessoas diferentes entre reais e falsas do banco SiW. Dessas imagens serão utilizadas 7000 para treinamento e 3132 para validação. No caso do banco de dados Replay Attack foram capturadas apenas 3 imagens de cada vídeo já que tem menor variedade de condições do ambiente assim como posições do rosto. Um total de 2891 imagens foram geradas de 50 pessoas diferentes para usar na etapa final de testes da rede com o SVM. Manualmente foram desconsideradas algumas imagens em cada um dos bancos de dados por diversos motivos: imagens de perfil, duplicidade quanto à posição do rosto, expressões faciais ou luminosidade.

A seguir, foram realizados os ajustes dimensionais com um aumento do 10% na área de detecção do rosto. Isso permitirá obter algumas informações relevantes no processamento da imagem como foi mostrado na Figura 4-5 do capítulo anterior. Depois, as imagens foram salvas em um repositório local separadas em imagens para treinamento, validação e testes, e etiquetadas em real ou falsa.

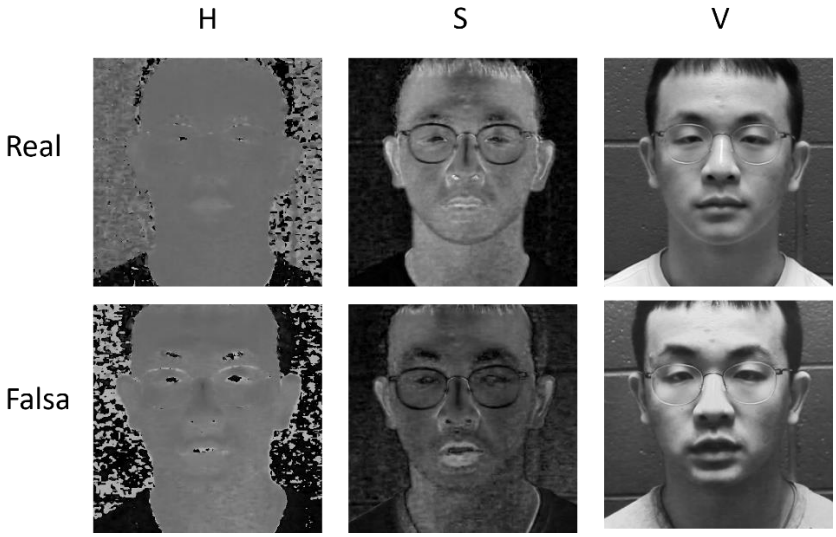
As imagens em um momento inicial estão em RGB e com a biblioteca opencv de Python foram convertidas aos espaços de cores HSV e YC_bC_r. Na Figura 5-1 (a) se

mostram duas imagens real e falsa em cada espaço de cores. As diferenças entre cada um dos canais para os espaços de cores HSV e YC_bC_r são mostrados na Figura 5-1 (b) e (c) respectivamente.

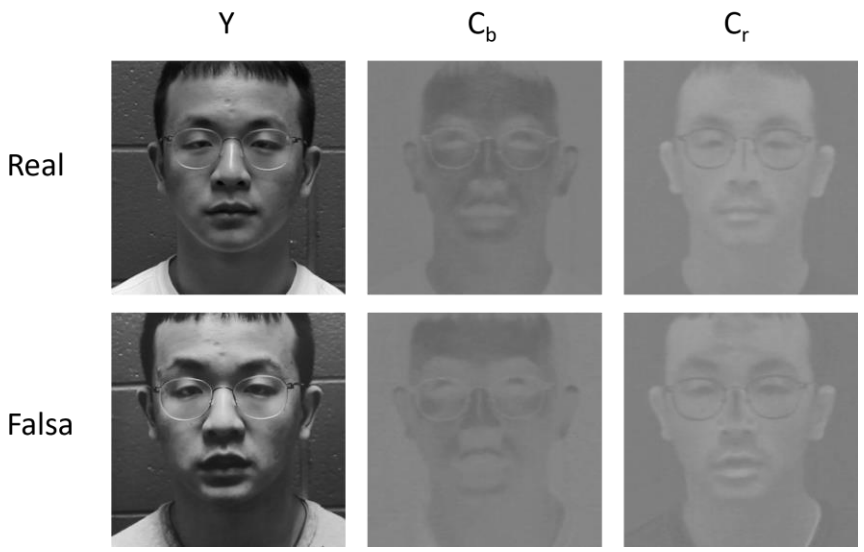
Figura 5-1: (a) Imagens real e falsa no espaço de cores RGB, HSV e YC_bC_r. (b) Canais do espaço de cores HSV. (c) Canais do espaço de cores YC_bC_r.



(a)



(b)

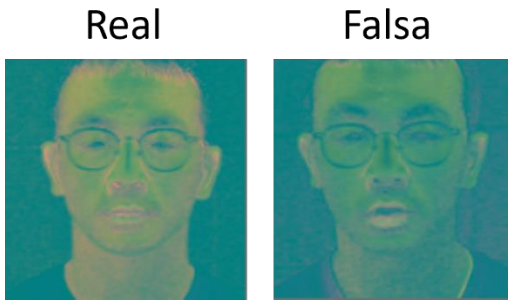


(c)

Fonte: Autor

Os canais C_b, C_r e S são unidos para formar a imagem C_bC_rS apresentada na Figura 5-2.

Figura 5-2: Novo espaço de cores CbCrS. Diferença entre a imagem real e falsa



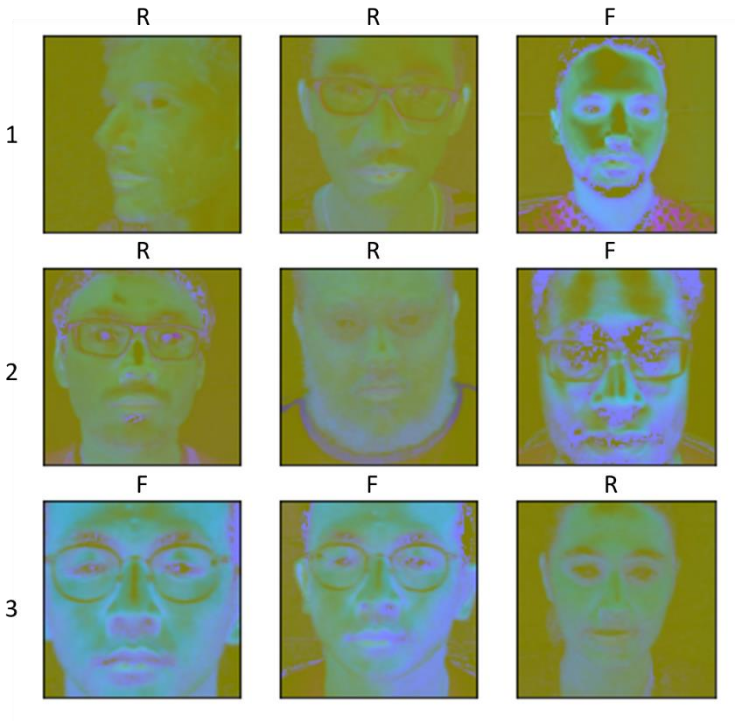
Fonte: Autor

Todas as imagens do novo espaço de cores são redimensionadas a 128x128 e salvas em arquivos .npy resultando em um *array* 128x128x3 para cada imagem. Os arquivos para treinamento, validação e teste resultaram em (7000, 128, 128, 3), (3132, 128, 128, 3) e (2891, 128, 128, 3) respetivamente, que representam a quantidade de imagens, dimensões e canais de cores. Além disso, foram criados os arquivos .npy com os valores 0 e 1 que correspondem à etiqueta de real ou falsa de cada uma das imagens. Esses arquivos salvam os valores de 0 e 1 em um *array* correspondendo

com a etiqueta da imagem e tem o mesmo tamanho dos arquivos de treinamento, validação e testes. Os arquivos em formato npy com as matrizes das imagens e os que tem os valores das etiquetas de real e falsa, foram salvos em um Drive de Google e carregados em Google Colab Pro para desenvolver as próximas etapas da proposta. Também foi utilizada uma GPU P100 e 24 GB de RAM. Toda a implementação e resultados ficaram salvos numa notebook .ipynb para futuros testes e melhoras.

Como definido, foi implementada uma rede neural Siamesa com função de perda *Triplet Loss.* Nesse ponto, uma etapa relevante no processamento dos dados foi criar os pares de trigêmeos que foram inseridos na rede. A função create_*batch* da solução recebe por parâmetros o tamanho do batch da rede e faz uma iteração de 0 até o tamanho definido do batch para gerar os trigêmeos das etapas de treinamento, validação e teste. Um valor aleatório dentro do conjunto de imagens foi selecionado sendo a imagem âncora e, uma vez obtida, foi procurada uma segunda imagem da mesma classe. Se a imagem escolhida como âncora é uma imagem real o algoritmo procurará outra imagem real e, no caso contrário, uma imagem falsa. A última imagem selecionada foi uma imagem de diferente classe em comparação com as outras duas. A Figura 5-3 mostra um exemplo de trigêmeos criados.

Figura 5-3: Diferenças entre trigêmeos aleatórios

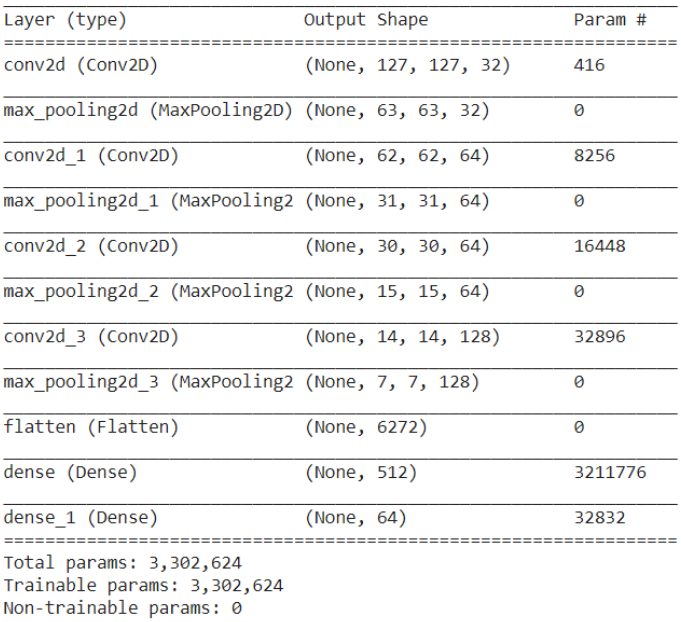


Fonte: Autor

5.2 Modelo Convolucional e Rede Siamesa

A rede Siamesa é constituída por 3 redes convolucionais idênticas. Cada rede recebe como parâmetro a imagem (*input_shape* = (128, 128, 3)) que representa as dimensões e canais C_bC_rS. A rede neural tem 4 camadas convolucionais e cada uma com uma função de ativação ReLu. Depois de cada camada de convolução é feito um *MaxPooling* e foram retornados 64 valores de características que representam a imagem. O modelo da rede é mostrado na Figura 5-4.

Figura 5-4: Arquitetura da rede convolucional



Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 127, 127, 32)	416
max_pooling2d (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_1 (Conv2D)	(None, 62, 62, 64)	8256
max_pooling2d_1 (MaxPooling2	(None, 31, 31, 64)	0
conv2d_2 (Conv2D)	(None, 30, 30, 64)	16448
max_pooling2d_2 (MaxPooling2	(None, 15, 15, 64)	0
conv2d_3 (Conv2D)	(None, 14, 14, 128)	32896
max_pooling2d_3 (MaxPooling2	(None, 7, 7, 128)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 512)	3211776
dense_1 (Dense)	(None, 64)	32832

Total params: 3,302,624
Trainable params: 3,302,624
Non-trainable params: 0

Fonte: Autor

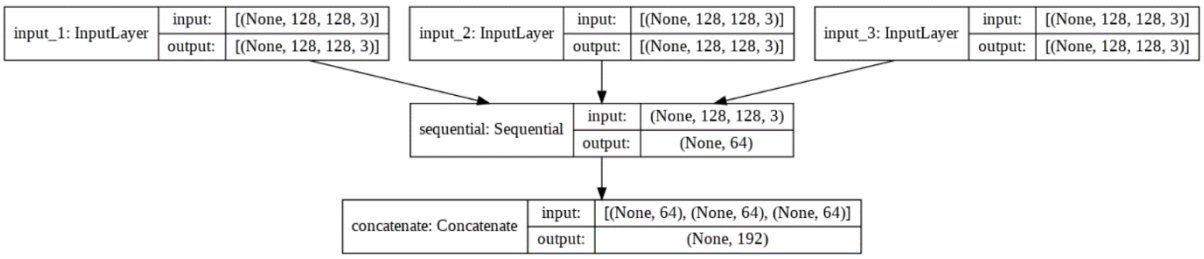
A entrada de cada uma das redes são as imagens geradas pela função *create_batch*. A imagem âncora foi inserida na primeira rede, a segunda imagem da mesma classe na segunda rede convolucional e a imagem de diferente classe na terceira rede. Como cada rede tem uma imagem foi preciso fazer uma concatenação entre todas elas para criar a rede siamesa e calcular a função *Triplet Loss*. A Figura 5-5 (a) e (b) representam a concatenação das três redes convolucionais criando assim a rede Siamesa.

Figura 5-5: Arquitetura da rede Siamesa

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 128, 128, 3)	0	
input_2 (InputLayer)	[(None, 128, 128, 3)	0	
input_3 (InputLayer)	[(None, 128, 128, 3)	0	
sequential (Sequential)	(None, 64)	3302624	input_1[0][0] input_2[0][0] input_3[0][0]
concatenate (Concatenate)	(None, 192)	0	sequential[0][0] sequential[1][0] sequential[2][0]

Total params: 3,302,624
Trainable params: 3,302,624
Non-trainable params: 0

(a)



(b)

Fonte: Autor

O objetivo da rede é diminuir a distância entre imagens da mesma classe e aumentar a distância entre imagens de diferentes classes. A função *Triplet Loss* é definida com a equação 2

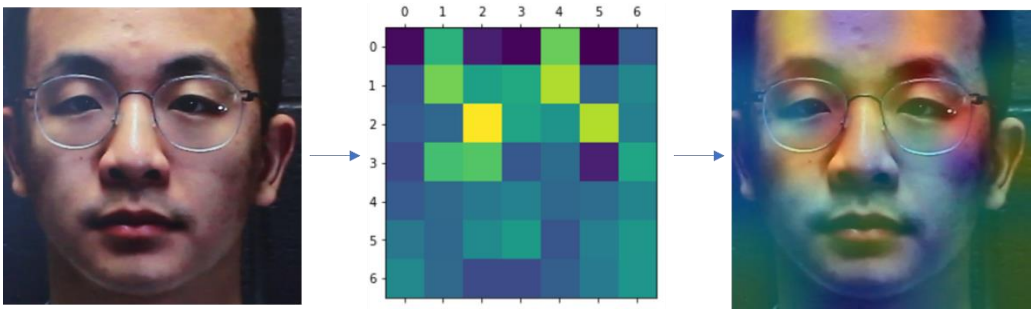
$$L = \max(d(A, P) - d(A, N) + \alpha, 0) \quad [2]$$

onde “A” é a entrada âncora, “P” a entrada positiva, “N” a entrada negativa e “α” a margem entre “P” e “N” que foi definida em 0,2. A função com o cálculo das distâncias entre as imagens determina o ajuste dos pesos da rede em cada iteração. O modelo foi compilado passando como parâmetros a função de perda *triplet_loss* implementada, um otimizador *adam* e a métrica *accuracy*. Depois de compilado o modelo foi treinado criando um gerador de dados que utiliza a função create_batch já

comentada. Os parâmetros para treinamento foram a função *data_generator* com os dados das imagens para treinamento, com 200 épocas, e 109 passos por época, sendo esse último calculado pela divisão entre o total de amostras de treinamento (7000) e o tamanho do *batch* definido em 64. O tamanho do lote (*batch*) limita o número de amostras a serem analisadas pela rede antes que uma atualização de peso seja realizada. Para a validação dos dados foi chamada a mesma função de *data_generator* com as imagens de validação. Com o total de 3132 imagens de avaliação o sistema consegue atingir um 97,37% de acurácia e uma perda de 0,0298. O modelo foi salvo com os pesos para posteriores análises.

Para fazer uma avaliação adicional do sistema proposto, foram criados alguns casos de exemplos para verificar a veracidade dos resultados e os cálculos feitos. O primeiro teste realizado foi usando o algoritmo chamado Grad-CAM para visualizar os mapas de ativação de classes. Como os algoritmos de aprendizado são uma caixa preta, é possível que em diferentes etapas ou camadas a rede comece 'olhar' no lugar errado. O *Gradient-weighted Class Activation Mapping* ou Grad-CAM, permite validar visualmente para onde a rede está 'olhando', verificando se ela está realmente obtendo os valores da área de interesse na imagem. Foi analisada a última camada da rede e a Figura 5-6 apresenta mediante um mapa de calor a área de análise da rede. Isso permite verificar que a região de interesse se encontra bem distribuída na imagem, o que permite analisar maior quantidade de caraterísticas, e não em uma área especifica da imagem.

Figura 5-6: Mapa de calor da área de interesse para processamento



Fonte: Autor

O segundo teste proposto consiste em implementar uma funcionalidade para determinar, mediante a similaridade de cosseno, imagens de classes iguais ou diferentes. A Figura 5-7 (a) mostra 3 imagens aleatórias do sistema, sendo as duas primeiras reais e a última falsa. A Figura 5-7 (b) apresenta os resultados da comparação entre a imagem âncora - positiva e âncora – negativa. O valor mais próximo de 1 representa maior similaridade entre imagens, e sendo mais próximo de 0 o contrário. O valor de predição é um *array* de 192 valores, 64 características de cada umas das imagens. O valor de 0 até 64 do *array* representa a predição do sistema para a primeira imagem, de 64 até 128 seria a segunda e, por fim, de 128 até 192 a terceira imagem.

Figura 5-7: (a) Trigêmeos aleatórios para cálculo de similaridade de cosseno. (b) Resultado do cálculo de similaridade de cosseno entre imagem real-real e real-falsa

(a)

```
[ ]  print("similarity-score between positive and anchor image: ", cosine_similarity(prediction[0][:64], prediction[0][64:128]))

     similarity-score between positive and anchor image:  0.9482001

[ ]  print("similarity-score between anchor and negative image: ", cosine_similarity(prediction[0][:64], prediction[0][128:]))

     similarity-score between anchor and negative image:  0.32821876
```

(b)

Fonte: Autor

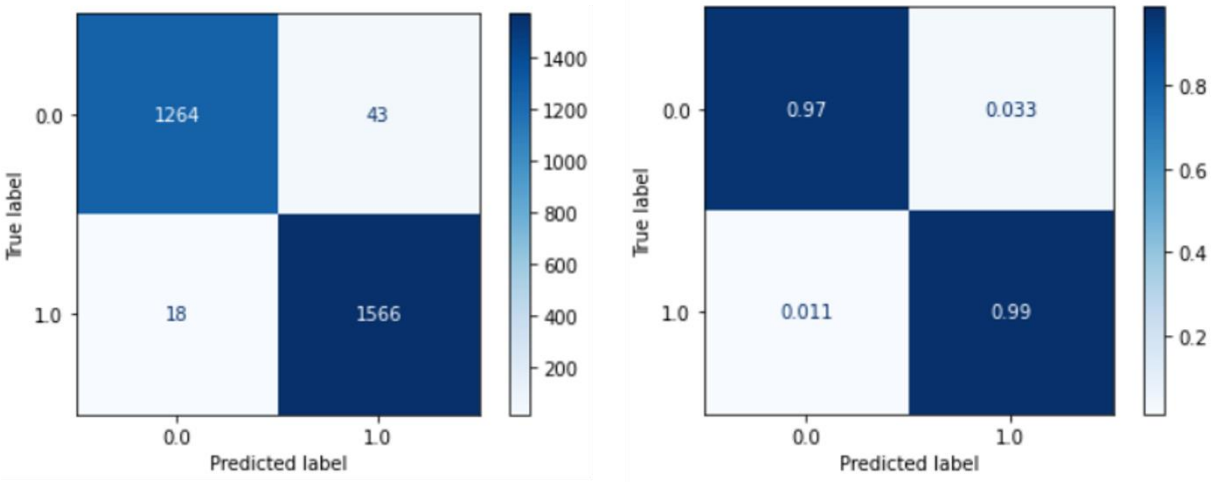
A partir do resultado satisfatório alcançado na etapa de validação do sistema foi criado o arquivo treinamento_svm.csv com os valores de predição de cada imagem de treinamento e validação com os pesos da rede já ajustados. Além disso, foi criado o arquivo treinamento_label.csv com a etiqueta que identifica as imagens como real ou falsa. Os arquivos salvos representam os *array* de características de cada imagem que serão utilizados para o treinamento do SVM.

5.3 Classificação e análise

Com a biblioteca *sklearn* do Python, foi criado o classificador SVM que recebe os dados de cada imagem junto com a etiqueta de real ou falsa. Foi considerado um kernel linear e para o treinamento foram utilizadas 10132 imagens. Esse é o total das imagens de treinamento e validação salvas no arquivo .csv. Uma vez concluído o treinamento do SVM, foi realizado a predição com dados que o sistema não conhecia. Esses dados de testes são as imagens extraídas do banco de dados Replay Attack, contendo diferentes cenários, usuários, etnias e qualidade de imagem referente ao utilizado para o treinamento.

Com a predição do sistema para os dados desconhecidos, é possível concluir que o classificador consegue índices similares à predição realizada pela rede Siamesa. Para levantar os pontos fracos e determinar quais tipos de imagens trazem maiores dificuldades para serem interpretadas e classificadas, foi criada uma matriz de confusão conforme a figura 5-8 que mostra uma comparação entre os dados reais ou iniciais do sistema com a predição do classificador.

Figura 5-8: Matriz de confusão do sistema



Fonte: Autor

Interpretando a tabela anterior, os valores no eixo Y (vertical) de 0.0 e 1.0 representam as imagens iniciais, reais e falsas do sistema. Os valores no eixo X (horizontal) representam as imagens reais e falsas resultado da predição do sistema. As células azuis do canto superior esquerdo e inferior direito contêm o número de amostras que o modelo previu com precisão. As células brancas contêm o número de amostras que foram previstas incorretamente.

O total de amostras no conjunto de teste é de 2891. Olhando para a matriz de confusão, é possível verificar que o modelo previu 2830 vezes com precisão a partir do total de amostras e apenas 61 previsões incorretas. Para as amostras que o modelo acertou, se observa que ele previu com precisão que as imagens seriam reais 1264 vezes e previu 18 vezes que as imagens eram reais quando não eram. Por outro lado, o modelo previu com precisão que as imagens eram falsas 1566 vezes e previu 43 imagens como falsas sendo que eram reais.

Para a validação foram utilizados os seguintes indicadores:

Acurácia: é o número de predições certas do total de predições feitas, equação 3.

$$Acurracy = \frac{Predições\ certas}{Predições\ total} \quad [3]$$

Precisão: é o número de Verdadeiros Positivos (VP) dividido pelo número de Verdadeiros Positivos e Falsos Positivos (FP). Em outras palavras, é o número de previsões positivas dividido pelo número total de valores de classe positivos previstos, equação 4.

$$Precisão = \frac{VP}{VP + FP} \quad [4]$$

Recall: é o número de VP dividido pelo número de VP somado ao número de Falsos Negativos (FN). Em outras palavras, é o número de previsões positivas dividido pelo número de valores de classe positivos nos dados de teste, equação 5.

$$Recall = \frac{VP}{VP + FN} \quad [5]$$

F1-*Score*: média ponderada da precisão e o recall, equação 6.

$$F1\ Score = \frac{2 * Precisão * Recall}{Precisão + Reacall} \quad [6]$$

Como resultados para cada uma das equações de validação se obteve uma **Acurácia** de 0,978, uma **Precisão** de 0,985, um ***Recall*** de 0,967 e finalmente um **F1-*Score*** de 0,975.

Com os valores de FAR e FRR de 0,011 e 0,033 respetivamente como mostra a figura 5-8, foi usada a equação (1) para validar o sistema e comparar os resultados das pesquisas, obtendo um valor de **HTER** de 2,2%.

5.4 Análise dos resultados

Na seção anterior foram apresentados os principais resultados do experimento. Os valores obtidos em cada uma das métricas usadas na etapa de classificação são considerados resultados alentadores e promissores. As métricas como a acurácia, precisão, recall e F1-*Score* representam valores acima de 95% o que pode ser considerado como um sistema que atinge um bom resultado no ambiente e condições que foram preestabelecidas. O resultado da métrica HTER de 2,2% é considerado um resultado competitivo referente às últimas pesquisas levantadas na área e nos trabalhos relacionados escolhidos. A tabela 5-1 apresenta os resultados desses trabalhos em comparação com a proposta.

Tabela 5-1: Comparativa do resultado da proposta com os trabalhos

Autores	Resultados (HTER)
Yang (YANG; LEI; LI, 2014)	< 5%
Boulkenafet (BOULKENAFET; KOMULAINEN; HADID, 2016)	2.8%
Hao (HAO; PEI; ZHAO, 2019)	0.86%
Proposta	**2,2%**

Fonte: Autor

Fazendo uma análise comparativa com os trabalhos estudados podemos concluir que na maioria deles não é possível, pela descrição do trabalho, reproduzir o experimento e validar os resultados obtidos. Outra caraterística é o uso de banco de dados que não são livres para fazer os testes e experimentos. Os recursos computacionais utilizados na maioria dos trabalhos não são comentados o que dificulta conhecer os tempos de processamento de treinamento e testes dos modelos. A variedade de bancos utilizados por cada proposta também dificultou na identificação de uma correlação entre trabalhos e conseguir comparações mais detalhadas para validar o modelo. Essas questões foram consideradas no início do trabalho e por esse motivo foram escolhidos trabalhos que por sua similaridade, relevância e resultados estivessem no escopo da nossa proposta.

No caso de Hao se considera que o valor ficasse embaixo do resultado desse trabalho por causa de não combinar diversos bancos de dados. As testes feitos no seu trabalho foram resultados obtidos fazendo treinamentos e testes com a mesma fonte de dados no caso NUAA e Replay-Attack. Boas práticas recomendam combinar diversos bancos de dados, com o objetivo de testar e validar o método treinado em diferentes condições e ambientes como foi considerado em esta proposta e como foi argumentado na seção 3.6.

Finalmente, é possível observar que os bancos de dados escolhidos representam pessoas de diferentes etnias, posturas, qualidade de imagem, entre outras características que garantem robustez no sistema em diferentes condições. É apresentado um novo espaço de cores a partir da união de diferentes canais cromáticos que proporcionam diferenças significativas para a análise, considerando-se uma escolha certa. Na maioria das imagens é possível definir a simples vista algumas diferenças entre as imagens reais e as falsas, facilitando o processamento delas dentro da rede neural e obter melhores resultados.

Com os dados pré-processados foi treinada uma rede siamesa simples por sua arquitetura, mas que consegue chegar a resultados significativos. Foram criados dois testes para validar que os dados estavam fornecendo informação útil e no caso contrário fazer os ajustes na rede. Depois de aplicados os testes mencionados anteriormente para validação, os resultados foram positivos o que proporcionou confiabilidade para continuar nas próximas etapas da solução.

Conclusões e trabalhos futuros

Nesta dissertação, foi apresentada a proposta para um método não intrusivo de detecção de ataques de suplantação de identidade por reconhecimento facial. Se realizou uma pesquisa do estado atual de sistemas para detecção desse tipo de ataques, assim como as principais ferramentas e algoritmos para sua detecção. Foi realizado um estudo rigoroso dos principais trabalhos relacionados que contemplam o uso de métodos não intrusivos como mecanismo *anti-spoofing*. Os trabalhos foram escolhidos por sua relevância, resultados e novidade na área. Essas pesquisas foram o ponto de partida e comparação com a proposta desse trabalho.

Também foi realizado um estudo dos principais bancos de dados na bibliografia para treinamento e teste de sistemas *anti-spoofing* por reconhecimento facial. Como resultado dessa análise foram escolhidos os bancos de dados SiW e Replay Attack para treinamento e testes respectivamente. Se realizou um estudo das principais técnicas e soluções para a extração de características das imagens e conseguir obter as diferenças entre imagens reais e falsas. A técnica escolhida foi a análise da textura das imagens a partir de diferentes canais de cromaticidade dentro de diferentes espaços de cores. Se considerou o uso de redes neurais, especificamente as redes convolucionais com bons resultados na área de reconhecimentos e extração de características. Para a classificação do sistema foi utilizado um SVM para definir se a imagem de entrada é uma imagem real ou falsa.

A arquitetura proposta tem uma primeira etapa de pré-processamento de imagens que cria um novo espaço de cores (C_bC_rS) a partir dos canais cromáticos dos espaços de cores YC_bC_r e HSV. Esses canais mostraram diferenças relevantes para a classificação das imagens. A imagem no novo espaço de cores é a entrada da rede neural siamesa na segunda etapa da proposta. A rede neural é composta por 3 redes convolucionais que fazem a análise de 3 imagens, duas da mesma classe e outra de uma classe diferente. Mediante a função *Triplet Loss* se calculam os pesos para garantir a menor distância entre as imagens da mesma classe e, a maior distância entre as de diferentes classes. Com a rede ajustada os dados são novamente inseridos para obter a codificação de cada umas das imagens. Na terceira parte da

solução, a codificação da imagem com a etiqueta de real ou falsa, são inseridas no SVM para treinamento. Com um novo banco de dados o sistema foi testado e validado com o uso de diferentes métricas.

Foram consideradas diferentes métricas para validar o sistema conseguindo valores competitivos como a acurácia de 97,8%, precisão de 98,5%, recall de 96,7% e F1-Score de 97,5%. O erro médio calculado a partir das taxas de falsa aceitação e falsa rejeição é uma medida de desempenho comumente usada em sistemas de autenticação para validar o sistema e fazer a comparação com outros trabalhos. Como resultado desta pesquisa se obteve um valor de HTER de 2,2% representando um resultado competitivo na área e especificamente em sistemas não intrusivos.

Como trabalhos futuros se recomenda implementar o sistema em um ambiente real. Adicionalmente, recomenda-se ampliar o banco de dados com mais características de ambiente e maior variedade de pessoas, podendo contribuir a resultados melhores e obter taxas de erro menores.

No sistema não foi contemplado o uso de máscara 3D e sendo um instrumento de ataque de apresentação, se recomenda incluir nas etapas de treinamento e testes banco de dados com dados dessa abordagem. Dessa forma seria possível garantir um sistema mais robusto.

Referências

AHONEN, T.; HADID, A.; PIETIKAINEN, M. Face Description with Local Binary Patterns: Application to Face Recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 28, n. 12, p. 2037–2041, dez. 2006.

AKBULUT, Yaman et al. Deep learning based face liveness detection in videos. In: **2017 international artificial intelligence and data processing symposium (IDAP)**. Ieee, 2017. p. 1-4.

ALI, Asad; DERAVI, Farzin; HOQUE, Sanaul. Liveness detection using gaze collinearity. In: **2012 Third International Conference on Emerging Security Technologies**. IEEE, 2012. p. 62-65.

ALI, Asad; DERAVI, Farzin; HOQUE, Sanaul. Directional sensitivity of gaze-collinearity features in liveness detection. In: **2013 Fourth International Conference on Emerging Security Technologies**. IEEE, 2013. p. 8-11.

ALOTAIBI, Aziz; MAHMOOD, Ausif. Deep face liveness detection based on nonlinear diffusion using convolution neural network. **Signal, Image and Video Processing**, v. 11, p. 713-720, 2017.

ALSAADI, Israa M. Physiological biometric authentication systems, advantages, disadvantages and future development: A review. **International Journal of Scientific & Technology Research**, v. 4, n. 12, p. 285-289, 2015.

ANDERSON, James A. A simple neural network generating an interactive memory. **Mathematical biosciences**, v. 14, n. 3-4, p. 197-220, 1972.

ATOUM, Yousef et al. Face anti-spoofing using patch and depth-based CNNs. In: **2017 IEEE International Joint Conference on Biometrics (IJCB)**. IEEE, 2017. p. 319-328.

AUBERT, A. **What is Object Detection?** Disponível em: <https://www.saagie.com/blog/object-detection-part1>. Acesso em: 5 mar. 2020.

BAI, Jiamin et al. Is physics-based liveness detection truly possible with a single image?. In: **Proceedings of 2010 IEEE International Symposium on Circuits and Systems**. IEEE, 2010. p. 3425-3428.

BALAKRISHNAMA, Suresh; GANAPATHIRAJU, Aravind. Linear discriminant analysis-a brief tutorial. **Institute for Signal and information Processing**, v. 18, n. 1998, p. 1-8, 1998.

BERTINETTO, Luca et al. Fully-convolutional siamese networks for object tracking. In: **Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II 14**. Springer International Publishing, 2016. p. 850-865.

BIOMETRICS INSTITUTE. **Types of Biometrics**. Disponível em: <https://www.biometricsinstitute.org/what-is-biometrics/types-of-biometrics/>. Acesso em: 3 mar. 2020.

BKAV'S CORPORATION. **Bkav's new mask beats Face ID in "twin way": Severity level raised, do not use Face ID in business transactions**. Disponível em: <https://www.bkav.com/top-news/-/view-content/65202/bkav-s-new-mask-beats-face-id-in-twin-way-severity-level-raised-do-not-use-face-id-in-business-transactions>. Acesso em: 4 fev. 2020.

BOULKENAFET, Zinelabinde et al. OULU-NPU: A mobile face presentation attack database with real-world variations. In: **2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017)**. IEEE, 2017. p. 612-618.

BOULKENAFET, Zinelabidine; KOMULAINEN, Jukka; HADID, Abdenour. Face anti-spoofing based on color texture analysis. In: **2015 IEEE international conference on image processing (ICIP)**. IEEE, 2015. p. 2636-2640.

BOULKENAFET, Zinelabidine; KOMULAINEN, Jukka; HADID, Abdenour. Face spoofing detection using colour texture analysis. **IEEE Transactions on Information Forensics and Security**, v. 11, n. 8, p. 1818-1830, 2016.

BREWSTER, Thomas. We broke into a bunch of android phones with a 3d-printed head. **2018-12-13]. https://www. forbes. com/sites/thomasbrewster/2018/12/13/we-broke-into-a-bunch-of-android-phones-with-a-3d-printed-head/# 510662081330**, 2018. Acesso em: 10 fev. 2020.

BROMLEY, Jane et al. Signature verification using a" siamese" time delay neural network. **Advances in neural information processing systems**, v. 6, 1993.

BRUCE, V.; YOUNG, A. Understanding face recognition. **British Journal of Psychology**, v. 77, n. 3, p. 305–327, ago. 1986.

CANZIANI, Alfredo; PASZKE, Adam; CULURCIELLO, Eugenio. An analysis of deep neural network models for practical applications. **arXiv preprint arXiv:1605.07678**, 2016.

CHAKRABORTY, S.; DAS, D. AN OVERVIEW OF FACE LIVENESS DETECTION. **International Journal on Information Theory (IJIT)**, v. 3, n. 2, 2014.

CHINGOVSKA, Ivana et al. Evaluation methodologies for biometric presentation attack detection. **Handbook of biometric anti-spoofing: Presentation attack detection**, p. 457-480, 2019.

CHINGOVSKA, Ivana; ANJOS, André; MARCEL, Sébastien. On the effectiveness of local binary patterns in face anti-spoofing. In: **2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG)**. IEEE, 2012. p. 1-7.

CHINGOVSKA, Ivana; DOS ANJOS, Andre Rabello; MARCEL, Sebastien. Biometrics evaluation under spoofing attacks. **IEEE transactions on Information Forensics and Security**, v. 9, n. 12, p. 2264-2276, 2014.

DE FREITAS PEREIRA, Tiago et al. LBP− TOP based countermeasure against face spoofing attacks. In: **Computer Vision-ACCV 2012 Workshops: ACCV 2012 International Workshops, Daejeon, Korea, November 5-6, 2012, Revised Selected Papers, Part I 11**. Springer Berlin Heidelberg, 2013. p. 121-132.

DE MARSICO, Maria et al. Moving face spoofing detection via 3D projective invariants. In: **2012 5th IAPR International Conference on Biometrics (ICB)**. IEEE, 2012. p. 73-78.

DE SOUZA, Gustavo Botelho; PAPA, João Paulo; MARANA, Aparecido Nilceu. On the learning of deep local features for robust face spoofing detection. In: **2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. IEEE, 2018. p. 258-265.

DE HAAN, Gerard; JEANNE, Vincent. Robust pulse rate from chrominance-based rPPG. **IEEE Transactions on Biomedical Engineering**, v. 60, n. 10, p. 2878-2886, 2013.

DENNING, Dorothy E. Why I love biometrics: It is 'liveness,'not secrecy, that counts. **Information Security**, 2001.

ERDOGMUS, Nesli; MARCEL, Sebastien. Spoofing face recognition with 3D masks. **IEEE transactions on information forensics and security**, v. 9, n. 7, p. 1084-1097, 2014.

FREIBERG, A.-K. **Presentation attack detection prevents biometric fakes**. Disponível em: <https://www.bioid.com/blog/2019/07/presentation-attack-detection/>. Acesso em: 4 mar. 2020.

FRISCHHOLZ, Robert W.; WERNER, Alexander. Avoiding replay-attacks in a face recognition system using head-pose estimation. In: **2003 IEEE International SOI Conference. Proceedings (Cat. No. 03CH37443)**. IEEE, 2003. p. 234-235.

FUNG, Brian; GARCIA, Ahiza. Facebook has shut down 5.4 billion fake accounts this year. **CNN Business**, 2019.

GARG, Shilpa et al. DeBNet: multilayer deep network for liveness detection in face recognition system. In: **2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)**. IEEE, 2020. p. 1136-1141.

KAUR GILL, J. Automatic Log Analysis using Deep Learning and AI. **Accès https://www. xenonstack. com/blog/log-analytics-deep-machine-learning**, 2018. Acesso em: 29 fev. 2020.

GUO, Qing et al. Learning dynamic siamese network for visual object tracking. In: **Proceedings of the IEEE international conference on computer vision**. 2017. p. 1763-1771.

HADSELL, Raia; CHOPRA, Sumit; LECUN, Yann. Dimensionality reduction by learning an invariant mapping. In: **2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)**. IEEE, 2006. p. 1735-1742.

HAO, Huiling; PEI, Mingtao; ZHAO, Meng. Face liveness detection based on client identity using siamese network. In: **Pattern Recognition and Computer Vision: Second Chinese Conference, PRCV 2019, Xi'an, China, November 8–11, 2019, Proceedings, Part I 2**. Springer International Publishing, 2019. p. 172-180.

HAYKIN, Simon. **Redes neurais: princípios e prática**. Bookman Editora, 2001.

HOSKINS, Josiah Collier; HIMMELBLAU, David M. Artificial neural network models of knowledge representation in chemical engineering. **Computers & Chemical Engineering**, v. 12, n. 9-10, p. 881-890, 1988.

HUA, Yuming; GUO, Junhai; ZHAO, Hua. Deep belief networks and deep learning. In: **Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things**. IEEE, 2015. p. 1-4.

ISO/IEC JTC1 SC37 Biometrics (2016) ISO/IEC 30107-1. Information technology - biometric presentation attack detection - part 1: framework. International Organization for Standardization

JAIN, Anil K.; MAO, Jianchang; MOHIUDDIN, K. Moidin. Artificial neural networks: A tutorial. **Computer**, v. 29, n. 3, p. 31-44, 1996.

JAIN, Anil K.; ROSS, Arun; PANKANTI, Sharath. Biometrics: a tool for information security. **IEEE transactions on information forensics and security**, v. 1, n. 2, p. 125-143, 2006.

JAIN, Anil K.; FLYNN, Patrick; ROSS, Arun A. (Ed.). **Handbook of biometrics**. Springer Science & Business Media, 2007.

JAKKULA, Vikramaditya. Tutorial on support vector machine (svm). **School of EECS, Washington State University**, v. 37, n. 2.5, p. 3, 2006.

JOURABLOO, Amin; LIU, Xiaoming. Pose-invariant 3D face alignment. In: **Proceedings of the IEEE international conference on computer vision**. 2015. p. 3694-3702.

JOURABLOO, Amin; LIU, Xiaoming. Large-pose face alignment via CNN-based dense 3D model fitting. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2016. p. 4188-4196.

JOURABLOO, Amin; LIU, Xiaoming. Pose-invariant face alignment via CNN-based dense 3D model fitting. **International Journal of Computer Vision**, v. 124, p. 187-203, 2017.

KANNALA, Juho; RAHTU, Esa. Bsif: Binarized statistical image features. In: **Proceedings of the 21st international conference on pattern recognition (ICPR2012)**. IEEE, 2012. p. 1363-1366.

KARLIK, B.; OLGAC, A. V. Performance analysis of various activation functions in generalized MLP architectures of neural networks. **International Journal of Artificial Intelligence and Expert Systems**, v. 1, n. 4, p. 111–122, 2011.

KHAN, Asifullah et al. A survey of the recent architectures of deep convolutional neural networks. **Artificial intelligence review**, v. 53, p. 5455-5516, 2020.

KHANDELWAL, R. **Support Vector Machines - Data Driven Investor**. Disponível em: <https://medium.com/datadriveninvestor/support-vector-machines-ae0ff2375479>. Acesso em: 27 fev. 2020.

KIM, Gahyun et al. Face liveness detection based on texture and frequency analyses. In: **2012 5th IAPR international conference on biometrics (ICB)**. IEEE, 2012. p. 67-72.

KIM, Wonjun; SUH, Sungjoo; HAN, Jae-Joon. Face liveness detection from a single image via diffusion speed model. **IEEE transactions on Image processing**, v. 24, n. 8, p. 2456-2465, 2015.

KOCH, Gregory et al. Siamese neural networks for one-shot image recognition. In: **ICML deep learning workshop**. 2015.

KOKKINOS, Iasonas; YUILLE, Alan. Scale invariance without scale selection. In: **2008 IEEE conference on computer vision and pattern recognition**. IEEE, 2008. p. 1-8.

KOLLREIDER, Klaus et al. Real-time face detection and motion analysis with application in “liveness” assessment. **IEEE Transactions on Information Forensics and Security**, v. 2, n. 3, p. 548-558, 2007.

KOLLREIDER, Klaus; FRONTHALER, Hartwig; BIGUN, Josef. Non-intrusive liveness detection by face images. **Image and Vision Computing**, v. 27, n. 3, p. 233-244, 2009.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. **Communications of the ACM**, v. 60, n. 6, p. 84-90, 2017.

WENG, L. **Meta-Learning: Learning to Learn Fast**. Disponível em: <https://lilianweng.github.io/posts/2018-11-30-meta-learning/>.

LAWRENCE, Steve et al. Face recognition: A convolutional neural-network approach. **IEEE transactions on neural networks**, v. 8, n. 1, p. 98-113, 1997.

LeCun Y, Bengio Y, Hinton G. Deep learning. **Nature**. 2015 May 28;521(7553):436-44. doi: 10.1038/nature14539. PMID: 26017442.

LI, Jiangwei et al. Live face detection based on the analysis of fourier spectra. In: **Biometric technology for human identification**. SPIE, 2004. p. 296-303.

LI, Lei et al. An original face anti-spoofing approach using partial convolutional neural network. In: **2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)**. IEEE, 2016. p. 1-6.

LI, Lei et al. 3D face mask presentation attack detection based on intrinsic image analysis. **Iet Biometrics**, v. 9, n. 3, p. 100-108, 2020.

LIU, Yaojie et al. Deep tree learning for zero-shot face anti-spoofing. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. 2019. p. 4680-4689.

LIU, Yaojie; JOURABLOO, Amin; LIU, Xiaoming. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2018. p. 389-398.

LUKAC, Rastislav; PLATANIOTIS, Konstantinos N. (Ed.). **Color image processing: methods and applications**. CRC press, 2018.

MÄÄTTÄ, Jukka; HADID, Abdenour; PIETIKÄINEN, Matti. Face spoofing detection from single images using micro-texture analysis. In: **2011 international joint conference on Biometrics (IJCB)**. IEEE, 2011. p. 1-7.

MASI, Iacopo et al. Deep face recognition: A survey. In: **2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)**. IEEE, 2018. p. 471-478.

MAULIK, U.; CHAKRABORTY, D. Remote Sensing Image Classification: A survey of support-vector-machine-based advanced techniques. **IEEE Geoscience and Remote Sensing Magazine**, v. 5, n. 1, p. 33–52, 2017.

MCAFEE, A. et al. Big data: the management revolution. **Harvard business review**, v. 90, n. 10, p. 60–68, 2012.

MELEKHOV, Iaroslav; KANNALA, Juho; RAHTU, Esa. Siamese network features for image matching. In: **2016 23rd international conference on pattern recognition (ICPR)**. IEEE, 2016. p. 378-383.

MITCHELL, T. M. Machine learning and data mining. **Communications of the ACM**, v. 42, n. 11, p. 30–36, 1999.

MOHAMED, Abdelrahamn Ashraf et al. Face liveness detection using a sequential CNN technique. In: **2021 IEEE 11th annual computing and communication workshop and conference (CCWC)**. IEEE, 2021. p. 1483-1488.

MOINDROT, O. **Triplet Loss and Online Triplet Mining in TensorFlow**. Disponível em: <https://omoindrot.github.io/triplet-loss>.

MUKUNDAN, Ramakrishnan. Local Tchebichef moments for texture analysis. 2014.

NOBLE, William S. What is a support vector machine?. **Nature biotechnology**, v. 24, n. 12, p. 1565-1567, 2006.

NOSAKA, Ryusuke; OHKAWA, Yasuhiro; FUKUI, Kazuhiro. Feature extraction based on co-occurrence of adjacent local binary patterns. In: **Advances in Image and Video Technology: 5th Pacific Rim Symposium, PSIVT 2011, Gwangju, South Korea, November 20-23, 2011, Proceedings, Part II 5**. Springer Berlin Heidelberg, 2012. p. 82-91.

OJALA, T.; PIETIKAINEN, M.; MAENPAA, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 24, n. 7, p. 971–987, 2002.

OJANSIVU, Ville; HEIKKILA, Janne. Blur insensitive texture classification using local phase quantization. **Lecture Notes in Computer Science**, v. 5099, p. 236-243, 2008.

PALA, Federico; BHANU, Bir. On the accuracy and robustness of deep triplet embedding for fingerprint liveness detection. In: **2017 IEEE International Conference on Image Processing (ICIP)**. IEEE, 2017. p. 116-120.

PALA, Federico; BHANU, Bir. Iris liveness detection by relative distance comparisons. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops**. 2017. p. 162-169.

PAN, Gang et al. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In: **2007 IEEE 11th international conference on computer vision**. IEEE, 2007. p. 1-8.

PAN, Gang; WU, Zhaohui; SUN, Lin. Liveness detection for face recognition. **Recent advances in face recognition**, p. 109-124, 2008.

PARKHI, Omkar M.; VEDALDI, Andrea; ZISSERMAN, Andrew. Deep face recognition. 2015.

PATEL, Keyurkumar; HAN, Hu; JAIN, Anil K. Secure face unlock: Spoof detection on smartphones. **IEEE transactions on information forensics and security**, v. 11, n. 10, p. 2268-2283, 2016.

QIAN, Yichen; DENG, Weihong; HU, Jiani. Unsupervised face normalization with extreme pose and expression in the wild. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. 2019. p. 9851-9858.

RAMACHANDRA, Raghavendra; BUSCH, Christoph. Presentation attack detection methods for face recognition systems: A comprehensive survey. **ACM Computing Surveys (CSUR)**, v. 50, n. 1, p. 1-37, 2017.

RAWAT, Waseem; WANG, Zenghui. Deep convolutional neural networks for image classification: A comprehensive review. **Neural computation**, v. 29, n. 9, p. 2352-2449, 2017.

R.C. LACHER; NARAYAN, S.; CYBENKO, G. **Can neural network computers learn from experience, and if so, could they ever become what we would call "smart"? And could two different neural networks teach each other what they know, thereby making each other a better network? - Scientific American**. Disponível em: <https://www.scientificamerican.com/article/can-neural-network-comput/>. Acesso em: 4 mar. 2020.

SAMUEL, A. L. Some Studies in Machine Learning Using the Game of Checkers. **IBM Journal of Research and Development**, v. 3, n. 3, p. 210–229, 1959.

SATORRA, A.; BENTLER, P. M. A scaled difference chi-square test statistic for moment structure analysis. **Psychometrika**, v. 66, n. 4, p. 507–514, 2001.

SCHROFF, Florian; KALENICHENKO, Dmitry; PHILBIN, James. Facenet: A unified embedding for face recognition and clustering. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2015. p. 815-823.

SMITH, Daniel F.; WILIEM, Arnold; LOVELL, Brian C. Face recognition on consumer devices: Reflections on replay attacks. **IEEE Transactions on Information Forensics and Security**, v. 10, n. 4, p. 736-745, 2015.

SRISKANDARAJA, Kaavya; SETHU, Vidhyasaharan; AMBIKAIRAJAH, Eliathamby. Deep siamese architecture based replay detection for secure voice biometric. In: **Interspeech**. 2018. p. 671-675.

SUBASIC, M. et al. Face image validation system. In: **ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.** IEEE, 2005. p. 30-33.

SUFYANU, Zahraddeen et al. Feature extraction methods for face recognition. **Int. J. Applied Eng. Research (IRAER)**, v. 5, p. 5658-5668, 2016.

SUN, Yi et al. Deep learning face representation by joint identification-verification. **Advances in neural information processing systems**, v. 27, 2014.

SZEGEDY, Christian et al. Rethinking the inception architecture for computer vision. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2016. p. 2818-2826.

TAN, Xiaoyang et al. Face Liveness Detection from a Single Image with Sparse Low Rank Bilinear Discriminative Model. **ECCV (6)**, v. 6316, p. 504-517, 2010.

TANG, Di et al. Face flashing: a secure liveness detection protocol based on light reflections. **arXiv preprint arXiv:1801.01949**, 2018.

TANG, Yichuan. Deep learning using linear support vector machines. **arXiv preprint arXiv:1306.0239**, 2013.

UDYAVAR, N. **A Beginner's Guide to Neural Networks: Part Two**. Disponível em: <https://towardsdatascience.com/a-beginners-guide-to-neural-networks-part-two-bd503514c71a>.

UNNIKRISHNAN, Shilpa; ESHACK, Ansiya. Face spoof detection using image distortion analysis and image quality assessment. In: **2016 International Conference on Emerging Technological Trends (ICETT)**. IEEE, 2016. p. 1-5.

VINAY, A. et al. Cloud based big data analytics framework for face recognition in social networks using machine learning. **Procedia Computer Science**, v. 50, p. 623-630, 2015.

WU, H. et al. A CNN-SVM combined model for pattern recognition of knee motion using mechanomyography signals. **Journal of Electromyography and Kinesiology**, v. 42, p. 136–142, 2018.

WU, Z.; SHEN, C.; VAN DEN HENGEL, A. Wider or deeper: Revisiting the resnet model for visual recognition. **Pattern Recognition**, v. 90, p. 119–133, 2019.

YAN, Junjie et al. Face liveness detection by exploring multiple scenic clues. In: **2012 12th International Conference on Control Automation Robotics & Vision (ICARCV)**. IEEE, 2012. p. 188-193.

YANG, Jianwei et al. Face liveness detection with component dependent descriptor. In: **2013 International Conference on Biometrics (ICB)**. IEEE, 2013. p. 1-6.

YANG, Jianwei; LEI, Zhen; LI, Stan Z. Learn convolutional neural network for face anti-spoofing. **arXiv preprint arXiv:1408.5601**, 2014.

YI, Jizheng et al. Illumination normalization of face image based on illuminant direction estimation and improved retinex. **PloS one**, v. 10, n. 4, p. e0122200, 2015.

YUEN, P. C. **3D Mask Face Anti-Spoofing**. Disponível em: <http://www.comp.hkbu.edu.hk/v1/proj/sre/2019/pc1/>. Acesso em: 3 fev. 2020.

ZAGHETTO, Cauê et al. Projeto e implementação de uma rede neural artificial para detecção do mal-posicionamento rotacional de dedos em dispositivos de captura de

impressões digitais multivista sem toque. In: **Anais do XI Simpósio Brasileiro de Sistemas de Informação**. SBC, 2015. p. 211-218.

ZAVYALOVA, V. **Hiding from artificial intelligence in the age of total surveillance - Russia Beyond**. Disponível em: <https://www.rbth.com/science_and_tech/2017/07/22/hiding-from-artificial-intelligence-in-the-age-of-total-surveillance_808692>. Acesso em: 4 fev. 2020.

ZHANG, Hao et al. Poseidon: An Efficient Communication Architecture for Distributed Deep Learning on GPU Clusters. In: **USENIX Annual Technical Conference**. 2017. p. 1.2.

ZHANG, Yuanhan et al. Celeba-spoof: Large-scale face anti-spoofing dataset with rich annotations. In: **Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16**. Springer International Publishing, 2020. p. 70-85.

ZHANG, Zhiwei et al. A face antispoofing database with diverse attacks. In: **2012 5th IAPR international conference on Biometrics (ICB)**. IEEE, 2012. p. 26-31.

ZHU, Xiangxin; RAMANAN, Deva. Face detection, pose estimation, and landmark localization in the wild. In: **2012 IEEE conference on computer vision and pattern recognition**. IEEE, 2012. p. 2879-2886.