UNIVERSIDADE ESTADUAL DO PIAUÍ - UESPI
CAMPUS PROF. ALEXANDRE ALVES DE OLIVEIRA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

LUCAS ROCHA DA COSTA

REDES NEURAIS ARTIFICIAIS NO PROCESSO DE IDENTIFICAÇÃO EM UM SISTEMA BIOMÉTRICO DE BAIXO CUSTO COM HARDWARE LIVRE

PARNAÍBA
2018

LUCAS ROCHA DA COSTA

REDES NEURAIS ARTIFICIAIS NO PROCESSO DE IDENTIFICAÇÃO EM UM SISTEMA BIOMÉTRICO DE BAIXO CUSTO COM HARDWARE LIVRE

Trabalho de Conclusão de Curso submetido ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual do Piauí, Campus Prof. Alexandre Alves de Oliveira, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Luciano Kelvin da Silva

PARNAÍBA
2018

C837r Costa, Lucas Rocha da.

Redes neurais artificiais no processo de identificação em um sistema biométrico de baixo custo com *hardware* livre / Lucas Rocha da Costa. - 2018.
37 f.

Monografia (graduação) – Universidade Estadual do Piauí - UESPI, Curso de Bacharelado em Ciência da Computação, *Campus* Profº. Alexandre Alves de Oliveira, Parnaíba-PI, 2018.
"Orientador: Prof. Me. Luciano Kelvin da Silva ".

1. Inteligência artificial. 2. Redes neurais artificiais. 3. Sistema de identificação biométrico.
I. Título.

CDD: 006

Ficha elaborada pelo Serviço de Catalogação da Universidade Estadual do Piauí – UESPI
Aureste de Sousa Lima (Bibliotecário) CRB 3/1215

LUCAS ROCHA DA COSTA

REDES NEURAIS ARTIFICIAIS NO PROCESSO DE IDENTIFICAÇÃO EM UM SISTEMA BIOMÉTRICO DE BAIXO CUSTO COM HARDWARE LIVRE

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual do Piauí – UESPI, Campus Prof. Alexandre Alves de Oliveira, como parte das exigências da disciplina de Estágio Supervisionado, requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Luciano Kelvin da Silva

Monografia Aprovada em: **27 de julho de 2018**.

BANCA EXAMINADORA:

Prof. Me. Luciano Kelvin da Silva
Orientador - UESPI

Prof. Me. Rodrigo Augusto Rocha Souza Baluz
Avaliador - UESPI

Prof. Me. Marcos Aurélio Ayres da Silva
Avaliador - UFPI

Este trabalho é dedicado às crianças adultas, que quando pequenas sonharam em se tornar cientistas.

AGRADECIMENTOS

A minha gratidão infinda em primeiro lugar a Deus, que independentemente da situação em que me encontrei, Ele sempre esteve comigo. Fonte inesgotável de amor, força, vida e paz. Que com graça e misericórdia permitiu que eu chegasse até aqui.

Aos meus pais Francisco Caldas da Costa e Joana Maria Rocha da Costa, sustentáculos incontestes da minha formação moral, civil e ética. Obrigado por todo o esforço e dedicação em proporcionar-me a chance de conquistar e realizar esse sonho que nem de longe é só meu. Grato pelas vossas orações que me sustentaram até que ele se tornasse realidade.

Aos meus irmãos Loide, Líria e Luan por todo o apoio e incentivo, por acreditarem que era possível vencer e por estarem sempre ao meu lado em todos os momentos, me dando forças e tendo as melhores palavras de conforto. Não posso esquecer dos meus irmãos postiços (filho e sobrinho também! risos), que são como parte de mim. Amo todos vocês!

Ao meu tio João Ferreira dos Santos, por ter me acolhido em sua casa durante esses anos, sendo como um pai pra mim e até mesmo aos 101 anos sempre demonstrando preocupação e interesse em meu bem-estar.

Às minhas primas (carinhosamente tratadas como tias), tia Ady e tia Anete, que estiveram presentes durante toda a minha caminhada, agindo e cuidando de mim como mães. Serei sempre grato.

Ao meu amigo Kalil Vieira, que por muitas vezes guardou a própria dor no bolso pra cuidar da minha. Obrigado por sempre me fazer ver, perceber e crer que eu sou capaz de chegar onde quiser. Essa vitória é nossa!

Aos meus irmãos em Cristo da Assembleia de Deus, tanto em Magalhães de Almeida MA, na pessoa do meu Pr. Melkes Oliveira, como em Parnaíba PI, na pessoa do Pr. José Gonçalves. Obrigado por contribuírem com meu crescimento espiritual e por me ajudarem em oração.

Aos amigos e colegas de turma André Igor Lopes (Dedé), Daniel Araújo (Pan), Gyrrece Oliveira (Gy) e José Maria Jr. (Cobra), o meu Agradecimento mais que especial. Manos, sem vocês e vossa colaboração eu não teria terminado esse curso. Sou eternamente grato por tudo o que vocês fizeram por mim quando eu mais precisei. No momento mais difícil de todos esses anos, quando eu estava já sem forças pra continuar, lá estavam vocês. Obrigado mesmo.

E antes que acabe o espaço nessa página, deixo o meu super obrigado à Universidade Estadual do Piauí, aos funcionários, a todo o corpo docente e coordenação do curso de Ciência da Computação, ao meu orientador Prof. Luciano Kelvin por todo o apoio, ajuda e paciência. À Melhor Turma de Todos os Tempos (Computaria e Pacote de Maldades). Obrigado por comporem essa parte indelével da história da minha vida! Muito obrigado!

“Irei à sua frente, e tornarei planos os montes; quebrarei os portões de bronze e destruirei as trancas de ferro. Eu lhe darei os tesouros escondidos na escuridão, sim, riquezas secretas. Farei isso para que saiba que eu sou o SENHOR, o Deus de Israel, que chama você pelo nome.”

(Bíblia Sagrada, Isaías 45.2,3)

RESUMO

A biometria vem sendo aplicada como forma de identificação há muito tempo. Ela é utilizada em muitos casos quando há a necessidade de se identificar e autenticar usuários, principalmente por possuir características que proporcionam um nível maior de segurança. A impressão digital é o tipo mais comum de biometria utilizado, devido à sua facilidade de coleta e grande aceitação pública. Por esta razão os Sistemas Biométricos baseados na Impressão Digital tem se tornado tão populares e comuns no dia a dia, estando presentes inclusive nos *smartphones* atuais. Contudo, há a probabilidade desses Sistemas Biométricos identificarem incorretamente seus usuários, o que pode causar brechas de segurança, afetando o seu nível de confiabilidade. Com isso, buscaram-se formas de melhorar o acerto desses sistemas. É perceptível que a Inteligência Artificial (IA) tem demonstrado uma capacidade notável de melhorar processos em diversas áreas. A IA dispõe de muitas técnicas e algoritmos que são capazes de desenvolver um aprendizado em cima de dados de treinamento passados a eles. Esses algoritmos são capazes de assimilar um conhecimento sobre a maneira de classificar dados semelhantes. Por essa razão, foram escolhidas as Redes Neurais Artificiais (RNA) para terem o seu desempenho no processo de identificação biométrica avaliado e fosse feita uma verificação de como estas se comportam em relação aos erros de classificação. Utilizamos bases de dados de imagens de Impressões Digitais para compor os conjuntos de dados (*datasets*) que foram usados para testar a eficiência da RNA. Para gerar os *datasets*, foi necessário aplicar técnicas de Tratamento de Imagem, onde fizemos uso da biblioteca OpenCV para implementar as etapas de Equalização, Binarização e Afinamento. A extração das informações relevantes das impressões digitais (minúcias) foi realizada através do algoritmo *Crossing Number* que identificou e marcou as minúcias do tipo Bifurcação e Final de Crista. Ao fim do processo de Extração, foram filtradas as minúcias falsas e gerado o *Template* que representa cada Impressão Digital, sendo possível então montar os datasets que foram testados na RNA com auxílio da ferramenta WEKA. Assim, para um conjunto de dados composto de 80 instâncias, sendo 10 indivíduos e 8 amostras da impressão digital de cada um deles, obtivemos o melhor resultado por meio da técnica *Percentage Split*, que divide o dataset em um conjunto de treino e um conjunto de teste segundo uma porcentagem definida (60% para treino e 40% para teste, nesse caso). Desta forma foi possível reavaliar o modelo gerado pelo conjunto de treinamento utilizando o conjunto de teste. Assim, tivemos como resultado uma taxa de acerto de 97,5% e uma taxa de Falso Positivo de apenas 0,3%.

Palavras-chave: Biometria. Segurança. Redes Neurais Artificiais.

ABSTRACT

Biometrics has been applied as a form of identification for a long time. It is used in many cases when there is a need to identify and authenticate users, mainly because it has features that provide a higher level of security. Fingerprint is the most common type of biometrics used because of its ease of collection and wide public acceptance. For this reason, Biometric Systems based on Fingerprint have become so popular and common day to day, being present even in current smartphones. However, these Biometric Systems are likely to misidentify their users, which can cause security breaches, affecting their level of reliability. For this reason, were sought ways to improve the accuracy of these systems. It is noticeable that Artificial Intelligence (AI) has demonstrated a remarkable ability to improve processes in several areas. The AI has many techniques and algorithms that can develop a learning over training data passed to them. These algorithms can assimilate knowledge about how to classify similar data. For this reason, the Artificial Neural Networks (ANN) were chosen to have their performance evaluated in the biometric identification process and to verify how they behave in relation to classification errors. We used Fingerprint image databases to compose datasets that were used to test ANN efficiency. To generate the datasets, it was necessary to apply Image Processing techniques, where we used the OpenCV library to implement the Equalization, Binarization and Thinning steps. The extraction of the relevant information of the fingerprints (minutiae) was performed through the Crossing Number algorithm, which identified and marked the Bifurcation and Final Ridge minutiae. At the end of the Extraction process, the false minutiae were filtered and the Template that represents each Fingerprint was generated, being possible to assemble the datasets that were tested in the ANN with the help of the WEKA tool. Thus, for a Dataset composed of 80 instances, 10 individuals and 8 samples of the fingerprint of each of them, we obtained the best result through the technique Percentage Split, which divides the dataset into a training set and a test set following a defined percentage (60% for training and 40% for test, in this case). In this way it was possible to re-evaluate the model generated by the training set using the test set. Thus, we have resulted in a hit rate of 97.5% and a false positive rate of only 0.3%.

Keywords: Biometrics. Security. Artificial Neural Network.

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO

A biometria sempre foi utilizada como forma de identificação humana. Os traços do rosto, a voz e até o caminhar podem ser formas de se reconhecer um indivíduo. Com a evolução da tecnologia e a necessidade constante de estar se conectando a várias contas, o uso da biometria como uma ferramenta de autenticação foi se tornando imprescindível. A praticidade, a simplicidade e a facilidade de utilização das quais ela dispõe, fazem da biometria uma forte candidata a substituir as senhas atuais (MAZI; JÚNIOR, 2009).

A segurança é um fator indispensável quando se fala de autenticação, já que há a possibilidade de um usuário tentar se passar por outro. Nesse quesito a biometria traz um nível de confiabilidade inegável, uma vez que as chances de se fraudar uma característica biométrica ou de se *hackear* como acontece com senhas comuns, embora existam, são muito pequenas.

Os sistemas de identificação baseados na biometria da Impressão Digital (ID) já são uma realidade no cotidiano das pessoas e estão tornando-se cada vez mais presentes. Terminais de auto-atendimento de bancos, sistemas de ponto e até *smartphones* já possuem essa opção de identificação, usada normalmente para fins de autenticação, segurança e privacidade .

Entretanto, sistemas biométricos (SB) possuem taxas de erro relacionadas à sua capacidade de rejeitar ou aceitar uma tentativa de acesso. A Taxa de Falsa Aceitação (FAR - *False Acceptance Rate*) e a Taxa de Falsa Rejeição (FRR - *False Rejection Rate*) determinam o quão preciso é um SB. A FAR é definida pela probabilidade que o sistema tem de aceitar um usuário impostor, afetando diretamente o nível de segurança do SB, estando associada à Taxa de Falso Positivo (FPR - *False Positive Rate*). Já a FRR consiste da probabilidade do sistema de rejeitar um usuário autêntico, o que afeta a conveniência do SB, estando associada à Taxa de Falso Negativo (FNR - *False Negative Rate*) (COSTA; OBELHEIRO; FRAGA, 2006).

A comunidade biométrica diferencia vários tipos de erros, conforme a localização lógica de sua ocorrência. Os diversos tipos de aplicações biométricas podem ter definições distintas dos erros associados. Consequentemente, há muita terminologia para expressar a precisão de uma aplicação (BOLLE et al., 2013). É bastante claro e aceito que qualquer SB cometerá erros e que o verdadeiro valor associado às diversas taxas de erro não pode ser estabelecido teoricamente, por cálculo, mas somente por estimativas estatísticas dos erros, que são expressos em taxas e porcentagens (COSTA; OBELHEIRO; FRAGA, 2006). Como forma de garantir a segurança e a confiabilidade de um sistema biométrico de baixo custo desenvolvido com hardware livre, buscaram-se maneiras de promover uma redução nessas taxas de erro. Assim, a partir do estudo da temática, percebeu-se que a Inteligência Artificial é dotada de subsídios capazes de contribuir com essas melhorias.

A Inteligência Artificial possui um amplo leque de ferramentas que são nitidamente capazes de proporcionar melhorias em processos dos mais diversos nichos. Na busca por essas

melhorias, chegou-se às Redes Neurais Artificiais (RNA), que são algoritmos baseados na forma de funcionamento do cérebro humano, que são como um contratipo às Redes Neurais Naturais (RNN), capazes de desenvolver um aprendizado a partir de treinamento (HAYKIN, 2007). As RNAs vêm sendo cada vez mais utilizadas, devido sua capacidade de generalização e precisão em classificar instâncias que não fizeram parte do conjunto de treino.

O objetivo que norteia este trabalho é realizar um estudo da eficiência das Redes Neurais Artificiais no processo de identificação biométrica em um Sistema Biométrico de baixo custo desenvolvido com Hardware Livre (Arduino). Para tanto, faz-se necessário fragmentar esse objetivo geral em alguns objetivos específicos, como seguem: (i) Implementar o sistema de aquisição de imagem da ID com o Arduino e o sensor biométrico compatível, possibilitando a criação da base de dados utilizada para testes; (ii) Aplicar técnicas de tratamento de imagem nas amostras adquiridas a fim de melhorar o processo de extração das informações e geração de *Template*; (iii) Preparar os *Datasets* a partir das imagens da base de dados, adaptando-os ao formato compatível com a ferramenta WEKA; (iv) Treinar e testar a RNA usando os conjuntos de dados definidos pra cada fim; (v) Analisar os resultados obtidos pela RNA.

A partir do Sistema de Aquisição de Imagem da Impressão Digital que foi implementado utilizando uma placa Arduino UNO, um sensor biométrico Adafruit e o software SFGDemo, coletaram-se 8 amostras da ID de 10 indivíduos gerando a Base de Dados utilizada para teste, seguindo o formato de base de dados de imagens disponibilizadas pela FVC - *Fingerprint Verification Competition*, da qual também foram utilizadas imagens para montagem de database para efeito de comparação de resultados. As imagens dessas bases de dados precisam passar por um processo de tratamento, que visa melhorar a qualidade das amostras e facilitar o processo de extração das características relevantes da ID (minúcias). Essa etapa de Tratamento foi implementada usando a biblioteca OpenCV, a qual dispõe de funções capazes de realizar as operações de Equalização, Binarização e Afinamento.

A Extração das Minúcias se deu através do Algoritmo *Crossing Number* (CASTRO, 2008) que varre a imagem pixel a pixel em busca de minúcias dos dois tipos comumente utilizados: bifurcação e final de crista. A partir do momento que se tem as informações extraídas da imagem é gerada uma representação dela, conhecida por *Template*. Com os templates criados tornou-se viável a montagem dos conjuntos de dados (*Datasets*) destinados aos testes com a RNA. Para tal, foi utilizada a Ferramenta WEKA, onde se realizaram os experimentos sobre os datasets.

Os experimentos levaram em consideração a quantidade de instâncias e classes dos Datasets, tanto da FVC como do nosso Dataset. O melhor resultado foi obtido pelo Dataset que consistia da *Percentage Split* de 60% do Dataset da FVC 2002 (com 80 instâncias) para treinamento e os 40% para teste, chegando-se a 97,5% de Taxa de Acerto e 0,3% de Taxa de Falso Positivo.

Após esse capítulo introdutório, que expôs uma visão geral deste trabalho, o documento segue organizado da seguinte forma:

Capítulo 2: Fundamentação Teórica - Neste capítulo é feita uma explanação dos fundamentos teóricos que apoiam este trabalho. Trata-se da forma como a Biometria tem estado presente no cotidiano no que tange a identificação pessoal, e como ela tem ajudado a melhorar a segurança em muitos sistemas que necessitam de utilizar autenticação. Também aborda-se os conceitos utilizados dentro da Inteligência Artificial, onde falamos das Redes Neurais Artificiais, suas aplicações e como elas tem sido empregadas no contexto da Identificação em Sistemas Biométricos. Por fim tratamos de Trabalhos Relacionados ao contexto dessa pesquisa, seja na implementação de SBs, na aplicação de Técnicas de Tratamento de Imagem e como estas afetam o resultado final da Extração de Minúcias e principalmente quanto ao uso de técnicas de IA como melhoria no processo de identificação através de Biometria.

Capítulo 3: Métodos e Técnicas - Aqui estão expostos os principais conceitos e definições acerca das ferramentas utilizadas no processo de construção e desenvolvimento deste trabalho, além de abordar os métodos e procedimentos aplicados para a execução da metodologia proposta.

Capítulo 4: Análise e Discussão dos Resultados - Neste capítulo são expostos os resultados alcançados ao fim da realização dos testes pertinentes aos experimentos definidos.

Capítulo 5: Considerações Finais - Aqui expõem-se os principais objetivos alcançados e deixam-se sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os fundamentos teóricos que apoiam este trabalho. Aqui demonstra-se a maneira que a Biometria tem estado presente no cotidiano das pessoas e como ela tem se mostrado essencial quanto à necessidade de autenticação e garantia da segurança computacional, assim como expõe-se a inserção da inteligência artificial nesse processo visando melhorias.

2.1 A Biometria como forma de identificação

No decorrer do processo de desenvolvimento de sistemas computacionais é importante lembrar de um fator primordial: a Segurança da Informação. Uma forma de promover a segurança computacional é criar restrições de acesso apenas às pessoas autorizadas utilizando-se de autenticação. De acordo com Silva e Filho (2017) a autenticação é de fundamental importância no que tange a promoção da Segurança no contexto computacional, pois através dela é possível validar a identidade dos usuários. Enquanto a identificação é a função na qual o usuário alega sua identidade para o sistema, a autenticação é responsável por validar essa alegação de identidade. Apenas após a identificação e autenticação é que o sistema dará (ou não) a autorização de acesso ao usuário.

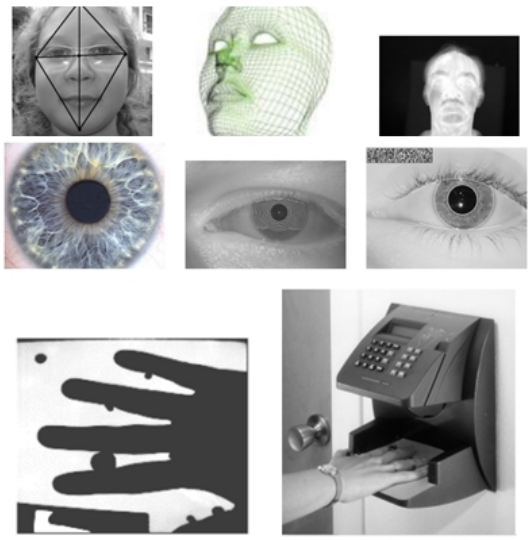
Quando trata-se do conceito de segurança em sistemas computacionais leva-se em conta a manutenção os três pilares essenciais: *confidencialidade*, *integridade* e *disponibilidade*. A confidencialidade garante a divulgação da informação apenas sob autorização; a integridade garante que a informação é verdadeira e íntegra, ou que só seja alterada mediante autorização; e a disponibilidade garante que a informação esteja acessível aos usuários legítimos, quando solicitada (COSTA; OBELHEIRO; FRAGA, 2006). Assim, o uso de identificação/autenticação nesse aspecto de garantir a segurança, pode ser feito de algumas formas: através do algum objeto (crachá ou cartão magnético), de algum conhecimento (senha, código, PIN) ou através de alguma característica física (características biométricas). Uma característica interessante e crucial da biometria em comparação a outros tipos de identificação é que as características biométricas não podem ser perdidas ou esquecidas (FONTANA; MARIM, 2009).

A necessidade de identificação existe desde os primórdios da humanidade. Há séculos as pessoas são reconhecidas naturalmente através da face e da voz (FONTANA; MARIM, 2009). A Biometria [*bio* (vida) + *metria* (medida)] pode ser definida como o estudo das medidas de traços físicos ou comportamentais de um indivíduo ou como o uso automatizado de características fisiológicas/comportamentais para designar e validar entidades (SILVA et al., 2007). Devido possuir características peculiares como: universalidade, unicidade, imutabilidade, classificabilidade, facilidade de coleta (MAZI; JÚNIOR, 2009), a biometria representa uma ótima opção a ser considerada no contexto de identificação de indivíduos.

Dentre os vários tipos de Biometria existentes pode-se citar as comumente usadas em Sistemas Biométricos, como: Impressão Digital (ID), Aparência da Face, Padrão da Íris, Retina, Geometria da Mão, Dinâmica da Assinatura e Padrão de voz (COSTA; OBELHEIRO; FRAGA, 2006). Alguns desses exemplos são mostrados na Figura 1.

A literatura relata muitos outros métodos de identificação biométrica como: o reconhecimento pela orelha, o modo de caminhar, a dinâmica de digitação, o DNA, o odor e o eletrocardiograma. Entretanto, eles não preenchem todos os requisitos importantes necessários em um método efetivo de reconhecimento. Devido ser a técnica mais estudada em anos, o uso da Impressão Digital para identificação possui grande aceitabilidade, é satisfatório quanto à rapidez, precisão e segurança, sendo utilizado para reconhecimento de pessoas há mais de um século pela Forense (BONATO, 2011).

Figura 1 – Exemplos de Biometria - Face, Íris e Geometria da Mão



Fonte: Costa, Obelheiro e Fraga (2006) - Adaptado pelo Autor

As impressões digitais são formadas por sulcos e cristas presentes na pele das mãos e dos pés. O início dos estudos envolvendo as características das linhas da impressão digital se deu por volta de 1667. Aos resultados desses estudos deu-se o nome de Datiloscopia (MAZI; JÚNIOR, 2009). Os chineses foram o primeiro povo a fazer uso da impressão digital em processos de divórcio no século VII e em casos da esfera criminal no século XIV. A Datiloscopia [*daktylos* (dedos) + *skopein* (examinar)] passou a ser aplicada à identificação quando Henry Faulds percebeu que as IDs eram únicas para cada indivíduo e sugeriu em 1880 que as impressões digitais encontradas em cenas de crime fossem utilizadas para identificação de criminosos (COSTA, 2003).

Dessa forma, percebe-se que há muito tempo as impressões digitais (ID) vêm sendo utilizadas para fins de identificação de indivíduos. A unicidade presente nos padrões formados pelas linhas da ID permitem que alguém específico seja identificado (FARIA, 2005). A ID é

uma característica muito popular devido ser largamente utilizada em Sistemas Biométricos, principalmente por possuir simplicidade e praticidade inegáveis, além de um custo baixo.

As IDs possuem detalhes significativos conhecidos como minúcias. Esses detalhes são formados por interrupções ou bifurcações nas cristas (linhas) da ID. É através das minúcias que torna-se possível o uso das IDs na identificação pessoal, já que são elas que determinam a unicidade da pessoa (MAZI; JÚNIOR, 2009). Embora existam vários tipos de minúcias como delta, crista curta, ilha, lago, espora e cruzamento, os detalhes do tipo terminação de linha (crista final) e bifurcação de linha são os mais comumente usados em sistemas biométricos baseados na ID (COSTA; OBELHEIRO; FRAGA, 2006). As figuras 2 e 3 demonstram alguns tipos de minúcias encontradas em impressões digitais:

Figura 2 – Exemplos de Minúcias

Fonte: Faria (2005)

Figura 3 – Bifurcações e Terminações de Linha

Fonte: Costa, Obelheiro e Fraga (2006)

Conforme evidenciado por Costa, Obelheiro e Fraga (2006), independente do tipo de biometria utilizado, ele deve estar condicionado a um **sistema biométrico**, o qual possui um modelo conceitual bem simples, que leva em conta o cadastro prévio do usuário (*enrollment*) e registro do seu perfil biométrico (*template*). Num momento posterior de utilização do sistema para identificação de um dado usuário, a característica biométrica é novamente coletada, e são feitas as etapas de processamento da imagem e extração das características importantes, gerando um perfil que é comparado com o perfil já existente, verificando então a similaridade entre os perfis, possibilitando a validação ou rejeição do usuário.

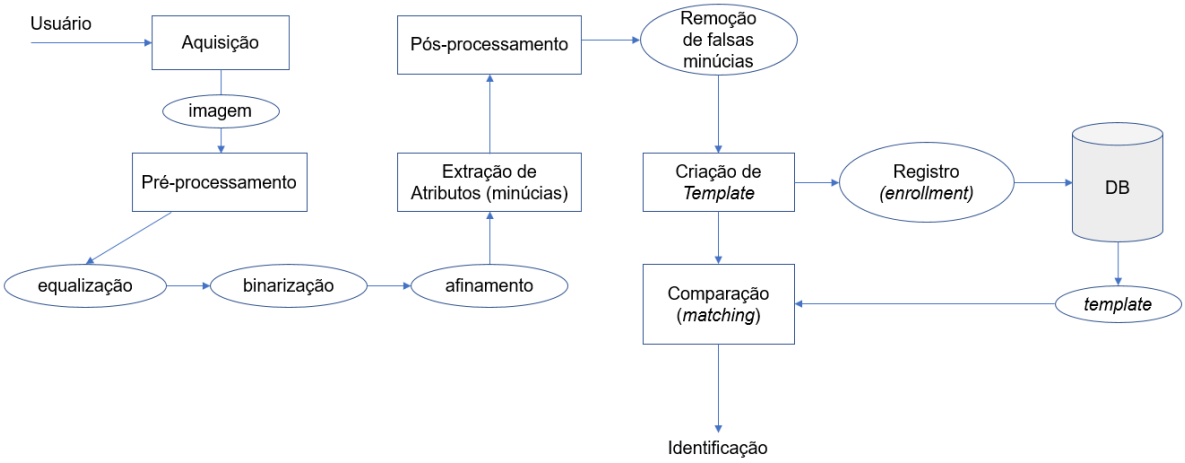
Normalmente em sistemas biométricos, o processo de aquisição é feito por sensores, ou em alguns casos específicos como usando ID ou Geometria da palma da mão a aquisição pode ser feita com tinta e papel, sendo digitalizadas por um scanner posteriormente. Após o processo de aquisição da amostra (*sample*), chega-se ao módulo de pré processamento da imagem, que tem como objetivo melhorar a qualidade da amostra e facilitar a extração dos atributos. Para tanto, são aplicados filtros de equalização para melhorar o contraste e filtros de aguçamento, e então a amostra é binarizada e passa por afinamento ou esqueletização (*thinning*). Despois de ser pré-processada, a amostra segue para ter seus atributos (*minúcias*) extraídos, gerando assim um *Template* que é armazenado e representa digitalmente a característica biométrica, não devendo

ser possível gerar a característica a partir do template. Ainda pode ser necessário passar pela etapa de pós processamento para remoção de possíveis atributos indesejados que porventura tenham sido detectados na etapa de extração (FARIA, 2005) (CASADO; PAIVA, 2008).

Um sistema biométrico deve ser capaz de confirmar uma identidade alegada, assim como também rejeitá-la caso seja necessário. Existem pelo menos duas maneiras diferentes de se realizar esse teste, sendo uma através de verificação (busca 1:1) e a outra através de identificação (busca 1:N). No processo de Verificação, o sistema recebe o sinal biométrico fornecido pelo usuário em conjunto com a identidade alegada a ser validada. Nesse modo de autenticação é realizada uma busca fechada (1:1), onde o sistema compara as informações e verifica a autenticidade, validando ou não o usuário. O resultado obtido pelo sistema quanto à verificação está pautado na resposta à pergunta "O usuário é quem ele diz ser?". A identificação por sua vez, é baseada em uma busca aberta (1:N), onde o usuário informa apenas a sua característica biométrica e cabe unicamente ao sistema varrer toda a base de dados, encontrar e identificar o usuário. A validação resultante da identificação é fundamentada em responder à pergunta: "Quem é o usuário?" (COSTA; OBELHEIRO; FRAGA, 2006).

A Figura 4 representa conceitualmente o esquema de funcionamento de um sistema biométrico:

Figura 4 – Esquema Conceitual de um Sistema Biométrico



Fonte: Prestes (2011) adaptado pelo autor

2.2 Redes Neurais Artificiais na Identificação Biométrica

O cérebro humano é um sistema de processamento de informações altamente complexo, que trabalha de maneira não-linear e paralela. O modo como ele opera e a forma de organização de suas estruturas internas de um jeito que proporciona a capacidade de processar informações (reconhecimento de padrões, percepção e controle motor) muito mais rapidamente que qualquer computador, vêm impulsionando os cientistas a estudarem e desenvolverem formas de imitar o

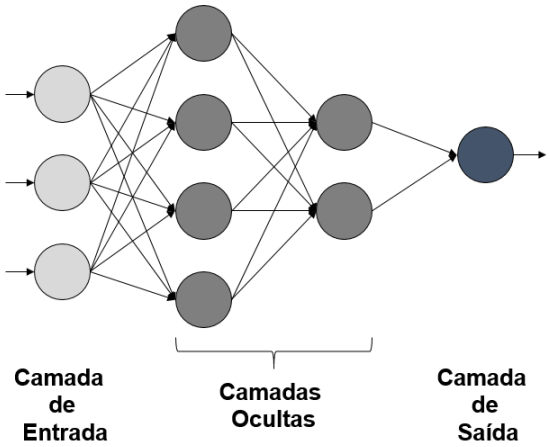
seu funcionamento. Os estudos que envolvem o desenvolvimento das Redes Neurais Artificiais (RNAs) existem desde quando compreendeu-se que o cérebro humano possui um processamento de informações distinto dos computadores digitais convencionais (HAYKIN, 2007).

A partir de então o número de pesquisas desenvolvidas na área de Inteligência Artificial e Redes Neurais Artificiais só cresceu. Atualmente as RNAs atuam não somente na Tecnologia da Informação, mas já é possível, inclusive, ver as RNAs sendo aplicadas na melhoria de processos na área da saúde, como auxiliando no diagnóstico de vários tipos de câncer, também como forma de previsão do número de casos de Dengue, por exemplo, e até mesmo aplicações em Metereologia como previsão de índices pluviométricos.

A literatura aborda vários tipos de RNAs, conforme suas especificações próprias. As mais conhecidas são ADALINE/MADALINE, Backpropagation Perceptron Time-Delay, Recurrent, BAM (Memória Associativa Bidirecional), Hofield e Kohonen (LOESCH; SARI, 1996, p. 46).

A RNA *feedforward*, também conhecida como MLP (*Multi Layer Perceptron*), é uma das mais especificada em livros. Além de ser a mais utilizada entre todas as outras RNAs, cerca de 90%. Possuem grande capacidade de abstração e generalização, o que permite que ela consiga classificar um padrão complexo embora ele não tenha pertencido ao conjunto de treino, e também possui uma robustez que a torna imune a pequenas falhas causadas por ruídos ou distorções de entrada (LOESCH; SARI, 1996, p. 67).

Figura 5 – Representação esquemática da Rede *feedforward*



Fonte: O Autor

As RNAs precisam passar por um processo de treinamento para que sejam capazes de resolver o problema para o qual foram desenvolvidas. O desempenho geral da RNA está fortemente relacionado ao processo de treinamento, que consiste em ajustar a força de conexão entre os neurônios (pesos sinápticos) e configurar os valores de saída em cada caso, fazendo uma verificação do erro em relação ao valor de saída esperado. Dois métodos de treinamento são conhecidos: treinamento supervisionado e treinamento não-supervisionado. O primeiro conta

com o auxílio de um treinador, sendo o método mais comum, no qual os dados de entrada são apresentados para a RNA juntamente com os valores de saída esperados. Após uma iteração há uma comparação dos valores de saída e então acontece um ajuste dos pesos, repetindo-se até que o erro seja minimizado. Já no método de treinamento não-supervisionado, apenas os dados de entrada são apresentados para a RNA, de modo que ela deva aprender sozinha, ajustando os pesos e os valores de saída automaticamente, classificando os dados de entrada com os neurônios (MAZI; JÚNIOR, 2009).

No contexto dos Sistemas Biométricos, a literatura trás grande apoio quanto ao uso de RNAs no processo de identificação. É notável a quantidade de trabalhos desenvolvidos com essa temática com os mais diversos objetivos e especificidades.

Mazi e Júnior (2009) fizeram uso de RNA para identificação biométrica utilizando a ID como caraterística biométrica. O propósito deste sistema desenvolvido por eles se resume na aplicação de redes neurais artificiais para a identificação e reconhecimento biométrico de impressões digitais. O ambiente de trabalho Matlab 7.0 foi utilizado para a elaboração dos algoritmos. O sistema baseou-se na aquisição da imagem através de um leitor biométrico da Microsoft, no tratamento da imagem (etapa de pré-processamento), na captura das minúcias (extração de atributos), na eliminação das falsas minúcias (pós-processamento), no treinamento da rede neural e na comparação dos resultados através de validações.

Uma outra abordagem foi descrita por Fontana e Marim (2009), onde o foco principal do projeto foi utilizar as Redes Neurais Artificiais para autenticação/identificação de indivíduos através da biometria da palma da mão. O processo de captura da imagem foi realizado com um scanner de uma impressora multifuncional HP 1500 series. Após a captura da imagem da palma da mão, eles obtiveram como resultado uma imagem de 256 tons de cinza, escala de 8 bits com 75 ppi de resolução e formato jpeg, a qual após passar pelas etapas de pré-processamento, foi convertida a uma matriz numérica que a representou.

Neves et al. (2017) realizaram a parametrização de uma RNA para verificar o seu desempenho no processo de identificação de impressões digitais. Com o uso da ferramenta WEKA (*Waikato Environment for Knowledge Analysis*), muito popular dentro da Inteligência Artificial e da Aprendizagem de Máquinas, eles compararam os resultados de diversas outras técnicas de IA com os resultados obtidos pela RNA. Essa abordagem comparativa e os parâmetros definidos por eles para validação da proposta são muito importantes e devido aos detalhes expostos na análise dos resultados, possuem uma utilidade inquestionável, porque são norteadores que auxiliam principalmente trabalhos de cunho comparativo.

Muitos outros trabalhos que embora não tratem diretamente do uso de RNA, são listados aqui por estarem claramente associados a outras partes do desenvolvimento desta proposta. Costa (2003) fez uma investigação das plataformas computacionais para identificação de IDs, evidenciando também as etapas de Aquisição, Pré-processamento, Extração de Atributos,

Pós-processamento e Identificação.

Nogueira (2011) utilizou a plataforma de prototipagem Arduino (hardware livre) na captura do sinal biométrico no sistema desenvolvido, com o intuito de representar a imagem da ID com suas respectivas minúcias em uma matrix de LED 8x8. Faria (2005) também propôs um sistema de reconhecimento de digitais para controle de acesso com baixo custo computacional, já que os terminais seriam gerenciados por microcontroladores.

Quanto ao processo de extração de minúcias, Casado e Paiva (2008) se utilizou de técnicas tratamento de imagem para melhorar o processo de extração de atributos de impressões digitais, já Castro (2008) propôs o desenvolvimento dos métodos de identificação pessoal, explorando sistemas baseados em casamento minúcias e implementando os métodos com auxílio da ferramenta computacional MatLab, sendo desenvolvido também um sistema de verificação automática de impressões digitais com uma interface gráfica. Os resultados apresentados foram encorajadores, uma vez que utilizou-se pouca informação para realizar a comparação entre duas imagens.

Prestes (2011) projetou o desenvolvimento de um sistema para o reconhecimento de IDs baseado na utilização do Sistema de Classificação de Henry (Classes de IDs) combinado com os Detalhes de Galton (minúcias). Consistindo em implementação de algoritmos de borramento, aguçamento, binarização e afinamento. Para a detecção de minúcias foi utilizado o método Crossing Number com pós-processamento para eliminação de falsas minúcias.

Por fim, Bonato (2011) utiliza-se alguns métodos que ajudam na melhoria da acurácia do sistema, usufruindo-se técnicas que melhoram a qualidade da imagem da ID para uma extração de minúcias mais exata. Esse trabalho estuda a técnica de processamento de imagem chamada afinamento e consiste em implementar o algoritmo de afinamento e testá-lo, incorporando-o ao software NBIS, um programa criado pelo NIST que faz o reconhecimento biométrico por digitais, e por fim compara o software original com o modificado com o objetivo de aumentar a acurácia do sistema.

3 MÉTODOS E TÉCNICAS

3.1 Aquisição de Imagem da ID com Arduino e Sensor Biométrico

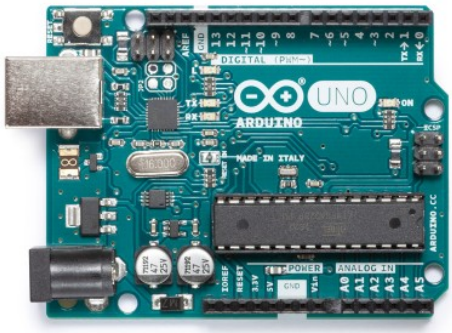
3.1.1 Arduino UNO

O Arduino é uma plataforma *open-source* de prototipagem eletrônica fundamentada no princípio de hardware e software *easy-to-use* (fácil de usar), com a capacidade de ler entradas e transformá-las em uma saída - ligar LEDs, acionar motores, etc (ARDUINO, 2018a). Ele é tipicamente utilizado na robótica, na domótica e nos mais diversos projetos de automação. A versatilidade e a compatibilidade dessa placa com uma gama enorme de sensores, acabam tornando-a uma ótima opção nesse contexto. Por ser de código aberto, é possível desenvolver uma placa própria com características que se adequam a cada tipo de projeto.

As aplicações do Arduino são muito amplas. Vão de projetos simples de robótica educacional a Ferramentas de auxílio a Portadores de Necessidades Especiais.

O Arduino UNO, versão utilizada nesse trabalho, é uma placa baseada no microcontrolador ATmega328P, que possui 14 pinos de entrada / saída digital, 6 entradas analógicas, um cristal de quartzo de 16 MHz, uma conexão USB, um conector de energia, um conector ICSP e um botão de reset. (ARDUINO, 2018b)

Figura 6 – Arduino UNO R3



Fonte: (ARDUINO, 2018b)

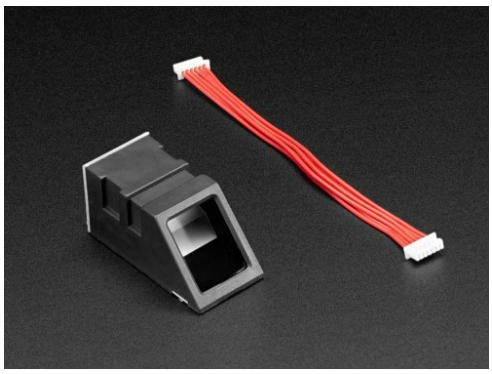
A programação se dá através de *sketchs* (conjunto de intruções codificadas) construídos no Software Arduino (IDE baseada em *Processing*), usando a linguagem de programação Arduino (baseada em *Wiring*) (ARDUINO, 2018a). Assim, enviando o *sketch* ao microcontrolador através de comunicação serial via cabo USB, é possível determinar todos os procedimentos e rotinas que o Arduino deve executar.

3.1.2 Sensor Biométrico

O sensor biométrico utilizado na construção do protótipo do Sistema de Aquisição de Imagem foi um módulo FPM10A (DY50_main_v3), e para estabelecer a comunicação com a Placa Arduino, foi usada a biblioteca *Adafruit Fingerprint Sensor Library*.

A Figura 7 mostra o sensor utilizado na montagem do módulo de aquisição:

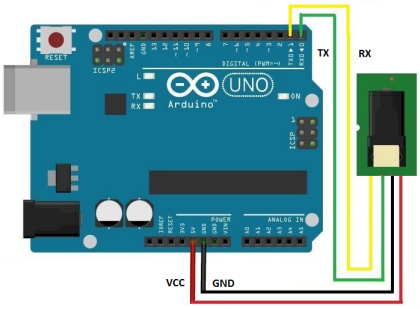
Figura 7 – Sensor Biométrico Adafruit



Fonte: Google Images

Após a comunicação ser estabelecida entre os dispositivos, torna-se possível então trabalhar o *enrollment* (cadastro de IDs) e a etapa de identificação, utilizando as funções disponíveis na biblioteca. Entretanto, como no nosso caso seria necessário a criação de um Banco de Dados de imagens para os testes com a RNA, foi utilizado o Software SFGDemo 2.0, também disponibilizado pela Adafruit, o qual possibilita que as imagens capturadas sejam salvas no computador e assim seja montada a base de dados necessária.

Figura 8 – Esquema de Montagem do Sistema de Aquisição



Fonte: Google Images - Adaptado pelo autor

Figura 9 – Amostra de ID Capturada

Fonte: O Autor

3.2 Tratamento de Imagens com OpenCV

O OpenCV (Open Source Computer Vision) é uma biblioteca *open-source* que implementa várias ferramentas de interpretação de imagens, partindo de operações simples como

aplicação de filtros, até operações complexas, tais como reconhecimento de padrões e análise de movimentos (MARENGONI; STRINGHINI, 2009). Suas funções estão divididas em cinco principais grupos:

- Processamento de imagens;
- Análise estrutural;
- Análise de movimento e rastreamento de objetos;
- Reconhecimento de padrões e
- Calibração de câmera e reconstrução 3D.

Comumente os processos de visão computacional, necessitam envolver o processamento de imagens, já que as imagens de onde queremos extrair alguma informação em alguns casos precisam ser convertidas para um formato diferente ou redimensionada e precisam ainda ser filtradas para remover ruídos provenientes do processo de aquisição (MARENGONI; STRINGHINI, 2009).

Como já evidenciado pela literatura, há a necessidade de as amostras de Impressões Digitais coletadas na fase de Aquisição passarem por um pré-processamento, que visa melhorar a qualidade das imagens a fim de tornar mais fácil e eficiente a extração de seus respectivos atributos. Para tal, utilizamos o OpenCV em sua versão 3.4.1 com a Linguagem C++.

Nossa pesquisa implementou a etapa de pré-processamento das imagens, a qual consiste em Equalização por Histograma, Binarização e Afinamento, utilizando as funções disponíveis no OpenCV. Foi utilizada a IDE CodeBlocks 16.01, no Sistema Operacional Deepin Linux 15.6.

O Algoritmo 1 demonstra a fase de Pré-Processamento, evidenciando as funções do OpenCV, responsáveis por equalizar (ε), binarizar (β) e afinar (α) a amostra de entrada (ι), retornando por fim a imagem resultante (ρ). As variáveis τ e θ armazenam os resultados intermediários do pré-processamento, que são a imagem equalizada e binarizada, respectivamente.

Algoritmo 1: PRÉ-PROCESSAMENTO

Entrada: ι

Saída: ρ

1 **início**

2 $\quad \iota \leftarrow cv.imread(\text{“}caminhodaimagem\text{”})$

3 $\quad \tau \leftarrow cv.\varepsilon(\iota)$

4 $\quad \theta \leftarrow cv.\beta(\tau)$

5 $\quad \rho \leftarrow cv.\alpha(\theta)$

6 **fim**

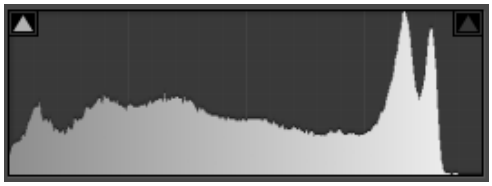
7 **retorna** ρ

Cada processo que compõe a etapa de pré-processamento de imagem, visa melhorar o resultado do procedimento posterior. Desta forma, a Equalização das imagens da ID é feita para que o processo de Binarização seja mais eficiente e consequentemente, o processo de Binarização tem como objetivo melhorar o resultado do Afinamento. Nesse caso, é a imagem resultante do processo de afinamento que segue para o Módulo de Extração de Características.

3.2.1 Equalização de Histograma

O histograma de uma imagem é a representação gráfica da distribuição de intensidade dos *pixels* dessa imagem. Ele quantifica o número de *pixels* para cada valor de intensidade considerado (OPENCV, 2018a). A Figura 10 mostra um Histograma retirado do software de edição de imagens Adobe Photoshop Lightroom Classic CC 2018:

Figura 10 – Exemplo de Histograma

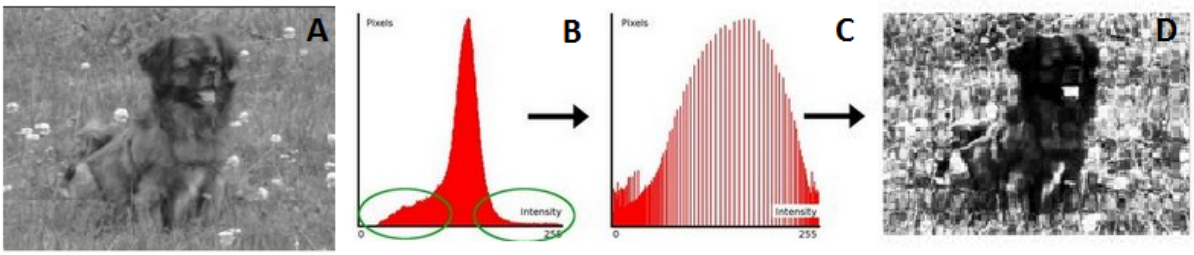


Fonte: O Autor

Os histogramas são ferramentas de processamento de imagens que possuem grande aplicação prática, dentre as quais pode-se citar a melhora da definição de uma imagem, a compressão de imagens, a segmentação de imagens ou ainda a descrição de uma imagem.

A Equalização de Histograma é uma operação bastante comum no contexto do uso de histogramas no tratamento de imagens. Consiste basicamente no ajuste dos valores de intensidade visando melhorar o contraste em uma imagem, mapeando esses valores de um intervalo pequeno (pouco contraste) para um intervalo maior (muito contraste), distribuindo os *pixels* de forma a tentar obter uma distribuição uniforme de intensidades (MARENGONI; STRINGHINI, 2009).

Figura 11 – Equalização de Histograma



Fonte: OpenCV (2018a) - Adaptado pelo autor

Para deixar mais claro, analisando a Figura 11 a partir da subfigura B, pode-se perceber que os pixels parecem agrupados no meio do intervalo de intensidades disponíveis. O que a equalização do histograma faz é esticar esse intervalo. Os círculos indicam as intensidades pouco

povoadas. Ao fim da aplicação de equalização, obtém-se um histograma como mostra a subfigura C, do qual a imagem resultante é mostrada na subfigura D.

A Figura 12 mostra o resultado do processo de Equalização de Histograma de uma amostra coletada pelo nosso Sistema de Aquisição.

Figura 12 – Imagem Coletada *versus* Imagem Equalizada

Fonte: O Autor

A partir desse ponto a imagem passa à etapa de Binarização conforme os passos determinados na definição do Algoritmo 1.

3.2.2 Binarização

A Binarização da imagem é feita através de um procedimento conhecido por Limiarização (*thresholding*), o qual é um processo que segmenta imagens baseando-se na diferença dos tons de cinza que compõem objetos distintos de uma imagem. Tomando por base um limiar estabelecido levando-se em consideração as características dos objetos que se quer isolar, a imagem pode ser segmentada em dois grupos: o grupo de pixels que possuem níveis de cinza abaixo do limiar e o grupo de pixels que possuem níveis de cinza acima do limiar. Em uma imagem limiarizada, um valor fixo é atribuído para todos os pixels de um mesmo grupo.

De um modo mais claro, o processo verifica o valor de cada pixel e compara com o valor do limiar, sendo que, caso o valor de intensidade do pixel seja menor que o limiar, o pixel é então mapeado para um dado valor (0 - preto, por exemplo) e caso o valor de intensidade do pixel seja maior que o limiar, o pixel é mapeado para um outro valor (255 - branco, nesse caso).

Para tal operação, pode-se utilizar a Binarização por limiar adaptativo (*Adaptative Thresholding*) na qual a função *adaptativeThreshold()* analisa cada região da imagem e determina o limiar automaticamente ou a Binarização com limiar fixo, na qual o valor do limiar é definido por parâmetros (*flags*) que são passados para a função *threshold()*. O OpenCV dispõe desses parâmetros os quais definem o valor do limiar conforme o tipo de binarização que será aplicada na imagem.

Para desenvolver esse passo da etapa de pré-processamento, tomamos por base o código implementado por Howse et al. (2015, p. 260), o qual está disponível no github do livro OpenCV

3 Blueprints. Nessa implementação foi utilizada Binarização com limiar fixo com as flags THRESH_BINARY_INV e THRESH_OTSU, onde o algoritmo Otsu é usado para definir o valor ótimo do limiar (OPENCV, 2018b).

A Figura 13 demonstra o resultado obtido nesse procedimento. A imagem binarizada resultante segue para a fase de Afinamento.

Figura 13 – Imagem Equalizada *versus* Imagem Binarizada

Fonte: O Autor

3.2.3 Afinamento

O objetivo principal da fase de Afinamento (*thinning*) é gerar uma imagem que seja formada apenas pelo esqueleto da imagem original, no caso as linhas presentes na imagem da ID são afinadas à largura de um *pixel*, visando melhorias no processo de Extração de Minúcias.

A maioria dos métodos de afinamento de imagem são implementados de forma iterativa e por isso tendem a ser demorados, uma vez que os *pixels* em excesso são eliminados utilizando-se uma varredura em todos os *pixels* da imagem (CASTRO, 2008). Segundo Bonato (2011), existem alguns métodos de se aplicar a técnica de Afinamento em uma imagem, como por exemplo Método de Holt, Método de Stentiford e Método Morfologia Matemática. Cada método tem suas particularidades, porém o objetivo é o mesmo. Geralmente, o algoritmo de *thinning* percorre a imagem analisando os pixels que compõem o objeto, marcando os que serão removidos, para que estes sejam eliminados em um segundo momento. Esses passos se repetem até que não hajam mais *pixels* redundantes, restando assim apenas os *pixels* que fazem parte do esqueleto do objeto, respeitando-se algumas propriedades como: as regiões afinadas precisam ter um pixel de largura, mantendo-se a conectividade e a forma original do objeto.

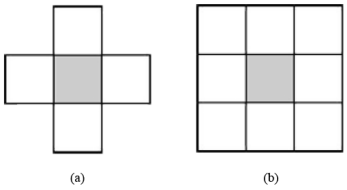
Assim como na fase anterior, nossa implementação tomou por base a implementação do livro OpenCV 3 Blueprints (HOWSE et al., 2015), a qual foi feita conforme o processo de afinamento de Zhan-Suen descrito por Castro (2008), onde o algoritmo de Afinamento recebe uma imagem binarizada e realiza sobre ela as iterações necessárias ao procedimento de *thinning*. A ideia básica é decidir se um determinado pixel vai ser eliminado após verificar a conectividade dos *pixels* de sua vizinhança D8, conforme Figura 14. O método, é realizado em duas sub-iterações, as quais visam a eliminar continuamente os elementos de borda de um objeto até que

este fique com um *pixel* de espessura. A primeira sub-iteração verifica a condição dos *pixels* localizados nas borda superior e direita de um elemento, consequentemente a segunda verifica a condição dos *pixels* localizados na borda inferior e esquerda. Adicionalmente, cada sub-iteração verifica o número de vizinhos não nulos $N(p)$, definido na Equação 3.1, e também a continuidade desses *pixels* vizinhos. A continuidade é obtida ao verificar o número de transições de preto para branco existentes nos vizinhos de borda de um *pixel*, ao ser circulado no sentido anti-horário partindo de $p0$ a $p7$.

$$N(p) = p0 + p1 + p2 + p3 + p4 + p5 + p6 + p7 \tag{3.1}$$

As sub-iterações são responsáveis por marcar os pixels candidatos a eliminação. Após realizar toda a varredura na imagem, esses elementos devem ser removidos da imagem.

Figura 14 – Tipos de Vizinhança de um Pixel (D4 e D8)



Fonte: Castro (2008)

Os elementos marcados para remoção são eliminados após a execução das duas iterações e o método é aplicado à imagem até que não haja mais elementos para serem removidos.

Figura 15 – Imagem Binarizada *versus* Imagem Afinada

Fonte: O Autor

Quando o afinamento termina a imagem resultante passa para a etapa de Extração de Características, onde as minúcias são detectadas e é criada a representação de cada imagem com seus respectivos atributos das mesmas para que posteriormente a RNA identifique as IDs.

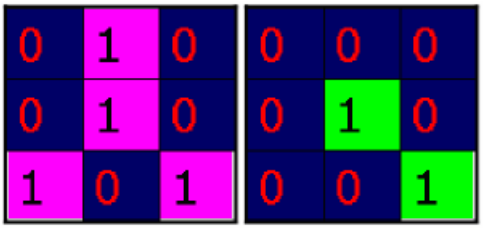
3.3 Extração de Características

Ao fim da etapa de Pré-processamento, chega-se à etapa onde as características relevantes da imagem são extraídas. Essa fase é completamente dependente de todas as etapas anteriores.

As minúcias presentes na formação das linhas da ID são a causa da possibilidade do uso de IDs com a finalidade de identificação pessoal, o que torna essa etapa fundamentalmente importante no contexto dos sistemas biométricos.

Conforme Casado e Paiva (2008), a partir do momento que se tem a imagem da ID com suas cristas papilares afinadas, o processo de extração de minúcias se torna relativamente mais fácil, já que uma vez que as linhas tenham a espessura de um *pixel* é possível fazer uma varredura linha a linha da imagem a fim de encontrar as minúcias usando uma máscara 3x3, detectando terminações e bifurcações nas cristas papilares, conforme mostra a Figura 16. Por exemplo, se o pixel central tem valor 1 (um) e também mais três vizinhos possuírem valor um, então este local será marcado como uma minúcia do tipo bifurcação. Agora se o pixel central possuir valor um e apenas um de seus *pixels* vizinhos também possuir valor um, então o local será marcado como uma minúcia do tipo terminação.

Figura 16 – Bifurcação e Terminação



Fonte: Casado e Paiva (2008)

Esse método de encontrar os pixels com mesmo valor na vizinhança é conhecido por *Crossing Number*, no qual o tipo de minúcia será definido em função do resultado obtido para o *Cn*. Dependendo do como for a relação entre o pixel central *px* e os pixels vizinhos, indicará se o ponto em questão é uma minúcia ou uma linha simples (CASTRO, 2008). A Tabela 1 mostra o tipo de minúcia que irá se encontrar em função do valor do *Cn*.

Tabela 1 – Valores do *Cn* para o método *Crossing Number*

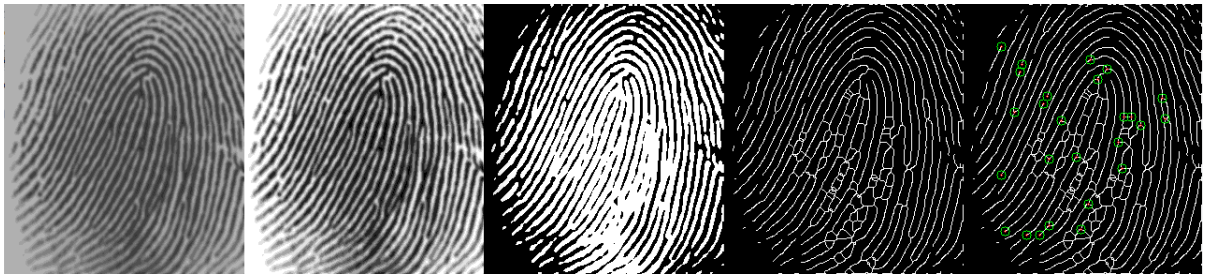
Valor do *Cn*	Tipo de Minúcia
0	Ponto isolado
1	Terminação
2	Crista normal
3	Bifurcação
4 ou mais	Não considerado

Fonte: Castro (2008)

O Crossing Number é um método bastante comum utilizado para extrair minúcias em imagens de IDs. É notável a quantidade de trabalhos que se utilizaram do algoritmo Crossing Number na fase de Extração, principalmente nos casos onde foi utilizado a técnica de Afinamento.

Como exemplo do uso do CN em suas pesquisas podemos citar Casado e Paiva (2008), Castro (2008), Mazi e Júnior (2009) e Prestes (2011).

Figura 17 – Etapas do Pré-processamento e Extração de Minúcias



Fonte: O Autor

3.3.1 Templating

O *Template* é uma representação mais simples da imagem, a qual consiste mais propriamente das informações relevantes da imagem a ser representada. No nosso caso, a forma utilizada foi um vetor de inteiros, o qual possui as posições das minúcias na imagem seguida do seu respectivo tipo, sendo que o tipo 0 (zero) refere-se às minúcias do tipo bifurcação e o tipo 1 (um) às minúcias do tipo terminação.

A cada execução do *Crossing Number*, caso uma minúcia seja detectada, a sua posição (x, y) na imagem seguida do tipo de minúcia detectada (0 ou 1) são adicionados no Vetor. Ao fim da execução tem-se um vetor com todas as minúcias encontradas. Após todas as minúcias serem encontradas, é aplicado um algoritmo que filtra os atributos detectados de modo a excluir falsas minúcias e que através do cálculo da Distância Euclidiana entre os pontos marcados como relevantes exclui também minúcias que estejam muito próximas. O resultado do Filtro é refletido em um novo vetor, o qual possui as características que realmente são importantes.

O vetor resultante é então impresso em um arquivo texto para que posteriormente seja convertido ao formato utilizado pela ferramenta WEKA para análise dos resultados obtidos pelos algoritmos de *Machine Learning* disponíveis nela.

3.4 Preparação de Dados para RNA

3.4.1 Criação dos Datasets

Para montar os conjuntos de dados (*datasets*) para treinamento e testes com a RNA foram utilizados os Databases da FVC - *Fingerprint Verification Competition* (FVC, 2002), bem como as imagens capturadas pelo nosso sistema de aquisição de imagem.

Os *databases* da FVC são compostos por quatro diretórios (DB1, DB2, DB3, DB4), onde cada diretório contém imagens que representam 10 indivíduos, sendo 8 exemplares de cada indivíduo, perfazendo um total de 80 imagens por diretório. As imagens que estão nesses

databases normalmente são obtidas de diferentes formas, um diretório pode ter sido obtido através de impressão com tinta e papel especiais, outro por sensor biométrico e alguns são imagens sintéticas geradas por softwares.

Ao montar a nossa base de dados, seguimos o mesmo formato de organização da FVC, onde utilizamos 10 pessoas com 8 exemplares da mesma ID de cada indivíduo.

Cada exemplar de ID é distinto, ou seja, embora sejam imagens da mesma ID, cada imagem é única, variando em posição e angulação. Com isso, ao montar o *Dataset* que vai pro software WEKA, é necessário passar um atributo que informe ao algoritmo que as instâncias de cada grupo de 8 imagens representam o mesmo indivíduo. Esse atributo é chamado de Classe e além de ser utilizado pra dizer à RNA que essas 8 imagens representam a mesma pessoa, também é atavés dele que se pode visualizar a quantidade de instâncias classificadas correta ou incorretamente.

Para montar os *DataSets*, tanto das imagens da FVC como as nossas amostras, foi implementado em C++ um programa que automatizou esse processo, no qual cada imagem passada como entrada para o programa passa por todas as fases da etapa de Pré-Processamento, seguindo para a Extração de Características e Pós-processamento, resultando no *Template* final, que é gravado em um arquivo de texto, onde cada linha representa uma instância que é composta do nome da imagem, as minúcias e seus respectivos tipos e a classe. Nesse ponto, para que os *templates* fossem padronizados, decidimos manter no arquivo 15 minúcias de cada imagem.

Assim, tivemos como resultado os Datasets utilizados para teste dessa proposta, os quais estão organizados da seguinte forma:

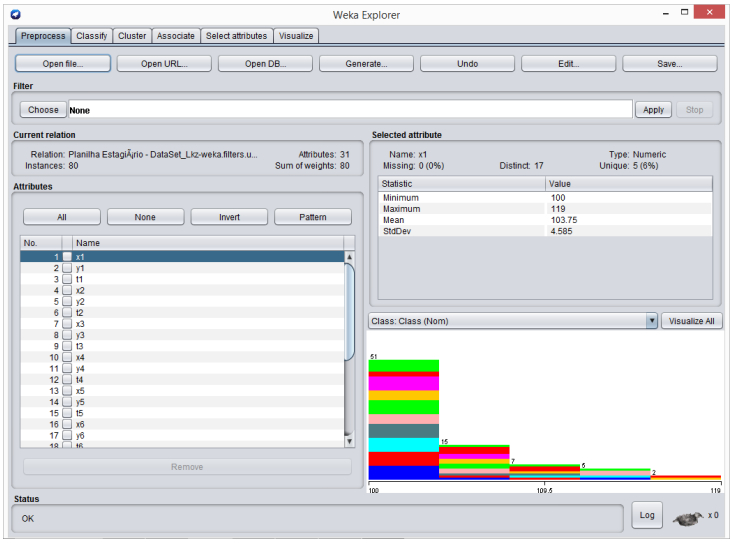
- Dataset referente ao database de imagens da FVC 2002 com o uso de todos os diretórios, totalizando 320 instâncias e 40 classes (Dataset 1).
- Dataset referente ao nosso database, obtido via sensor biométrico e Arduino, consistindo em 80 instâncias e 10 classes (Dataset 2).
- Dataset da FVC 2002 para efeito de comparação, sendo constituído apenas do diretório DB1, ou seja, 80 instâncias e 10 classes (Dataset 3).

Após os dados estarem devidamente gravados no arquivo texto, este foi convertido ao formato de planilha do Microsoft Excel, o qual por sua vez foi convertido ao formato CSV (*Comma-separated values*) através do conversor online Convertio (https://convertio.co/pt/xlsx-csv/). Esse arquivo CSV é lido nativamente pelo software WEKA o qual faz a conversão para seu formato padrão ARFF (*Attribute-Relation File Format*).

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Os testes experimentais utilizando a ferramenta WEKA 3.8.2 foram realizados em um computador com processador Intel Core i5, 4GB de memória RAM rodando o Sistema Operacional Microsoft Windows 8.1.

Figura 18 – Ferramenta *WEKA*



Fonte: O Autor

Os experimentos foram realizados com os Datasets gerados a partir das bases de dados. O WEKA permite que os parâmetros dos algoritmos sejam alterados de modo que proporciona a possibilidade de se fazer uma análise de um melhor cenário, conforme evidenciado por Neves et al. (2017). No entanto, ao mudarmos os parâmetros seguindo a metodologia deles, não se percebeu grandes melhorias nos testes com o nosso formato de dataset, exceto quanto à alteração do valor da *Learning Rate* (Taxa de Aprendizado) do valor padrão 0.3 para 0.2, onde se observou uma melhoria considerável, o que nos fez adotar essa configuração.

O Experimento 1 consistiu na aplicação do algoritmo *Multilayer Perceptron* em todos os Datasets, mantendo-se os parâmetros padrão da ferramenta WEKA, alterando apenas a *Learning Rate* e utilizando a técnica *Using Training Set*, a qual avalia o quanto a RNA classifica corretamente as instâncias ao realizar os testes sobre o mesmo conjunto de dados usado para treinamento.

O Experimento 2, consistiu em utilizar-se de subconjuntos (*trainning set* e *test set*). Essa técnica consiste em dividir a base de dados em conjuntos exclusivos de dados e utilizar um para treinamento e o outro para os testes. Neste caso, para dividir o dataset foi utilizado o próprio WEKA, o qual possui algoritmos de filtragem que possibilitam fazer esse tipo de operação. Utilizamos o filtro *Resample* o qual possibilitou a criação de dois subsets através de *Percentage Split*, que é justamente essa divisão do conjunto total de dados, onde definimos 60% da base total para treinamento e os 40% restantes para testes. A técnica utilizada foi a (*Supplied Test Set*), que

consiste basicamente na criação de um modelo baseado no treinamento e na reavaliação deste ao se usar um conjunto de dados para teste.

Para demonstração dos resultados vamos analisá-los pelas métricas de Taxa de Acerto, Taxa de Falso Positivo e Matriz de Confusão.

4.1 Comparação das Taxas de Acerto

A Tabela 2 demonstra o percentual de acertos da RNA para cada Dataset de acordo com suas especificidades, sendo resultado do Experimento 1.

Tabela 2 – Taxas de acerto do *Multilayer Perceptron* para o Experimento 1

DataSet	Acerto (%)
DataSet 1 - FVC2002 (320 instâncias - 40 Classes)	86,79
DataSet 2 - Nossa Base (80 instâncias - 10 Classes)	86,25
DataSet 3 - FVC2002 (80 instâncias - 10 Classes)	95

Fonte: O Autor

Ao compararmos os resultados desse experimento, podemos perceber que no Dataset 3 o acerto é bem maior se comparado ao DataSet 1. Quando utilizamos apenas um diretório da FVC 2002 (Dataset 3), o acerto é significativamente maior, passando de 86,79% a 95%.

A seguir vamos verificar a acurácia da RNA ao se utilizar conjuntos de dados distintos para treinamento e teste, obtidos por *Percentage Split* através do filtro *Resample* disponível na ferramenta WEKA. A RNA foi treinada com o conjunto de dados definido para esta finalidade (training set), em seguida gerou-se o modelo referente a esse treinamento possibilitando uma reavaliação do modelo gerado com a atuação do algoritmo sobre o subconjunto de dados definidos para teste (*test set*). Esse processo foi feito para os *Datasets* 1, 2 e 3. Os resultados obtidos são mostrados na Tabela 3:

Tabela 3 – Taxas de acerto do *MLP* para o Experimento 2

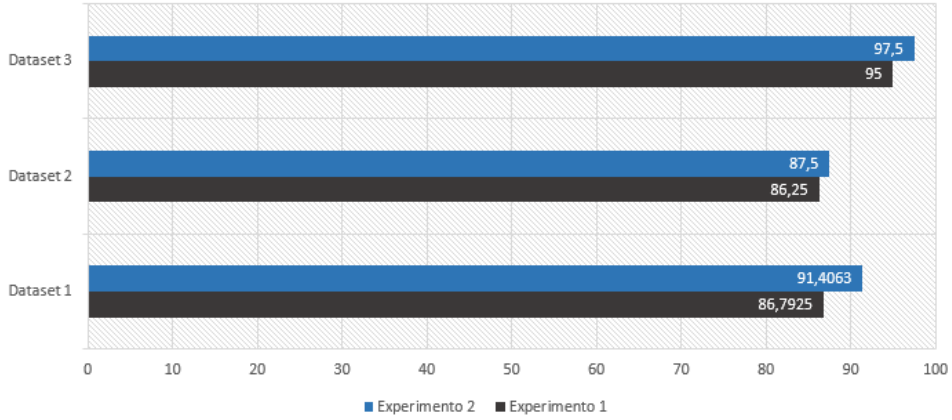
DataSet	***Acerto (%) - Test Set 40%***
DataSet 1	91,40
DataSet 2	87,5
DataSet 3	97,5

Fonte: O Autor

O Gráfico 1 mostra os resultados de ambos experimentos para cada dataset utilizado, onde claramente pode-se notar que embora o resultado obtido com o Dataset 3 tenha sido o melhor nos dois experimentos, principalmente pelo fato de se ter menos classes, o dataset que

apresenta a melhoria mais significativa em relação a seu resultado preliminar é o Dataset 1. Essa melhora considerável se deve ao fato de que a quantidade de dados utilizada na criação do modelo (treinamento) impacta diretamente no resultado da reavaliação com um Test set.

Gráfico 1 – Comparação das Taxas de Acerto do *MLP*



Fonte: O Autor

4.2 Taxas de Falso Positivo

Segundo Bolle et al. (2013) diferentes terminologias são utilizadas para representar a precisão de um sistema considerando a ocorrência de erros. Em sistemas biométricos a Taxa de Falso Positivo (FPR - False Positive Rate) é conhecida por Taxa de Falsa Aceitação - FAR, a qual é a razão das instâncias impostoras que foram classificadas como genuínas (GONZAGA, 2018).

Pra se chegar nos valores dessas taxas são levados em conta as quantidades de instâncias classificadas correta ou incorretamente Sendo que, dessa forma, essa razão implica no nível de segurança do SB. Assim, definimos que esta é uma das métricas que precisa ser avaliada. Gonzaga (2018) aborda os valores necessários para se chegar nessa métrica:

- Verdadeiro Positivo - O elemento de entrada é Genuíno (Positivo) e o Classificador o classifica como Positivo.
- Verdadeiro Negativo - O elemento de entrada é Impostor (Negativo) e o Classificador o classifica como Negativo.
- Falso Positivo - O elemento de entrada é Impostor (Negativo) e o Classificador o classifica como Positivo.
- Falso Negativo - O elemento de entrada é Genuíno (Positivo) e o Classificador o classifica como Negativo.

A equação 4.1 demonstra como se chega ao valor da Taxa de Falso Positivo (FPR), onde FP é o número de Falsos Positivos (*False Positive*) e TN é o número de Verdadeiros Negativos

(*True Negative*). O WEKA retorna o valor dessa taxa para cada classe analisada, e informa o valor final da FPR referente ao conjunto de dados testado.

$$FPR = \frac{FP}{TN + FP} \tag{4.1}$$

A Tabela 4 mostra os valores da Taxa de Falso Positivo resultantes do Experimento 1 realizados sobre os Datasets:

Tabela 4 – Taxas de Falso Positivo do *MLP* para o Experimento 1

DataSet	*FPR (%)*
DataSet 1	0,3
DataSet 2	1,5
DataSet 3	0,6

Fonte: O Autor

Observa-se que o melhor resultado para o Experimento 1 foi obtido pelo Dataset 1, com uma FPR de 0,3%. Como se trata da taxa de erro que afeta a segurança do sistema, vale lembrar que quanto menor o valor da FPR mais seguro é o SB e menos Falsas Aceitações ocorrem.

Já no Experimento 2, realizado também sobre todos os datasets, buscou-se reduzir as taxas de erro do tipo FP visando melhorar a segurança caso esse sistema estivesse em um contexto real. Os resultados podem ser vistos na Tabela 5:

Tabela 5 – Taxas de Falso Positivo do *MLP* para o Experimento 2

DataSet	*FPR (%) - Test Set 40%*
DataSet 1	0,2
DataSet 2	1,4
DataSet 3	0,3

Fonte: O Autor

Analisando os resultados obtidos através do Experimento 2, podemos perceber que em todos os casos os resultados foram melhorados, já que o objetivo era reduzir as taxas de erro. Contudo, também é perceptível que o Dataset 3 teve a maior melhoria em relação ao resultado do Experimento 1.

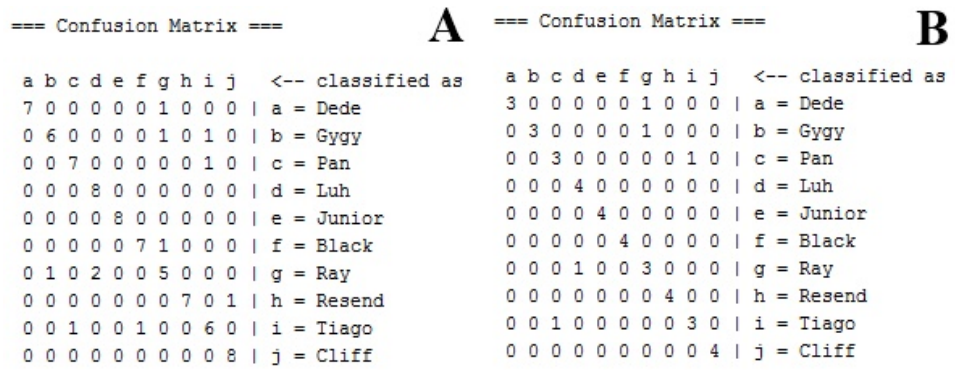
O Dataset 2, resultou em taxas relativamente aceitáveis. A acurácia é menor se comparada aos resultados dos demais datasets, mas ainda assim obteve-se uma taxa de acerto de 87,5% e uma FPR de 1,4%. Isso pode se dever ao fato de que a qualidade das amostras adquiridas pelo sensor influenciam no resultado final. O que levanta o questionamento de que se é realmente viável a implantação de um SB de baixo custo. Pois, a depender do tipo de aplicação, ou do nível de segurança necessário, essa pode não ser a melhor alternativa.

4.3 Matriz de Confusão

Uma outra forma de se analisar os resultados é através da Matriz de Confusão. Ela demonstra como o algoritmo classificou cada instância, de forma que as quantidades de instâncias classificadas corretamente estão na diagonal principal da matriz.

Vamos analisar as Matrizes de Confusão dos Datasets 2 e 3, devido à quantidade de Classes que possuem, sendo mais simples de observar e perceber como a classificação das instâncias ocorreu. Como já foi mencionado, no Experimento 1 o conjunto de dados para esses dois datasets consistiu em 8 amostras de 10 diferentes classes. As linhas da Matriz representam a Classe esperada e as colunas representam a classificação feita pela RNA.

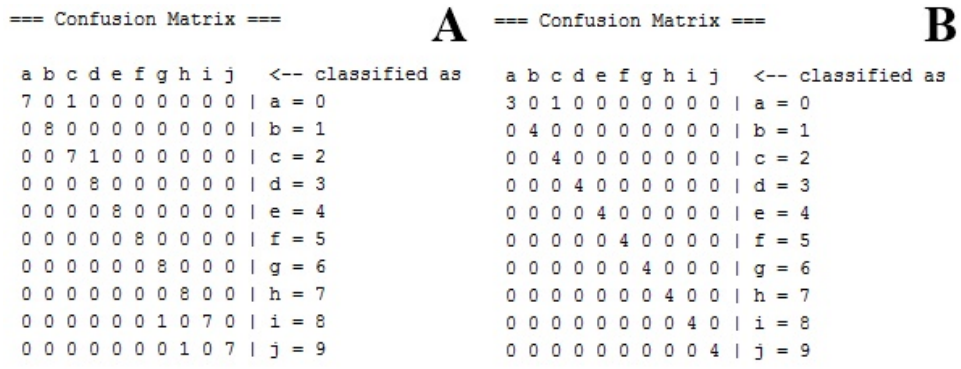
Figura 19 – Matrizes de Confusão do Dataset 2 - Experimento 1 (A) *vs* Experimento 2 (B)



Fonte: O Autor

Na primeira linha da Matriz de Confusão (Figura 19.A) a classe estimada é “Dede” e uma instância foi classificada como “Ray”. Já na segunda linha a classe estimada é “Gygy” e duas instâncias foram incorretamente classificadas, uma como “Ray” e outra como “Tiago”, e assim sucessivamente.

Figura 20 – Matrizes de Confusão do Dataset 3 - Experimento 1 (A) *vs* Experimento 2 (B)



Fonte: O Autor

As Figuras 19.B e 20.B demonstram as matrizes de confusão resultantes do Experimento 2, conforme os resultados listados nas Tabelas 3 e 5, sendo que o melhor resultado foi obtido pelo Dataset 3, com apenas uma instância classificada incorretamente.

5 CONSIDERAÇÕES FINAIS

A biometria tem se mostrado muito popular e útil no contexto de sistemas que necessitam de autenticação, no que tange à garantia da segurança de acesso ou identificação de usuários. Os erros intrínsecos a esses sistemas têm um impacto negativo na Segurança, uma vez que podem abrir brechas para pessoas não autorizadas acessarem informações, ou se passarem por outro usuário.

O objetivo desta proposta foi analisar o desempenho das Redes Neurais Artificiais *Multilayer Perceptron* no processo de identificação biométrica, visando verificar o quão eficiente elas são no tocante à possibilidade de acontecerem erros nesse processo, prezando por um nível aceitável de segurança. Para isso foram montados datasets a partir de bancos de dados de imagens de ID, tanto de uma Competição de Verificação de Impressões Digitais (FVC), como do banco de dados montado através da captura de IDs pelo nosso Sistema de Aquisição de Imagem da ID, ambos tiveram todas as imagens tratadas e suas minúcias extraídas, gerando-se um *template* que representa cada ID.

Os datasets foram analisados por meio da ferramenta WEKA, comumente utilizada em aplicações de Mineração de Dados e Aprendizado de Máquinas, subáreas da Inteligência Artificial. Foi utilizado o algoritmo de Redes Neurais Artificiais *Multilayer Perceptron* para classificar as instâncias dos datasets. Os resultados foram analisados através das métricas de Taxa de Acerto, Taxa de Falso Positivo e Matriz de Confusão.

Nos experimentos realizados com os Datasets, foram feitos testes usando a técnica *Using Training Set* (Experimento 1) e comparada à técnica *Supplied Test Set* que foi feita através do método *Percentage split* (Experimento 2). Os melhores resultados foram obtidos no Experimento 2 com o Dataset 3, criado a partir de um diretório do Banco de Dados da FVC 2002 (80 instâncias e 10 classes), atingindo uma taxa de acerto de 97,5% e uma taxa de falso positivo de apenas 0,3%. Para o Dataset 2, gerado pelo sistema de aquisição de imagem implementado com arduino, os resultados foram de 87,5% e 1,4% de taxa de acerto e taxa de falso positivo respectivamente.

Como trabalhos futuros deixamos a implementação de um Sistema Biométrico Automático que implemente as etapas definidas neste trabalho; melhorar o SB, utilizando outros módulos do Arduino, como um Ethernet Shield para intermediar a conexão via rede, possibilitando utilizar recursos como banco de dados em um servidor; Verificar a viabilidade de se usar um outro formato de *Dataset*, adicionando informações de distância relativa entre as minúcias; Testar a eficiência das RNAs com um conjunto maior de dados e também realizar um estudo comparativo com outros algoritmos.

REFERÊNCIAS

ARDUINO. **Arduino - Introduction**. 2018. Disponível em: <https://www.arduino.cc/en/Guide/Introduction>. Acesso em: 18 abril 2018.

ARDUINO. **Arduino Uno Rev3 - Most Popular - Arduino**. 2018. Disponível em: <https://store.arduino.cc/usa/arduino-uno-rev3>. Acesso em: 18 abril 2018.

BOLLE, R. M. et al. **Guide to biometrics**. [S.l.]: Springer Science & Business Media, 2013.

BONATO, C. d. S. Utilização da técnica de afinamento como melhoria na extração de minúcias de impressões digitais. 2011.

CASADO, R. S.; PAIVA, M. S. V. d. Extração de minúcias em imagens de impressões digitais. 2008.

CASTRO, T. d. S. Identificação de impressões digitais baseada na extração de minúcias. **Juiz de Fora**, p. 08–25, 2008.

COSTA, K. M. d. Investigação de plataformas computacionais para identificação de impressão digital. **Monografia apresentada à Universidade Federal de Pernambuco. Recife-PE**, 2003.

COSTA, L. R.; OBELHEIRO, R. R.; FRAGA, J. S. Introdução á biometria. **Livro texto dos Minicursos do VI Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg2006). SBC: Porto Alegre**, v. 1, p. 103–151, 2006.

FARIA, D. R. Reconhecimento de impressões digitais com baixo custo computacional para um sistema de controle de acesso. 2005.

FONTANA, D. R.; MARIM, L. Sistema de autenticação/identificação pessoal biométrica através da palma da mão com o auxílio de redes neurais artificiais. **Anais do 14O Encontro de Iniciação Científica e Pós-Graduação do ITA–XV ENCITA, SP, Brasil**, 2009.

FVC. **FVC2002: the Second International Fingerprint Verification Competition**. 2002. Disponível em: <http://bias.csr.unibo.it/fvc2002/>. Acesso em: 01 junho 2018.

GONZAGA, A. **Métricas de Avaliação de Classificadores**. 2018. Disponível em: <http://iris.sel.eesc.usp.br/sel886/Aula_9.pdf>. Acesso em: 09 julho 2018.

HAYKIN, S. **Redes neurais: princípios e prática**. [S.l.]: Bookman Editora, 2007.

HOWSE, J. et al. **OpenCV 3 Blueprints**. [S.l.]: Packt Publishing Ltd, 2015.

LOESCH, C.; SARI, S. T. **Redes neurais artificiais: fundamentos e modelos**. [S.l.]: Ed. da FURB, 1996.

MARENGONI, M.; STRINGHINI, S. Tutorial: Introdução à visão computacional usando opencv. **Revista de Informática Teórica e Aplicada**, v. 16, n. 1, p. 125–160, 2009.

MAZI, R. C.; JÚNIOR, A. D. P. Identificação biométrica através da impressão digital usando redes neurais artificiais. **Anais do XIV ENCITA**, p. 19–22, 2009.

NEVES, C. H. A. S. C. et al. Parametrização de rede multilayer perceptron para classificação de impressões digitais. **Journal of Exact Sciences**, v. 23, n. 2, p. 30–45, 2017.

NOGUEIRA, F. R. Captura de sinal biométrico utilizando arduino. 2011.

OPENCV. **Histogram Equalization - OpenCV 2.4.13.6 documentation**. 2018. Disponível em: <https://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram_equalization/histogram_equalization.html>. Acesso em: 27 junho 2018.

OPENCV. **OpenCV - Image Thresholding**. 2018. Disponível em: <https://docs.opencv.org/3.4/d7/d4d/tutorial_py_thresholding.html>. Acesso em: 29 junho 2018.

PRESTES, Á. N. Sistema de reconhecimento de impressões digitais. 2011.

SILVA, C. et al. A segurança através da biometria. **SEGeT–Simpósio de Excelência em Gestão e Tecnologia. Associação Educacional Dom Bosco. Resende-RJ**, 2007.

SILVA, M. S. da; FILHO, V. S. Biometria através de impressão digital biometrics through digital printing. **Cadernos UniFOA**, v. 6, n. 15, p. 19–28, 2017.