UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GABRIEL ALMEIDA AZEVEDO

CONTROLE DE ACESSO A AMBIENTE RESTRITO, A PARTIR DA IDENTIDADE VOCAL, UTILIZANDO COEFICIENTES MFCC E CLASSIFICADOR *K-MEANS*

CAMPINA GRANDE - PB

2021

GABRIEL ALMEIDA AZEVEDO

CONTROLE DE ACESSO A AMBIENTE RESTRITO, A PARTIR DA IDENTIDADE VOCAL, UTILIZANDO COEFICIENTES MFCC E CLASSIFICADOR *K-MEANS*

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientadora: Professora Dra. Joseana Macêdo Fechine Régis de Araújo.

CAMPINA GRANDE - PB

2021

A994c Azevedo, Gabriel Almeida.

Controle de acesso a ambiente restrito, a partir da identidade vocal, utilizando coeficientes MFCC e classificador K-Means. / Gabriel Almeida Azevedo. – 2021.

12 f.

Orientadora: Profa. Dra. Joseana Macêdo Fechine Régis Araújo.

Trabalho de Conclusão de Curso - Artigo (Curso de Bacharelado em Ciência da Computação) - Universidade Federal de Campina Grande; Centro de Engenharia Elétrica e Informática.

1. Identidade vocal. 2. Processamento digital de sinais de voz. 3. Reconhecimento de voz. 4. Coeficientes MFCC. 5. Clusterização K-Means. 6. Verificação automática de identidade vocal. 7. Coeficientes mel-cepstrais. 8. Controle de acesso a ambientes. 9. Aprendizagem de máquina. 10. Locutores – identificação automática da voz. 11. Mel-Frequency Cepstral Coefficients – MFCC. I. Araújo, Joseana Macêdo Fechine Régis. II. Título.

CDU:004(045)

Elaboração da Ficha Catalográfica:

Johnny Rodrigues Barbosa
Bibliotecário-Documentalista
CRB-15/626

GABRIEL ALMEIDA AZEVEDO

CONTROLE DE ACESSO A AMBIENTE RESTRITO, A PARTIR DA IDENTIDADE VOCAL, UTILIZANDO COEFICIENTES MFCC E CLASSIFICADOR *K-MEANS*

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Professora Dra. Joseana Macêdo Fechine Régis de Araújo
Orientador – UASC/CEEI/UFCG

Professor Dr. Cláudio de Souza Baptista
Examinador – UASC/CEEI/UFCG

Professor Dr. Tiago Lima Massoni
Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 25 de maio de 2021.

CAMPINA GRANDE - PB

ABSTRACT

The machine learning area is a great ally to ensure privacy and security, as it promotes advances in the methods used for access control. The use of techniques for Automatic Recognition of the Voice Identity of Speakers, for authentication purposes, represents one of these advances. Given the above, this article aims to present a system for automatic verification of the vocal identity of speakers, seeking to apply it for authentication and release of access to a restricted environment. The system is based on a pattern recognition task, divided into two stages: training and verification. In the training, techniques were applied for pre-processing the signal (pre-emphasis, division into frames and windowing), extraction of characteristics (Mel-Frequency Cepstral Coefficients - MFCC) and construction of a representative pattern of the vocal identity of each speaker (clustering). In the verification, pre-processing of the signal, extraction of characteristics and authentication occurred, the latter based on the comparison between the test characteristics and the previously stored pattern of the speaker. In the logic decision, thresholds were used for the authentication of an announcer (acceptance, rejection and indeterminacy). The results obtained demonstrate a correct authentication of the speaker in 81% of the cases and a rate of 94.89% of rejection of imposters, proving the efficiency of the proposed approach.

Controle de Acesso a Ambiente Restrito, a partir da Identidade Vocal, utilizando Coeficientes MFCC e Classificador *K-Means*

Gabriel Almeida Azevedo
Unidade Acadêmica de Sistemas e Computação
Universidade Federal de Campina Grande,
Campina Grande, Paraíba, Brasil
gabriel.almeida.azevedo@ccc.ufcg.edu.br

Joseana Macêdo Fechine Régis de Araújo
Unidade Acadêmica de Sistemas e Computação
Universidade Federal de Campina Grande,
Campina Grande, Paraíba, Brasil
joseana@computacao.ufcg.edu.br

RESUMO

A área de aprendizagem de máquina é uma grande aliada para garantir privacidade e segurança, pois promove avanços nos métodos empregados para controle de acesso. O uso de técnicas para Reconhecimento Automático da Identidade Vocal de Locutores, para fins de autenticação, representa um desses avanços. Diante do exposto, este artigo objetiva apresentar um sistema para verificação automática da identidade vocal de locutores[1], buscando aplicá-lo para autenticação e liberação de acesso a ambiente restrito. O sistema baseia-se numa tarefa de reconhecimento de padrões, dividida em duas etapas: treinamento e verificação. No treinamento, foram aplicadas técnicas para pré-processamento do sinal (pré-ênfase, divisão em *frames* e janelamento), extração de características (*Mel-Frequency Cepstral Coefficients* - MFCC) e construção de um padrão representativo da identidade vocal de cada locutor (clusterização). Na verificação, ocorreram o pré-processamento do sinal, extração de características e autenticação, esta última a partir da comparação entre as características de teste e o padrão previamente armazenado do locutor. Na lógica de decisão, foram utilizados limiares para autenticação de um locutor (aceitação, rejeição e indeterminação). Os resultados obtidos demonstram uma autenticação correta do locutor em 81% dos casos e uma taxa de 94,89% de rejeição de impostores, comprovando a eficiência da abordagem proposta.

Palavras-Chave

Controle de Acesso, Verificação Automática de Identidade Vocal, Coeficientes MFCC, Clusterização *K-Means*.

1. INTRODUÇÃO

A tecnologia tem evoluído bastante, adquirindo papel fundamental na área de segurança. No âmbito da segurança física, a tecnologia é aplicada tanto para monitoramento (câmeras, sensores, alarmes sonoros), como também para reconhecimento e autenticação (cartões de proximidade, biometria, reconhecimento facial, reconhecimento por voz). Entretanto, autenticar um usuário não é uma tarefa fácil, pois exige alto grau de acurácia, dado que, eventuais erros podem trazer graves consequências, a depender da importância do objeto/local para o qual se deseja garantir segurança.

Visando promover robustez e qualidade ao processo de autenticação automática, usualmente são aplicados conceitos de *machine learning* (aprendizagem de máquina). Essa área da computação é a base para a comunicação vocal homem-computador. Essa comunicação divide-se em 3 subáreas, a saber: sistema de síntese de voz, de reconhecimento de fala e de reconhecimento de locutor (ou identidade vocal). Este último poderá ser dividido em identificação de locutor e verificação (autenticação) de locutor. No primeiro caso, o locutor será reconhecido a partir da comparação dos dados de sua voz com os de todos os locutores (comparação de 1 para N). Para verificação do locutor, a comparação será feita apenas com os dados do locutor que ele alega ser (comparação 1 x 1).

O reconhecimento de locutor diz respeito à autenticação de alguém por meio da sua voz. Para este cenário, é necessário decidir sobre dois pontos: se haverá dependência de sentença e se o locutor será colaborativo. Quando o sistema é dependente de sentença, este só será capaz de reconhecer o locutor caso ele fale uma sentença predefinida. Quando não se tem esta dependência, o sistema deve ser capaz de reconhecer o locutor a partir de qualquer frase por ele produzida. Este segundo caso é bem mais custoso e adiciona complexidade ao sistema. Um usuário colaborativo ao sistema será aquele que fará o possível para ser reconhecido.

Na próxima seção, estes e outros conceitos serão descritos, de forma mais aprofundada, para facilitar o entendimento da verificação automática da identidade vocal de locutores.

2. TRABALHOS RELACIONADOS

A área de *Speaker Recognition* está cada vez mais presente no interesse de pesquisadores. Dessa forma, novas técnicas e modelos vêm sendo desenvolvidos.

No âmbito da identificação de locutor, pode-se citar o trabalho de Singh [17], que descreve um classificador, obtido a partir de um quantizador vetorial (QV), com características representadas por Coeficientes Mel-Cepstrais e Mel-Cepstrais invertidos, com intuito de recuperar informações complementares relevantes. No trabalho de Suksri [18] tem-se, aliado ao uso de MFCC, classificadores de máxima verossimilhança e SVM (*Support Vector Machine)*. Bharti [3] concluiu que unir MFCC ao QV-LBG (*Linde-Buzo-Gray*) se mostrou eficaz mesmo na presença de ruídos.

No âmbito da verificação de locutor, tem-se trabalhos como o de Schueler [15], que faz uso de modelos ocultos de Markov e MFCC unidos a parâmetros extraídos da análise glotal. Há também trabalhos como o de Melo [11], que usa um quantizador vetorial e faz uma comparação de resultados gerados para o verificador ao extrair as características a partir de coeficientes Mel-Cepstrais e coeficientes LPC (*Linear Prediction Coding*). O autor também apresenta o desenvolvimento do hardware utilizando FPGA - *Field Programmable Gate Array*. Ramos-Lara [14] apresenta a implementação de um verificador usando MFCC e SVM para incorporação em uma FPGA de baixo custo.

Conforme exposto, o uso de MFCC para representação das características de um locutor e um classificador baseado em clusterização, se mostram relevantes para desenvolvimento de um sistema para verificação automática da identidade vocal de locutores.

3. FUNDAMENTAÇÃO TEÓRICA

Nesta seção, serão apresentados conceitos necessários ao entendimento do trabalho desenvolvido.

3.1 Reconhecimento Automático de Voz

Um sistema de reconhecimento automático vocal é um sistema que utiliza a voz como chave biométrica. Estes sistemas abordam as áreas de: reconhecimento de discurso (*Speech Recognition*) e reconhecimento de locutor (*Speaker Recognition*) [12].

O foco do *Speech Recognition* é descobrir o que está sendo dito. Um bom exemplo da aplicação desse sistema é o mecanismo de geração de legendas automáticas no YouTube ou ainda o Echo Dot Alexa, da Amazon, que busca atender a comandos de voz diversos, pré definidos ou não, de seus usuários. No *Speaker Recognition*, o objetivo é descobrir quem está falando. Este segundo pode ainda ser dividido em duas categorias, sendo essas: *Automatic Speaker Recognition (ASR)* e *Automatic Speaker Verification (ASV)*.

No *Automatic Speaker Recognition* busca-se identificar de qual locutor é o sinal de voz submetido ao sistema, tendo como referência uma base de dados de locutores. No *Automatic Speaker Verification*, o objetivo é confirmar a identidade do locutor por meio de um sinal de voz submetido ao sistema, tendo como referência os dados de apenas um locutor [6]. Um ASR tenta responder à pergunta: "De quem é esta voz?", já um ASV responde à pergunta: "Esta voz é do locutor X?"

Tanto no ASR como no ASV, pode-se adotar um modelo de reconhecimento que dependa, ou não, da sentença. Quando o modelo é dependente, ele só será capaz de reconhecer/validar o locutor a partir de uma ou mais frases pré-definidas. Quando o modelo independe da sentença, o sistema deve ser capaz de reconhecer/validar o locutor a partir de qualquer frase por ele dita.

Ainda sobre o reconhecimento automático da identidade vocal, dependendo da funcionalidade que o reconhecedor terá, o locutor pode ter diferentes intenções. Por exemplo, se a polícia deseja utilizar um sistema de reconhecimento de locutor para identificar quem estava falando em uma gravação de uma ligação telefônica, deve-se admitir um locutor não-colaborativo, ou seja, potencialmente, o locutor se valerá de diferentes artifícios para evitar ser reconhecido, tal como alterar características da sua voz, abafar a fala ou ainda se expressar por meio de códigos. Quando o sistema tem a função de verificação/validação, o indivíduo real ou impostor, tentará ao máximo ser reconhecido, ou seja, assumirá papel de locutor colaborativo. Na Figura 1, estão sintetizadas as possíveis variações de um sistema de reconhecimento automático de locutor, foco do trabalho ora descrito.

Figura 1. Tipos de Reconhecimento Automático de Locutor.

Fonte: adaptado de Campbell - 1977 [4].

A verificação de locutor é uma tarefa de reconhecimento de padrões da voz e, portanto, se divide nas etapas de treinamento e verificação (autenticação). Essas etapas contemplam o pré-processamento do sinal, extração de características e construção de padrões. Para extração de características, destaca-se o uso do coeficientes Mel-Cepstrais (MFCC) e a construção de padrões. No contexto da verificação de locutor, uma boa relação custo-benefício pode ser obtida a partir do uso de classificadores baseados em agrupamento (clusterização), a exemplo do *K-Means*.

3.2 Coeficientes Mel-Cepstrais

Os Coeficientes Mel-Cepstrais são amplamente utilizados em sistemas de reconhecimento de locutor devido à melhoria proporcionada na precisão do reconhecimento e por terem se mostrado mais robustos na presença de ruído de fundo em comparação com outras características [14]. Estes coeficientes surgiram devido aos estudos na área de psicoacústica (a ciência que estuda a percepção auditiva humana).

A técnica para extração de atributos MFCC faz uma análise de características espectrais de curta duração, baseando-se no uso do espectro da voz convertido para escala Mel. Estes coeficientes são uma representação definida como o cepstrum de um sinal janelado no tempo, que pode ser obtido a partir da aplicação da Transformada Discreta de Fourier, em escalas de frequência não lineares [7].

A escala Mel foi criada com a intenção de mapear as características da sensibilidade do ouvido humano, pois verificou-se que a percepção humana de frequências de tons puros ou de sinais de voz, não seguem uma escala linear (o aparelho auditivo humano é muito mais sensível a sons de baixa frequência do que a sons de alta frequência). Para cada tom com uma determinada frequência, medida em Hz, é associado um valor medido na escala Mel. Como referência, foi definida a frequência de 1 KHz, com potência 40 dB acima do limiar mínimo da audição do ouvido humano, o que é equivalente a 1000 mel [11].

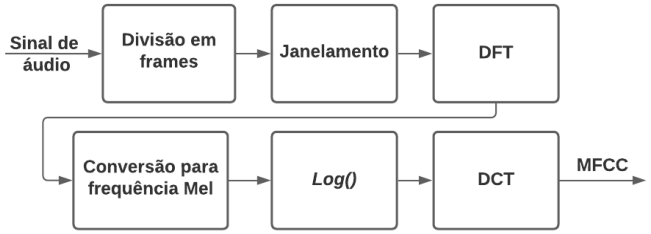
O mapeamento da frequência em Hz para a frequência na escala Mel (Fmel) é obtido segundo a Equação 1.

$$Fmel = 2595\, log\left(1 + \frac{Flinear(Hz)}{700}\right)mel, \quad (1)$$

em que *Flinear* representa a frequência de um tom, medida em Hz. Os valores 2.595 e 700 são obtidos experimentalmente.

O processo para obtenção dos coeficientes Mel-Cepstrais está representado na Figura 2 e consiste em seis etapas [2].

Figura 2. Etapas para cálculo das componentes Mel Cepstrais.



Fonte: Adaptado de [2].

Inicialmente, o sinal de áudio é dividido em *frames* e a superposição é aplicada. Posteriormente, ocorre o janelamento em cada *frame* para minimizar o efeito das descontinuidades do sinal. A etapa 3 consiste no cálculo da DFT (Transformada Discreta de Fourier) para converter cada *frame* do domínio de tempo para o domínio da frequência. O passo 4 é a conversão da escala da frequência linear para a escala Mel. Essa conversão é usualmente feita a partir de bancos de filtros triangulares, com a frequência central do filtro normalmente espaçada uniformemente, no eixo da frequência [2]. A saída do *i*-ésimo filtro pode ser calculada segundo a Equação 2.

$$y(i) = \sum_{j=1}^{N} S_j \Omega_i(j), \quad (2)$$

em que S_j é o espectro de magnitude de N pontos (j = 1 : N) e $\Omega_i(j)$ é a resposta de magnitude em um banco de filtros (i = 1 : M) de canal-M [2]. No quinto passo, o *log* da saída do banco de filtros é calculado. No sexto, e último passo, a DCT (Transformada Discreta do Cosseno) é calculada. Dessa forma, os MFCC podem ser calculados segundo a Equação 3 [2].

$$C_s(n, m) = \sum_{i=1}^{m} (log\, y(i)) cos\left[i\frac{2\pi}{N'}n\right], \quad (3)$$

em que N' é o número de pontos usados para calcular a DFT.

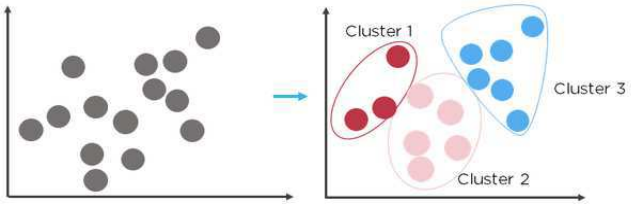
3.3 Clusterização

Clusterização é o agrupamento automático de instâncias similares, uma classificação dos dados. Um algoritmo que clusteriza dados os classifica em conjuntos de dados que 'se assemelham' de alguma forma (independentemente de classes predefinidas) [8]. Algoritmos de clusterização podem ser divididos em 10 grupos principais de acordo com o método adotado [5], sendo esses: métodos hierárquicos, particionais, baseados em densidade, em grade, em modelos, em redes neurais, na lógica *fuzzy*, em kernel, em grafos e baseados em computação evolucionária. Os mais tradicionais são os hierárquicos e os particionais. Métodos particionais dividirão os dados de entrada em K grupos, ou *clusters*. O método particional mais famoso é o *K-Means*.

3.3.1 K-Means

O *K-Means* é um algoritmo de clusterização não supervisionado baseado no cálculo da distância euclidiana quadrática. Seu processo se inicia quando o usuário define a quantidade de *clusters* (K). Nesse momento, o algoritmo escolhe os K centróides iniciais a partir dos dados de uma amostra. Centróides são os elementos que definem o centro do *cluster* (seus representantes). Em seguida, é calculada a distância de cada amostra para cada centróide. Move-se cada amostra para o *cluster* mais próximo (grupo que obteve menor distância entre a amostra e seu centróide). Repete-se este processo de cálculo e reorganização das amostras nos grupos até que as atribuições do *cluster* não mudem, ou uma tolerância definida pelo usuário seja atingida ou ainda, o número máximo de iterações seja alcançado [9]. A distância euclidiana quadrática pode também ser chamada de erro médio quadrático (MSE). Na Figura 3 estão representados dados antes e depois de passar por um processo de clusterização.

Figura 3. Ilustração de uma clusterização de dados.



Fonte: Figura extraída de [10].

4. DESCRIÇÃO DA SOLUÇÃO

A solução proposta se baseia no desenvolvimento de um sistema de verificação de locutor que funcione com duas etapas de autenticação.

Na primeira etapa, o usuário fornecerá sua senha numérica, que será utilizada pelo sistema como chave de seleção do padrão do locutor a ser verificado/autenticado.

Na segunda etapa, será fornecida uma senha vocal (sinal de áudio do locutor, com uma sentença previamente definida), a qual será processada (pré-processamento, extração de características e cálculo de uma medida de distorção em relação ao padrão previamente armazenado e selecionado na etapa anterior) para que o sistema verifique se o locutor é quem ele alega ser. Dado o valor de distorção, a partir de dois limiares (Limiar 1 e Limar 2), o sistema retornará uma das três respostas: locutor verificado/autenticado (distorção igual ou inferior ao Limiar 1), locutor não reconhecido (distorção superior ao Limiar 2) e indeterminação (valor da distorção entre os dois limiares). Esta última objetiva reduzir as taxas de falsa rejeição, bem como de falsa aceitação (para impostores).

Com o propósito de avaliar a conexão desse sistema a um dispositivo de hardware que possibilitará o controle da abertura da porta para acesso restrito por meio da identidade

vocal, será apresentada uma simulação do circuito responsável por essa conexão.

5. METODOLOGIA

Nesta seção, será descrita a metodologia adotada para desenvolvimento da solução. Inicialmente, procedeu-se à construção da base de dados, seguida do desenvolvimento da solução (software) e, por fim, da simulação do módulo de abertura do ambiente de acesso restrito.

A descrição da solução está apresentada na Figura 4, que consiste nas seguintes fases: treinamento e verificação.

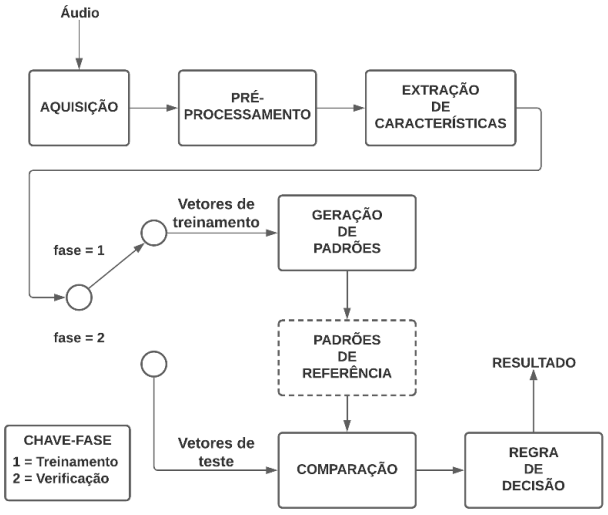
Na fase de treinamento, tem-se:

- Aquisição dos sinais de voz;
- Pré-processamento dos sinais;
- Obtenção dos vetores de características; e
- Geração dos padrões de referência (K-Means), um para cada locutor.

Na fase de verificação, tem-se:

- Aquisição dos sinais de voz;
- Pré-processamento dos sinais;
- Obtenção dos vetores de características;
- Comparação (cálculo do RMSE); e
- Regra de decisão.

Figura 4. Descrição do Verificador da Identidade Vocal.



Fonte: Figura adaptada de [13].

O software foi desenvolvido utilizando a Linguagem Python e fez uso das seguintes bibliotecas:

- *Numpy*: Oferece alto desempenho para manipulação de dados multidimensionais como vetores e matrizes;
- *SciPy*: É um módulo de *Python* que depende do *NumPy*, e oferece métodos para estatística, otimizações, álgebra, processamento de sinais e imagens;
- *Speech features*: provê um método para extração de coeficientes MFCC.
- *Scikit Learn*: É uma biblioteca focada no aprendizado de máquina, seja esse supervisionado ou não e oferece, por exemplo, uma implementação do algoritmo *K-Means*.

5.1 Base de Dados

A Base de dados é um dos pontos cruciais para a etapa de treinamento de qualquer algoritmo de aprendizagem de máquina. Existem diversas bases de dados de voz de qualidade, a exemplo da Mozilla Common-Voice, Vox-Celeb e a CEFALA-1.

Entretanto, com o objetivo de vivenciar todas as etapas do processo de desenvolvimento, proporcionando também mais controle sobre os dados adquiridos, optou-se pela construção de uma base de dados.

Idealmente, cada gravação deveria ocorrer em um ambiente controlado, tal como um estúdio, com revestimento especial para isolamento acústico e redução de reverberação, permitindo a criação de uma base de alta qualidade. Entretanto, dada a situação atual de pandemia e o curto prazo para desenvolvimento, as gravações se deram sem protocolos rígidos de aquisição, exigindo mais robustez na autenticação dos locutores. Foram convidados 10 locutores, sendo 5 homens e 5 mulheres. Cada locutor, que aceitou participar do processo de coleta, recebeu um formulário com esclarecimentos sobre a finalidade dos sinais de áudio da base. Tendo concordado com os termos apresentados, cada participante foi orientado a escolher uma frase, de um conjunto de 20 frases foneticamente balanceadas [1], para ser a sua senha vocal.

Cada usuário, de posse da sua frase, fez a gravação dos sinais de áudio, preferencialmente em um ambiente sem ruídos, com 30 repetições da sua frase e mais 5 repetições de cada frase dos demais locutores. Os usuários foram orientados a não gravar todos os áudios em uma única sessão, evitando assim que o cansaço da repetição gerasse variações indesejadas nos sinais de áudio. Todos os áudios foram capturados a partir dos celulares dos participantes.

Ao final do processo de coleta para os 10 locutores, foram obtidos 750 sinais de áudio. Os áudios foram coletados em formato OGG, com taxa de amostragem de 48.000 amostras/s, em modo estéreo e com 32 bits por amostra. Na Figura 5, é apresentado um fragmento de um sinal de áudio de um locutor.

Figura 5: Fragmento de sinal de áudio de um locutor.

Fonte: autoria própria.

5.2 Pré-Processamento e Divisão da Base

A finalidade desta etapa é reduzir os efeitos indesejados incorporados ou presentes no sinal de voz, preparando-o para as etapas seguintes do processo de verificação.

Inicialmente, todos os sinais de áudio estavam em formato OGG e com taxa de amostragem de 48 K amostras/s. Estes foram convertidos para o formato WAV, com taxa de amostragem de 44,1 K amostras/s (adequada ao processamento) e, posteriormente, passaram por um processo de remoção dos intervalos de silêncio.

A base de dados foi dividida da seguinte forma: para cada locutor foram utilizados 20 sinais de áudio para treinamento, 10 sinais para teste e 45 sinais de áudio de impostores (demais

locutores falando a senha vocal do locutor original) utilizados apenas na fase de teste.

5.3 Extração de Características

Nessa etapa, foram extraídos os coeficientes Mel-Cepstrais de cada sinal de áudio utilizado na fase de treinamento. Para tanto, foi utilizada a biblioteca *python speech features*, que provê um método para extração de coeficientes MFCC, dado um arquivo de áudio como entrada. Este método permite configurar vários parâmetros, tais como taxa de amostragem, tamanho da janela a ser aplicada, salto entre janelas sucessivas, número de coeficientes a serem extraídos, tamanho da transformada de Fourier (FFT), função de janelamento, coeficiente para o filtro de pré-ênfase, lifter, e outros. A seguir, a configuração utilizada.

- Taxa de Amostragem: 44.100 amostras/s;
- Janela: Hamming;
- Janelamento: 20 ms;
- Superposição: 50%;
- Pré-ênfase: 0,97; e
- Tamanho do vetor de características por *frame*: 13 coeficientes Mel-Cepstrais.

5.4 Fase de Treinamento

Nesta fase, é obtido o conjunto de padrões de referência, um para cada locutor. Para a aplicação ora descrita, o treinamento ocorreu de forma *offline*, dado que os padrões de referência não necessitam ser atualizados, desde que o conjunto de locutores cadastrados no sistema e suas respectivas sentenças permaneçam inalterados.

5.4.1 Geração de Padrões

Para geração do padrão representativo de cada locutor, foi utilizado o algoritmo de clusterização *K-Means* da biblioteca *Scikit Learn*. Como parâmetros foi adotado:

- *Clusters*: 64; e
- Algoritmo de inicialização: *K-Means++*.

O método de inicialização *kmeans++* define os *center clusters* iniciais de uma forma inteligente, a fim de acelerar a convergência. Para cada locutor, foi criada uma instância *K-Means*, recebendo como entrada os vetores de características (coeficientes MFCC). Ao final do treinamento, recuperou-se os *cluster centers* da última iteração (dicionário), os quais consistiram no padrão de referência do referido locutor. Tal procedimento foi adotado para a obtenção do padrão de referência dos demais locutores.

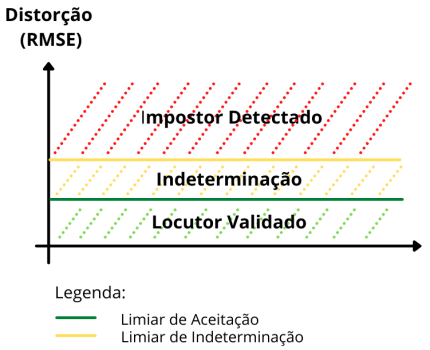
5.5 Fase de Teste

Nesta fase, o processo de pré-processamento e extração dos coeficientes MFCC se repete para o sinal de áudio do locutor a ser testado. Feito isto, é calculada a distorção total (uso do RMSE - *Root mean Squared Error*) entre os vetores MFCC extraídos e o dicionário (gerado na fase de treinamento) do locutor que se deseja autenticar.

A regra de decisão adotada para verificação do locutor consistiu na utilização de três saídas: locutor validado, indeterminação (o sistema não foi capaz de tomar a decisão de autenticar o locutor com grau de segurança adotado) e locutor rejeitado (Figura 6).

Para implementação da regra de decisão, foram utilizados dois limiares para cada locutor, um de aceitação e outro de indeterminação, obtidos empiricamente. O Limiar de aceitação foi definido com base na média da distorção gerada para os sinais de áudio de teste de cada locutor. Os ajustes foram feitos manualmente, sempre com o intuito de maximizar a quantidade de áudios corretos validados (correta aceitação) e minimizar a quantidade de impostores validados (falsa aceitação). O limiar de indeterminação ficou definido como 1,05% do limiar de aceitação, para todos os locutores.

Figura 6: Regra de Decisão.



Fonte: autoria própria.

6. RESULTADOS E DISCUSSÕES

Na Tabela 1 é apresentada a sumarização dos resultados obtidos durante a fase de verificação de cada locutor. A partir dos resultados, pode-se observar que a verificação/autenticação atingiu uma taxa de acerto global de 81% (81 dos 100 sinais áudio de teste foram autenticados corretamente). Para nenhum locutor, foi obtida uma taxa de acerto de 100 %, mas a maioria ficou entre 80% e 90%. A falsa rejeição ocorreu em áudios de apenas quatro dos dez locutores, totalizando uma taxa global de falsa rejeição de 6%. A adoção dos dois limiares se mostrou eficaz na redução das taxas de falsa aceitação.

Tabela 1. Desempenho do Sistema (Correta Aceitação).

Locutor	Validados	Invalidados	Dúvida	Total Parcial
L1	80%	0%	20%	100%
L2	80%	10%	10%	100%
L3	70%	0%	30%	100%
L4	90%	0%	10%	100%
L5	70%	20%	10%	100%
L6	80%	20%	0%	100%
L7	90%	0%	10%	100%
L8	80%	0%	20%	100%
L9	90%	10%	0%	100%
L10	80%	0%	20%	100%
Total Global	81%	6%	13%	100%

Fonte: autoria própria.

Na Tabela 2, são apresentados os resultados dos testes para os impostores. Para cada locutor, foram utilizados 45 sinais de áudio de impostores, totalizando 450 sinais de áudio. Desses, apenas 23 resultaram em falsa aceitação, representando uma taxa de erro de 5,11 %, ou seja, em 94,89 % dos casos, a regra de decisão acerta em não validar impostores. Vale salientar ainda, que para dois locutores, foram obtidos excelentes resultados, pois nenhum de seus impostores foi validado erroneamente.

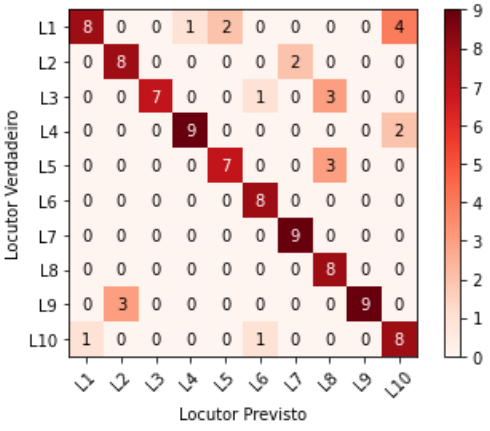
Tabela 2. Desempenho do Sistema (Falsa Aceitação).

Locutor	Validados	Invalidados	Dúvida	Total Parcial
L1	2,22%	91,11%	6,67%	100%
L2	6,67%	80%	13,33%	100%
L3	0%	97,98%	2,22%	100%
L4	2,22%	82,23%	15,55%	100%
L5	4,44%	91,11%	4,44%	100%
L6	4,44%	82,23%	13,33%	100%
L7	4,44%	75,56	20%	100%
L8	13,33%	55,56%	31,11%	100%
L9	0%	80%	20%	100%
L10	13,33%	55,56%	31,11%	100%
Total Global	5,11%	79,11%	15,78%	100%

Fonte: autoria própria.

Na Figura 7 está apresentada a matriz de confusão resultante dos testes com locutores verdadeiros e impostores. O objetivo da diagonal principal é que se atinja o valor máximo para cada locutor, 10 (100% de correta aceitação). Nas demais células, o valor varia entre 0 e 5 e deseja-se atingir sempre o 0 (0% de falsa aceitação). Assim, a partir do código de cores utilizado na figura, a diagonal principal deve apresentar um vermelho mais intenso enquanto que as outras células devem ter tons mais claros. Os resultados obtidos comprovam a eficiência do sistema.

Figura 7: Matriz de confusão do verificador



Fonte: autoria própria.

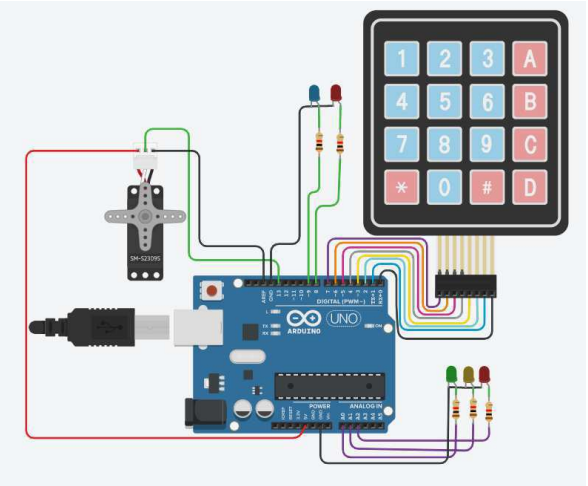
Em comparação com o desempenho apresentados nos trabalhos relacionados a verificação de locutor, a exemplo de [15], que uniu a técnica de extração de MFCC com parâmetros extraídos do sinal glotal, e atingiu uma acurácia de 93,3%, o trabalho ora apresentado ficou um pouco abaixo. Entretanto, um ponto de impacto muito forte foi a base de dados. O processo de coleta dos áudios não pôde seguir um processo rígido. Outro fator limitante é que, na fase de treinamento, foram utilizados apenas 20 sinais de áudio por locutor, diferente desses trabalhos que utilizam bases de dados bastante volumosas.

Dada a observação dos resultados, conclui-se que os MFCC conseguem representar informações relevantes do sinal de áudio e que, é possível construir um verificador de locutor relativamente simples e atingir níveis aceitáveis de desempenho. Com a extração dos MFCC combinada ao algoritmo *K-Means,* foi possível atingir 81% de acurácia para os locutores verdadeiros e 94,89% de rejeição para os impostores.

6.1 Hardware para Liberação do Acesso ao Ambiente Restrito

O componente de hardware a ser utilizado para liberação do acesso ao ambiente restrito foi desenvolvido apenas no âmbito da simulação. A construção desse módulo não faz parte do escopo atual da solução. Entretanto, a sua simulação fornece subsídios relevantes à compreensão da conexão software-hardware para acesso a um ambiente restrito. Para tanto, foi utilizada a ferramenta *TinkerCad* (Figura 8).

Figura 8. Simulação do módulo de hardware para acesso ao ambiente restrito.



Fonte: Adaptado de [16].

A implementação[2] foi realizada utilizando um Arduino Uno e os seguintes componentes:

- 1 *KeyPad* 4x4;
- 5 LED;
- 5 Resistores;
- 1 micro servo; e
- *Jumps*.

[2] https://www.tinkercad.com/things/0CGy150VI2A

O circuito funciona da seguinte forma: Os dois LED de cima referem-se ao primeiro passo de autenticação, a senha numérica. Esta senha permitirá realizar a verificação vocal unicamente para o locutor portador da senha. O LED vermelho inicia aceso e só se apaga caso o usuário digite a senha correta. Nesta ocasião, o LED azul acende. Após essa primeira etapa, o monitor serial fica liberado para receber a resposta do verificador (na simulação, o usuário digita o valor). De acordo com o valor-resposta, o arduino acenderá o LED (verde, amarelo ou vermelho) identificando a resposta final do sistema de verificação. Caso o locutor seja validado, o servo gira, simulando a abertura da trava da porta. Para reiniciar o processo, basta apertar a tecla *.

Esta simulação considera que o processo de aquisição e processamento do áudio do usuário é feito inteiramente via computador. O hardware recebe apenas o sinal que representa a decisão do software.

7. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Nesta seção, serão apresentadas as considerações finais sobre o trabalho desenvolvido[3], suas principais contribuições, suas limitações e também possíveis pontos de aprimoramentos para o futuro.

7.1 Contribuições e Limitações

Como principais contribuições do trabalho, pode-se destacar: que o uso de coeficientes Mel-Cepstrais, para extração de características e do *K-Means*, para agrupamento dessas características, se mostram eficientes para verificação da identidade vocal de locutores, mesmo sendo utilizados sinais de áudio que não passaram por tratamento de ruídos; e, para área de segurança/autenticação de usuário, o desenvolvimento de um modelo de sistema voltado à verificação automática de locutor que pode ser integrado a um hardware de baixo custo.

A maior limitação do trabalho está relacionada à base de dados. Grande empenho foi dado na busca por alguma base já existente que se adequasse bem ao intuito do propósito da pesquisa. Ao decidir pela construção de uma base de dados, em virtude do momento de pandemia, não foi possível criar um ambiente para aquisição dos sinais áudios de forma controlada, pois não era adequado requisitar a presença física dos locutores. O tempo adicional demandado para essa tarefa e para aprofundamento dos conhecimentos na área de processamento digital de sinais de voz e o impedimento para acesso físico ao laboratório, impossibilitou o desenvolvimento do módulo de hardware e sua conexão com o software.

7.2 Trabalhos Futuros

Futuramente, pode-se ampliar a base de dados em busca de melhores resultados. Também vale investir tempo na construção do hardware que trabalhe em sincronia com o software e que consiga realizar o processo de aquisição e processamento dos áudios nele mesmo (uso de outro componente, a exemplo de um ESP-32 que é um sistema em um chip, com microcontrolador integrado, Wi-Fi e Bluetooth, em substituição ao Arduino), removendo assim, a dependência de um computador para a fase de teste. Pode-se, também, investir em testes com outros tipos de classificadores, como SVM e ainda unir outras formas de extração de características como MFCC invertidos e Delta MFCC.

[3] https://github.com/gabriellmd/Verificador-de-Locutor

8. REFERÊNCIAS

[1] Alcaim, A. e Solewicz, A. J. e Moraes, A. J. Frequência de Ocorrência dos Fones e Listas de Frases Foneticamente Balanceadas no Português Falado no Rio de Janeiro. Journal of Communication and Information Systems. 7, 1 (Jun. 2015).

[2] Bharadwaj, N. N. and Shayana, P. KA. and Sreesha, S. S. KV. and Bhargavi, K. Recognition of Speaker Using Vector Quantization and MFCC. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCRACES - Vol. 7, Ed. 10 (2019).

[3] Bharti, R. and Bansal, P. Real Time Speaker Recognition System using MFCC and Vector Quantization Technique. (2015).

[4] CAMPBELL, J. P. Speaker Recognition: A Tutorial. Proceedings of the IEEE, v. 85, n. 9, p. 1437-1462 (Set. 1997).

[5] Clusterização de Dados. https://www.maxwell.vrac.puc-rio.br/24787/24787_5.PDF.

[6] de Leon, G. M., de la Rosa Vargas, J. I., and Dominguez, E. G. Application of an Annular/Sphere Search Algorithm for Speaker Recognition. In 15th International Conference on Electronics, Communications and Computers (CONIELECOMP'05), pages 190–194 (2005).

[7] Gordillo, C. D. A. e Alcaim, A. Reconhecimento de Voz Contínua Combinando os Atributos MFCC e PNCC com Métodos de Robustez SS, WD, MAP e FRN. Rio de Janeiro : PUC–Rio, Departamento de Engenharia Elétrica (2013).

[8] Introdução Básica a Clusterização. https://lamfo-unb.github.io/2017/10/05/Introducao_basica_a_clusterizacao/.

[9] K-Means Clustering with Scikit-Learn. https://towardsdatascience.com/k-means-clustering-with-scikit-learn-6b47a369a83c.

[10] KNN and Kmeans. https://harshkr21august.medium.com/knn-and-kmeans-b741dfccb69.

[11] Melo, Fabrício Gutemberg Lélis de. Avaliação do uso de coeficientes mel-cepstrais na representação das características vocais de um locutor. Campina Grande - PB (2014).

[12] Rabiner, L and Juang, B-H. Fundamentals of Speech Recognition. Prentice-Hall, Inc., USA (2008).

[13] Rabiner, L. and Schafer R. Theory and Applications of Digital Speech Processing (1st. ed.). Prentice Hall Press, USA (2010).

[14] Ramos-Lara, R. and López-García, M. and Cantó-Navarro, E. et al. Real-Time Speaker Verification System Implemented on Reconfigurable Hardware. J Sign Process Syst 71, 89–103 (2013).

[15] Schueler, C. F. e Silveira, Filipe Moreira da. Desenvolvimento de um sistema de verificação de locutor, usando modelos ocultos de Markov, unindo a técnica MFCC com parâmetros extraídos do sinal glotal. Niterói-RJ (2017).

[16] Simulador de trava para porta usando teclado para senha. https://www.tinkercad.com/things/dxHLPX56Fjs-porta-com-senha.

[17] Singh, S. and Rajan, E. Vector Quantization Approach for Speaker Recognition using MFCC and Inverted MFCC.

[18] Singh, S. and Rajan, E. Vector Quantization Approach for Speaker Recognition using MFCC and Inverted MFCC. International Journal of Computer Applications. 17. 10.5120/2188-2774 (2011).

[19] Suksri, S. and Yingthawornsuk, T. Speech Recognition using MFCC (2012).

Sobre os autores:

Gabriel Almeida Azevedo. Graduando em Ciência da Computação com 4 anos de experiência em Desenvolvimento Web. Participou por 10 meses do projeto ePol (Sistema de Apoio à Investigação de Polícia Judiciária) como *FullStack*. Atualmente é desenvolvedor da Synchro Soluções Fiscais em parceria com o LSI (Laboratório de Sistemas e Informação) da UFCG.

Joseana Macêdo Fechine Régis de Araújo. Professora orientadora do TCC.