ALGORITMO DE RECONHECIMENTO DE VOZ PARA A INSERÇÃO EM SISTEMAS EMBARCADOS

VOICE RECOGNITION ALGORITHM FOR INCLUSION IN EMBEDDED SYSTEMS

Thiago Lechner Ribeiro da Silva[1]

Abstract *— The speech recognition technology is been increasingly used in electronic systems and devices, in several applications. This mainly occurs due to advances in technology such as digital signal processing, microprocessors and the appearance of recognition algorithms more and more effective. This paper presents a recognition system of voice commands to drive a DC motor, whose algorithm was developed in MATLAB and tested in SIMULINK environment. The algorithm analyzes the speech signal characteristics compared to the reference signals by cross-correlation between them, to determine which command was performed. The developed system has only two commands: "activate engine" that results in a logic 1 used to drive the motor, and "end" that results in a logic level 0 to turn off the engine, by SIMULINK. Although simple, the system achieved a good set of performance.*

Index Terms — Cross-correlation, speech recognition, trigger engine.

INTRODUÇÃO

Sistemas que utilizam o reconhecimento de voz têm se tornado mais comuns no cotidiano, estando presentes em celulares, notebooks, *tablets* e até mesmo em brinquedos, trazendo diversos benefícios aos usuários como praticidade, comodidade e diminuição de trabalhos dispendiosos. Isto tem sido possível graças a melhorias nos algoritmos de processamento digital de sinais e de voz em particular, bem como o desenvolvimento de processadores específicos [1][2].

Este artigo apresenta um sistema que permite ao usuário acionar um motor por meio de comandos de voz. Solicita-se inicialmente ao usuário que escolha duas palavras que serão utilizadas como comando de acionamento e desligamento do motor, efetuando assim a configuração inicial das referências que serão utilizadas como comparação com o comando a ser efetuado. Após este processo, o sistema estará apto a ser utilizado, necessitando apenas que o usuário diga qual comando deverá ser executado. O sistema foi desenvolvido baseando-se principalmente em técnicas descritas em [1][3][4][5].

Após reconhecer o comando solicitado, é gerado um sinal de referência que será utilizado pelo Simulink para acionar um motor DC. No ambiente do Simulink, o sinal de referência é analisado por uma condição lógica que, de acordo com o comando, irá possibilitar o acionamento do motor.

Processo de Reconhecimento

O processo de funcionamento do sistema de reconhecimento de voz está representado no diagrama em blocos da figura 1.

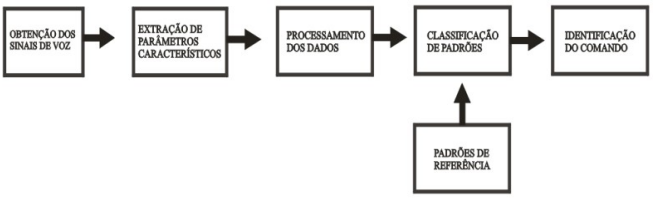

FIGURA 1
DIAGRAMA EM BLOCOS DO FUNCIONAMENTO DO ALGORITMO[5].

Obtenção dos Sinais de Voz

São obtidos inicialmente dois sinais que servirão como padrões de referência para a etapa de comparação com o comando efetuado pelo usuário. Estes sinais são obtidos através de um microfone por um período de dois segundos.

As frequências de amostragem de 10 kHz, 14 kHz e 16 kHz têm sido utilizadas em sistemas que necessitam de uma boa qualidade de voz. Essas frequências possibilitam uma melhor resolução de tempo e frequência do sinal [1]. Neste artigo utilizou-se a frequência de amostragem de 16 kHz. Desta forma, a largura de um sinal armazenado durante dois segundos será de 32000 unidades de tempo.

Extração dos Parâmetros Característicos

Para realizar a análise dos sinais, é necessário transformá-los do domínio do tempo para o domínio da frequência, o que pode ser feito com a utilização da Transformada Discreta de Fourier (DFT – *Discrete Fourier Transform*), conforme (1) [6][7].

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{-(j\frac{2\Pi NK}{n})} \qquad (1)$$

[1] Thiago Lechner Ribeiro da Silva, aluno do curso de mestrado em Engenharia Elétrica da Universidade Federal do Mato Grosso do Sul, tlrs01@hotmail.com

DOI 10.14684/ICECE.9.2015.183-187
© 2015 COPEC

Entretanto, o algoritmo da DFT não costuma ser utilizado, pois existe outro algoritmo que obtém o mesmo resultado em um período de tempo inferior, sendo este chamado de Transformada Rápida de Fourier (FFT – *Fast Fourier Transform*).

A FFT pode ser efetuada através de diversos algoritmos, sendo o mais comum o radix-2 de dizimação no tempo. Este algoritmo consiste em dividir a DFT em duas metades de índice par e índice ímpar amostrados no tempo. As saídas dessas FFTs são utilizadas para calcular diversas saídas, reduzindo assim o custo computacional. Este algoritmo pode ser representado por (2) [4].

$$x(k) = \sum_{n=0}^{\frac{N}{2}-1} x(2n) e^{-(j\frac{2\pi(2n)k}{N})} + \sum_{n=0}^{\frac{N}{2}-1} x(2n+1) e^{-(j\frac{2\pi(2n)k}{N})} * e^{-(j\frac{2\pi k}{N})} \quad (2)$$

Como o conteúdo espectral da voz varia ao longo do tempo, pois o sinal de voz não é estacionário, aplicar a DFT ao longo de uma grande janela de tempo não revela as transições no conteúdo espectral. Para resolver este problema, aplica-se a DFT sobre pequenos intervalos de tempo, sendo este método denominado Transformada de Fourier de Tempo Reduzido (STFT – *Short Time Fourier Transform*).

A STFT necessita de uma análise local para a representação do tempo e da frequência. Para isso, utiliza-se uma janela para analisar todos os segmentos do sinal, selecionando cada parte em pequenos intervalos de tempo. Este processo pode ser visto na figura 2.

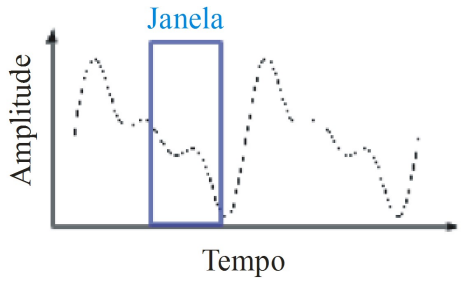

FIGURA 2
APLICAÇÃO DA FUNÇÃO JANELA EM UM SINAL.

A STFT de tempo discreto pode ser obtida com a aplicação de (3), que mapeia o sinal em uma função bidimensional do tempo e da frequência.

$$x(m,\omega) = \sum_{n=-\infty}^{\infty} x[n] * w[n-m] * e^{-j\omega n} \quad (3)$$

Sendo:

- m: parâmetro de tempo;
- ω: parâmetro de frequência;
- x[n]: sinal a ser analisado;
- w[n-m]: função de janela centralizada no tempo m.

A STFT representa um tipo de compromisso entre o tempo e a frequência. Ela fornece informações sobre quando um sinal ocorre e qual frequência ele possui. Entretanto, esta informação possui uma precisão limitada, sendo determinada pelo tamanho da janela a ser utilizada [1].

É possível obter a STFT através dos seguintes passos:

1. Escolher uma função de janela de largura finita;
2. Posicionar a janela em t = 0;
3. Truncar o sinal utilizando a janela;
4. Calcular a Transformada de Fourier do sinal truncado salvando os resultados;
5. Deslocar a janela para a direita;
6. Voltar ao passo 3 até que a janela chegue ao final do sinal.

Para a aplicação da STFT no MATLAB utilizou-se a função *specgram* que recebe como entrada o sinal a ser analisado, a largura da FFT, a frequência de amostragem, o valor das sobreposições e o tipo de janela a ser utilizada.

Para o desenvolvimento deste artigo utilizou-se a Janela de Hanning (4).

$$w[n] = 0.5 - 0.5 * \cos(\frac{2\pi n}{N-1}) \quad (4)$$

Para n = 0, 1, 2,...,N-1.

Esta janela possui um formato similar ao de uma onda cossenoidal de meio ciclo que pode ser vista na figura 3 [8].

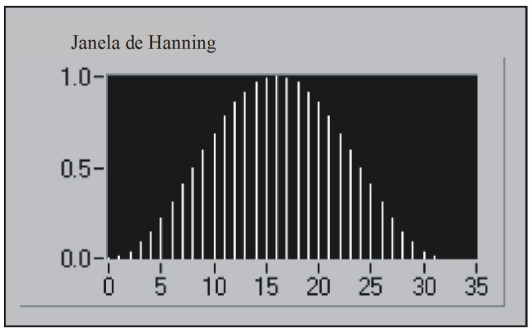

FIGURA 3
APLICAÇÃO DA JANELA DE HANNING EM UM SINAL.

Com a utilização da janela de Hanning, (3) pode ser escrita conforme (5).

$$x(m,\omega) = \sum_{n=-\infty}^{\infty} x[n] * w[0.5 - 0.5 * \cos(\frac{2\pi n}{N-1}) - m] * e^{-j\omega n} \quad (5)$$

Desta maneira obtém-se uma matriz que contém estimativas do conteúdo da frequência em função do tempo para cada sinal de voz. Nesta matriz, a frequência aumenta linearmente através das linhas, iniciando-se em 0 na primeira

linha até o valor máximo na última. Já o tempo aumenta através das colunas, da esquerda para a direita [1].

Processamento dos Dados

Na etapa atual do processo, tem-se uma matriz *s* com informações sobre o sinal de voz no domínio da frequência X(ω).

Efetua-se então a transposta desta matriz, obtendo-se a matriz *s'*. Em seguida, verificam-se quais os valores absolutos desta matriz, fazendo com que todos os valores se tornem reais. Somam-se as linhas desta matriz, sendo que estas possuem todos os componentes de frequência, e obtém-se um vetor que representa o espectro de frequências do sinal, representado por |X(ω)|.

Quando se compara a diferença entre dois sinais de voz distintos, é difícil obter resultados válidos para sinais que possuem diferentes padrões de medida. Sendo assim, utiliza-se a normalização para obter um mesmo padrão de medida. A normalização do sinal pode ser obtida através de (6).

$$y = \frac{(x - valor_mínimo)}{(valor_máximo - valor_mínimo)} \qquad (6)$$

Após a normalização, o vetor |X(ω)| é ajustado em um intervalo de valores entre [0, 1], pois esta operação modifica a amplitude dos sinais e mantém o formato da informação do espectro [1].

Classificação de Padrões

Nesta etapa, tem-se um vetor com dados normalizados que possui a informação do espectro de frequências do sinal.

Efetua-se a correlação cruzada entre o comando dito pelo usuário e cada palavra de referência, conforme (7).

$$r_{xy} = \sum_{n=-\infty}^{\infty} x(n) * y(n) \qquad (7)$$

A função de correlação cruzada é um método útil para estimar os parâmetros deslocados de um sinal em relação a outro.

Quando dois sinais não possuem tempo deslocado, sua correlação cruzada deve ser máxima.

Quanto mais correlacionados estiverem dois sinais, maior será a simetria do gráfico de correlação entre eles.

Identificação do Comando

Tendo as informações relativas às correlações cruzadas entre o comando dito pelo usuário e cada palavra de referência, deve-se verificar com qual referência o comando possui maior semelhança. Para isto, na etapa de identificação do comando, verifica-se qual o menor erro quadrático médio proveniente das correlações. Aquele que for o menor irá indicar qual palavra foi dita pelo usuário [1][4][5].

RESULTADOS OBTIDOS

Para a análise dos resultados, foram escolhidas como referências as palavras "acionar motor" e "fim". A figura 4 apresenta o gráfico do espectro de frequência da palavra "acionar motor".

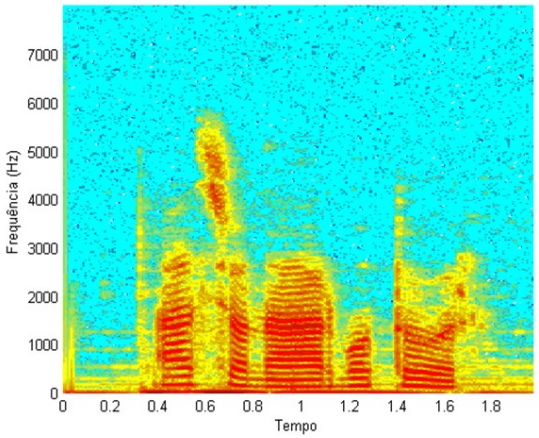

FIGURA. 4
ESPECTRO DE FREQUÊNCIA DA REFERÊNCIA "ACIONAR MOTOR".

O espectro de frequência da palavra "fim" pode ser visto na figura 5.

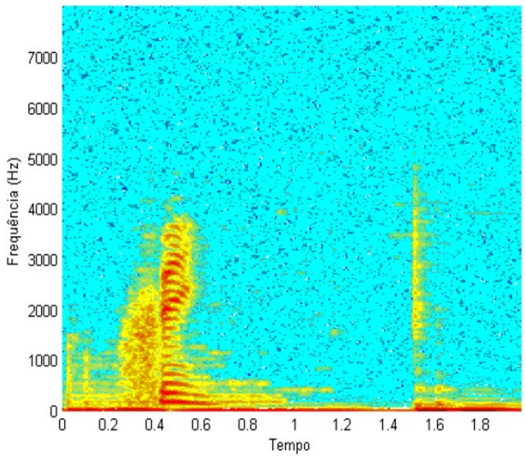

FIGURA. 5
ESPECTRO DE FREQUÊNCIA DA REFERÊNCIA "FIM".

Verifica-se uma grande diferença espectral entre as palavras de referência, fato que contribui para uma melhor taxa de acerto do comando a ser efetuado.

Utilizou-se o comando "acionar motor" para testar o programa.

A figura 6 apresenta uma comparação entre a palavra de referência "acionar motor" e o comando.

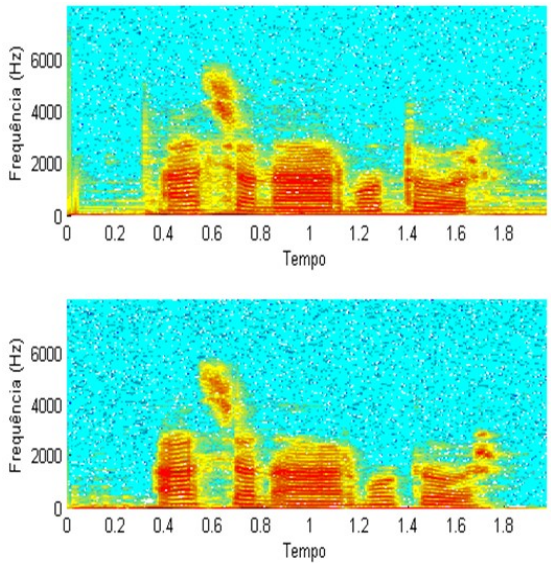

FIGURA.6

ESPECTRO DE FREQUÊNCIA DA REFERÊNCIA "FIM".

Já a figura 7 apresenta uma comparação entre a palavra de referência "fim" e o comando efetuado.

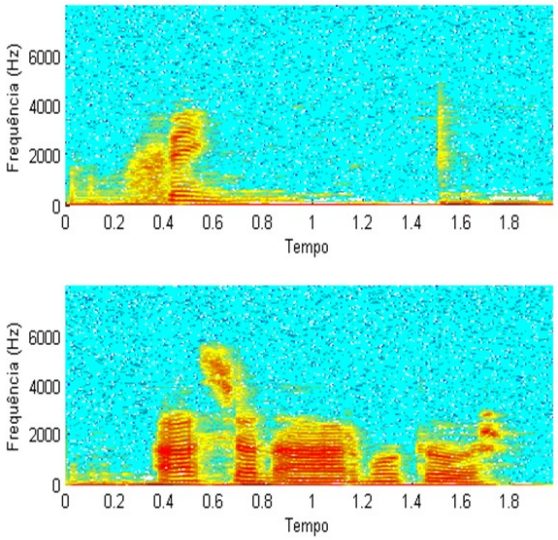

FIGURA. 7

ESPECTRO DE FREQUÊNCIA DA REFERÊNCIA "FIM".

Analisando as figuras, verifica-se uma grande semelhança entre os espectros de frequência da figura 6 e uma diferença drástica entre os espectros da figura 7. Está análise resultou no erro quadrático médio apresentado na tabela I.

TABELA I

ERRO QUADRÁTICO MÉDIO DE COMPARAÇÃO ENTRE O COMANDO "ACIONAR MOTOR" E AS PALAVRAS DE REFERÊNCIA.

Comando	acionar motor	
Palavras de referência	acionar motor	fim
erro quadrático médio da correlação cruzada	0,0707	0,2393

Como o erro proveniente da comparação entre o comando e a palavra de referência de acionamento do motor foi menor, o motor foi acionado pelo sistema.

A figura 8 apresenta o circuito de acionamento do motor desenvolvido no software Simulink.

De acordo com a palavra reconhecida na etapa de comparação dos sinais, a variável X recebe um determinado valor, sendo 1 para "acionar motor" e 0 para "fim". Este sinal é recebido no ambiente do Simulink com a utilização do bloco *Signal From Workspace*, possibilitando então que este seja analisado através do bloco *If Action Subsystem* e ocorra uma tomada de decisão de acionamento ou desligamento do motor.

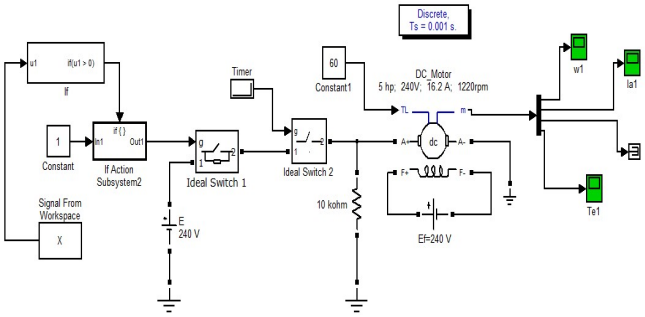

FIGURA 8

CIRCUITO DE ACIONAMENTO DO MOTOR DC.

O bloco *Ideal Switch 1*, caso receba o valor 1 na entrada g, irá fornecer na sua saída o sinal contido na entrada 1. Já o bloco *Ideal Switch 2* está configurado para fornecer à sua saída o sinal que estiver na entrada 1 após um período de 10 segundos, sendo este atraso definido pelo bloco *Timer* conectado à entrada g. Esta temporização foi efetuado com a finalidade de obter uma melhor análise gráfica da simulação.

Para o funcionamento do motor DC, utilizou-se como referência uma tensão de entrada de 240 V. Em sua saída é apresentados o valor de torque.

Considerou-se um período de simulação de 20 segundos para verificar graficamente o torque de saída do motor DC.

A figura 9 apresenta o torque de saída do motor.

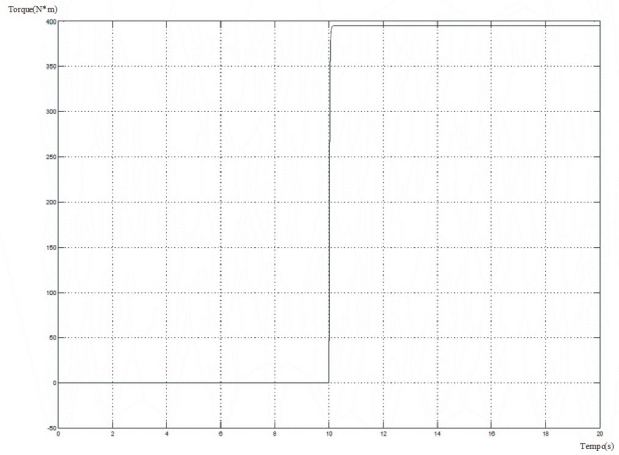

FIGURA 9
TORQUE DE SAÍDA DO MOTOR AO SER ACIONADO.

Verifica-se que após a constatação de que o comando efetuado pelo usuário foi o de acionamento do motor, o sinal de acionamento é enviado do ambiente MATLAB para o Simulink e dá-se início ao processo de acionamento do circuito. Após um atraso de 10 segundos atribuído pelo bloco *Timer*, o motor é acionado alcançando um torque de 400 Nm.

CONCLUSÃO

O sistema de reconhecimento de voz utilizado neste artigo empregou um dos métodos mais simples de reconhecimento da atualidade (correlação cruzada). O projeto obteve êxito, porém apresentou uma limitação, pois deve ser utilizado com uma baixa quantidade de comandos para garantir uma elevada diferença espectral entre os sinais e uma boa taxa de reconhecimento do comando correto.

Verificou-se que quanto maior a diferença entre as palavras utilizadas, melhor é o desempenho do sistema, sendo este o motivo pelo qual as palavras utilizadas para o teste do sistema foram "acionar motor" e "fim".

REFERÊNCIAS

[1] T. Yang, The Algorithms of Speech Recognition, Programming and Simulating in MATLAB, Janeiro, 2012. (TCC) University of Gaule, Faculty of Engineering and sustainable development, 2012.

[2] J. Picone, Signal Modeling Techniques In Speech, Systems and Information Sciences Laboratory, Tsukuba Research and Development Center, Tsukuba,Japan.

[3] L. R. Rabiner, R. W. Schafer. *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice Hall, 1978.

[4] X. Huang, A. Acero, HW. Hon, *Spoken Language processing – A Guide to Theory, Algorithm, and System Development*, Prentice Hall PTR, 2001.

[5] L. R. Rabiner, B.H. Juang. *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice Hall, 1993.

[6] S.S. Haykin, V.B.Veen. *Sinais e Sistemas*. Porto Alegre: Bookman, 2000.

[7] A. Oppenheim, R. W. Schafer. *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[8] H. Beigi. *Fundamentals of Speaker Recognition*. Springer, 2011.