UNIVERSIDADE FEDERAL DO CEARÁ - UFC
CENTRO DE TECNOLOGIA - CT
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA – DETI

FRANCISCO EUGÊNIO DE FREITAS

RECONHECIMENTO DE VOZ UTILIZANDO TRANSFORMADA WAVELET E CODIFICAÇÃO PREDITIVA LINEAR

FORTALEZA
2010

FRANCISCO EUGÊNIO DE FREITAS

RECONHECIMENTO DE VOZ UTILIZANDO TRANSFORMADA WAVELET E CODIFICAÇÃO PREDITIVA LINEAR

Monografia apresentada à disciplina de Projeto de Final de Curso do Curso de Engenharia de Teleinformática da Universidade Federal de Ceará.

Orientador: Prof. Carlos Pimentel de Sousa

FORTALEZA

2010

TERMO DE APROVAÇÃO

FRANCISCO EUGÊNIO DE FREITAS

RECONHECIMENTO DE VOZ UTILIZANDO TRANSFORMADA WAVELET E CODIFICAÇÃO PREDITIVA LINEAR

Monografia apresentada ao Programa de Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará, como requisito para obtenção do título de Engenheiro de Teleinformática.

Aprovado em __/__/____, com nota ___

_____
Prof. Carlos Pimentel de Sousa

Banca Examinadora:
Primeiro Examinador: _____
Segundo Examinador: _____
Terceiro Examinador: _____

FORTALEZA

2010

Dedico este trabalho aos meus familiares, em especial a minha esposa, e a todos os que contam com o meu sucesso.

RESUMO

A implementação de sistemas capazes de identificar comandos por voz possui uma grande quantidade de aplicações. Por este motivo, *softwares* de medição das características físicas (biométricas) são produzidos em grande escala.

A grande dificuldade do reconhecimento de voz é a sua natureza interdisciplinar. Além disso, as variabilidades acústicas do locutor e entre locutores estão relacionadas com o problema.

Este trabalho expõe os conceitos e algoritmos tradicionais utilizados para reconhecimento de voz analisando cada fase pertinente a todo este processo. Definimos o sinal de voz e estudamos suas características junto ao processo de fala. Analisamos diversas formas de reconhecimento biométrico analisando as vantagens e desvantagens de algumas abordagens e estudamos todo o processo computacional desde a aquisição dos dados a partir de um microfone, por exemplo, até a matemática e programação utilizada para cada um dos algoritmos.

Exibimos também melhorias computacionais implementadas que proporcionaram um aumento de 15% na taxa média de acerto em um dos algoritmos utilizados.

O objetivo deste trabalho é demonstrar algo que fez parte dos sonhos dos produtores de filmes de ficção científica das décadas de 70 e 80, o reconhecimento de voz pelas máquinas.

ABSTRACT

The implementation of systems capable to identify voice commands has a great amount of applications. For this reason, *softwares* for measure physical characteristics (biometrics) are produced in large-scale.

The great difficulty of the voice recognition is its interdisciplinary nature. Moreover, the acoustic variations of the speaker and between speakers are related with the problem.

This work presents the concepts and traditional algorithms used for speech recognition by analyzing each relevant stage of this whole process. Defined the voice signal and studied its characteristics in the process of talks. We considered many forms of biometric recognition analyzing the advantages and disadvantages of some approaches. And we studied the whole computational process since the acquisition of data from a microphone, for example, until the mathematics and programming used for each of the algorithms used.

The objective of this work is to demonstrate something that was part of the dreams of the scientific fiction film producers of the decades of 70 and 80, the voice recognition by the machines.

LISTA DE FIGURAS

Sumário

1. Introdução

A acessibilidade é um meio de quebrar as barreiras e promover a inclusão de pessoas com necessidades especiais. Uma das mais eficientes formas de se prover acessibilidade nos dias atuais é o desenvolvimento de sistemas que automatizam e facilitam a realização das diversas tarefas cotidianas, por mais simples que sejam.

Como exemplo do que pode ser posto em prática, há a implementação de um equipamento que reconheça palavras faladas e execute determinadas ações a partir dos comandos verbais recebidos. Embora essa idéia não seja nova, há uma carência de soluções desse tipo para que pessoas com deficiência motora, por exemplo, consigam ter maior independência e possam levar uma vida mais confortável.

De uma maneira geral, pode-se dizer que o reconhecimento de comandos de voz por uma máquina requer noções de diversas áreas do conhecimento, pois ele por si só possui uma natureza multidisciplinar. A produção e percepção da fala pelo ser humano é parte do estudo da Fisiologia e a Lingüística, por sua vez, é a responsável por estudar a associação dos fonemas às palavras e os significados conferidos às mesmas. No campo da Física se destaca a acústica, enquanto que nas áreas de engenharia e computação temos o processamento de sinais, o reconhecimento de padrões e o desenvolvimento de programas e equipamentos que implementem a solução desejada.

Ao realizar uma análise mais cuidadosa do que vem a ser o reconhecimento de voz, percebe-se que ele pode ser utilizado tanto para o reconhecimento do locutor quanto o de palavras.

O desenvolvimento de um sistema que implemente o reconhecimento do locutor visa atender a pelo menos um desses objetivos principais: identificação e autenticação. Para a identificação, comparam-se os dados provenientes da análise de voz da pessoa alvo com o de uma base de usuários, buscando o maior grau de semelhança e assim determinando a identidade do locutor. Já na fase de autenticação, é feita a verificação se o processamento da voz da pessoa é compatível com o padrão armazenado para a identidade que ela afirma possuir e então é decidido, dentro de um grau de precisão pré-determinado, se o locutor é ou não aquele determinado usuário. Uma possível terceira abordagem seria unir essas duas implementações e, a partir da coleta e processamento da voz, fazer a identificação do locutor dentre todos os usuários e verificar se ele possui ou não permissão de acesso ao sistema.

Por outro lado, no reconhecimento de palavras, os objetivos mudam, variando da simples tradução de poucas palavras ou frases curtas em ações programadas dentro do sistema até a compreensão individual de cada fonema pronunciado e o seu agrupamento em palavras. No primeiro caso citado se espera que o sistema compreenda a linguagem do locutor, podendo a partir daí converter a fala em texto ou passar a interagir com o usuário, caso possua inteligência artificial. Já o segundo caso é justamente o que este trabalho se propõe a fazer, que é verificar se a palavra pronunciada está contida na lista de comandos cadastrados e, caso positivo, executar a ação correspondente.

O trabalho que se segue tem como objetivo comparar duas técnicas de reconhecimento de voz no intuito de identificar melhorias e avançar no desenvolvimento de uma área da tecnologia que, apesar do tempo de estudo desprendido para ela, ainda tem muito a evoluir.

2. Biometria

Biometria se origina do grego: *bio*(vida) e *metron*(medida). Assim, biometria é o mapeamento de traços pessoais para identificadores biométricos únicos que sejam difíceis de compartilhar, alterar e/ou forjar. Esses identificadores podem ser voz, imagem facial, impressões digitais, geometria da mão, íris, retina, assinatura manual recolhida digitalmente, dinâmica de digitação, entre outros.

Os métodos biométricos são classificados como: comportamentais, como o reconhecimento por voz e pelas ondas cerebrais, e físicos, como o reconhecimento pela impressão digital e pela facial. A maior dificuldade está no reconhecimento através de identificadores comportamentais porque, ao contrário das características físicas, variam de acordo com o estado emocional da pessoa.

O mais importante parâmetro para avaliação de um sistema biométrico é o grau de fiabilidade que, por definição, é a qualidade de algo que é extremamente confiável. Uma das abordagens para se aferir de maneira ótima este parâmetro está em encontrar um ponto de equilíbrio, normalmente chamado de CER (*Crossover Error Rate* – Taxa de intersecção de erros) **(Figura 1)**, entre os valores FAR (*False Acceptance Rate* – Taxa de falsas aceitações) e o FRR (*False Rejection Rate* – Taxa de Falsas Rejeições) em um algoritmo de reconhecimento uma vez que estes valores são independentes entre si.

Todo sistema de reconhecimento de voz deve ter como um dos produtos de saída para análise de seu desempenho um gráfico de confronto entre os valores FRR e FAR. Quanto mais baixo for o ponto de intersecção das curvas FRR e FAR e menor for o limiar, maior é a precisão do sistema biométrico. Analisar esta precisão é bastante difícil, pois conta com uma grande quantidade de dados e a forma

complexa com que estes são obtidos.

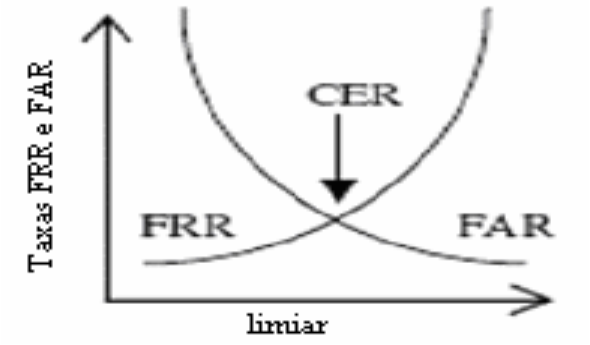

Figura 1 - *Crossover Error Rate* – CER

Existem outros parâmetros como o de aceitação e acomodação que visa ser o menos intrusivo possível para os usuários e também o custo a ser desenvolvido deve ser observado para a determinada aplicação.

Muitas empresas e entidades governamentais utilizam métodos biométricos para segurança na autenticação. Abaixo segue os métodos biométricos mais utilizados, bem como explicações conceituais e observações quanto à viabilidade técnica e financeira em alguns casos.

2.1 Tecnologias de identificação biométrica.

Existem várias tecnologias de identificação biométrica. Listaremos as mais utilizadas no mercado e, portanto, as que estão em estágio mais avançado de amadurecimento quanto a sua eficácia.

2.1.1 - Impressão digital

A impressão digital é formada por vários sulcos, que na sua formação possuem as chamadas papilas ou pontos de minúcias. As papilas são formadas

ainda enquanto feto e acompanham o ser por toda a sua vida. São usadas a mais de cem anos para identificação pessoal e, hoje, sabemos, que elas são únicas, diferindo até gêmeos univitelinos. Alguns indivíduos, que possuem a chamada Síndrome de Nagali, nascem com as pontas dos dedos lisas.

Esta tecnologia biométrica é o método mais utilizado em todo o mundo correspondendo, segundo a Associação Brasileira das Empresas de Sistemas Eletrônicos de Segurança, a aproximadamente 50% do mercado em produtos deste mesmo gênero. Por ser um método de reconhecimento físico, não considera fatores culturais como o idioma, por exemplo. Além disso, é mais barata que as demais técnicas e, também, bastante segura.

O primeiro sistema, que se tem noticia, de identificação por impressões digitais foi criado na Argentina, no século XIX, pelo croata *Juan Vucetich.*

2.1.2 - Reconhecimento da íris

A íris é a parte mais visível dos olhos, composta pelos anéis coloridos que rodeiam a pupila e é a menos intrusiva das tecnologias de reconhecimento biométrico que envolve os olhos, logo, é a mais aceita pelos usuários.

Este método é estudado desde a década de 60. Iniciou com o cientista *John Daugman*, da Universidade de Cambridge. Segundo estudos de *Daugman*, é um método seis vezes mais seguro que o reconhecimento da impressão digital e um dos que possuem o mais baixo custo para implantação uma vez que qualquer câmera pode ser adaptada para este método, embora a qualidade da imagem deva ser levada em consideração para a precisão do método.

Este método faz a leitura dos padrões de cor da região colorida dos olhos

apoiando-se no fato de este ser o único padrão individual que permanece inalterado por toda a vida do indivíduo. Está entre os mais seguros do mundo e é muito utilizado para sistemas de segurança por ser pouco intrusiva. No entanto, a dificuldade existente para integrar este método com os outros sistemas é um obstáculo para sua utilização.

2.1.3 - Reconhecimento da retina

A retina do organismo humano é uma parte extremamente estável. Compartilha seu espaço com estruturas como o cérebro e nunca entra em contato direto com o meio, o que torna toda a região que a envolve pouco suscetível a mudanças. Apenas a exposição demasiada à luz pode causar mudanças como a catarata, por exemplo, que prejudica o desempenho dos sistemas de reconhecimento de retina, como o fato da rápida deterioração da retina quando é retirada do local de origem do corpo humano.

Por estes motivos é um modo de reconhecimento biométrico bastante confiável, sendo difícil um individuo se passar por outro. Assim, em 1975, a idéia de se construir um aparato simples capaz de realizar a leitura da retina para a identificação humana foi adotada.

No entanto, este método é muito pouco utilizado por ser bastante intrusivo e de alto custo de produção.

2.1.4 - Reconhecimento facial

Os sistemas de reconhecimento facial são um dos mais utilizados no

mundo. Isto se deve ao fato de ser um dos métodos menos intrusivos diante das opções existentes no mercado. Através de uma serie de fotografias, uma câmera consegue capturar a imagem facial de um individuo e, com um mapeamento de dados do relevo facial, é possível fazer o reconhecimento sem a necessidade de fornecer dados pessoais ao sistema como a impressão digital ou o reconhecimento através da íris ou retina.

O grande uso também se deve ao fato de que o dispositivo de captura ser de fácil aquisição. Qualquer câmera digital pode ser adaptada para ser utilizada nestes sistemas.

Grandes estabelecimentos internacionais possuem, em seu banco de dados, os dados faciais de celebridades para que seu reconhecimento seja feito e a segurança pessoal seja reforçada.

O uso de óculos ou apetrechos artísticos pode dificultar o reconhecimento.

2.1.5 - Assinatura digital

A assinatura digital é um método de reconhecimento biométrico onde as características da assinatura do indivíduo são analisadas.

Esta análise não se limita apenas à forma da escrita do indivíduo, mas também na pressão exercida por ele ao escrever, na velocidade de escrita de cada letra e palavra e na rapidez de composição das letras. Este sistema é usado, principalmente, em sistemas de reconhecimento de assinatura de cheques e transferências bancárias.

É uma tecnologia de muito baixo custo e boa precisão, no entanto, não é muito utilizada por conta do seu alto grau de intrusividade.

2.1.6 - Geometria da mão

Esta tecnologia biométrica faz uso de sensores ópticos para capturarem características das mãos dos usuários.

É capturada a imagem a partir de um scanner e, assim, feita a analise da forma da mão, comprimento dos dedos e espaçamento entre eles e análise da localização das linhas caracterizadoras das mãos e dedos.

É um sistema de implementação simples, baixos custos e alta velocidade de resposta porque a quantidade de dados capturados e armazenados é muito pequeno em comparação com outras técnicas de reconhecimento biométrico. No entanto sua precisão é muito baixa para sistemas de segurança críticos. Por estes motivos é, muitas vezes, utilizada combinada com outras técnicas de reconhecimento como a leitura da impressão digital.

2.1.7 - Reconhecimento de voz

De todos os métodos listados, este é o único comportamental e, segundo o IEEE *Computer Society*, este é o método de reconhecimento biométrico que possui um longo futuro pela frente, sendo, provavelmente, o sucessor do mais popular método de reconhecimento biométrico hoje, a leitura das impressões digitais.

É baseado na captação da voz de um indivíduo e transformação dessa onda sonora em dados. Para isso é utilizado um microfone e um conversor A/D (analógico/digital) já existente na maioria dos computadores atuais. A seguir vem a comparação com dados armazenados em um banco de dados como entonação, amplitude, dispersão na freqüência, etc. Mais à frente detalharemos cada um

desses estágios de desenvolvimento.

Mesmo sendo tão simples o seu funcionamento, conceitualmente é bem complexo. O ambiente ao redor do usuário influencia bastante na precisão deste método, uma vez que o microfone capta também o som ambiente.

Muito são os motivos que nos leva a acreditar que este método biométrico tem uma promissora evolução, entre eles estão:

- é o menos intrusivo de todos os outros métodos abordados anteriormente;

- sua implantação é de baixíssimo custo, necessitando apenas de um microfone para a aquisição dos dados;

- o grande interesse das áreas de inteligência computacional de evoluir algoritmos nesta área para estudos.

Atualmente, sua aplicação é bastante limitada devido à dificuldade em trabalhar com dados comportamentais, ou seja, que variam com o tempo, espaço e estado emocional do indivíduo. Assim, seu nível de precisão ainda é bastante baixo comparado com os demais sistemas de reconhecimento biométrico.

Ao longo do trabalho, detalharemos este método de reconhecimento biométrico.

2.2 - Futuro da biometria

A biometria, como todas as demais áreas de estudos, evoluiu a cada dia. Novas técnicas são pesquisadas, implementadas e utilizadas ou descartadas de acordo com seu devido desempenho.

O odor, as ondas cerebrais, o DNA e a arquitetura da orelha são algumas das novas técnicas biométricas que estão sendo estudadas hoje.

Antes de a biometria vir a se tornar popular, o reconhecimento por

impressão digital ou facial era dito como impossível ou de um futuro muito distante. Hoje, são consideradas ultrapassadas e dá espaço para o aparecimento de novas técnicas ou melhoramento de técnicas já existentes como o reconhecimento de voz.

A seguir, mostraremos como um sistema de reconhecimento de voz se comporta e como deve ser projetado.

3. O som e a voz

A mais importante fase de todo um processo de implementação de uma determinada ferramenta é o estudo dos componentes principais deste processo. É preciso saber, exatamente, como estes componentes se originam, como se comportam e como é possível alterá-los, se necessário, durante o processo.

Como trataremos de um sistema de reconhecimento de voz, é preciso conhecer as principais características das ondas sonoras e como elas são produzidas pelas cordas vocais.

3.1. O som

O som é uma onda mecânica que se propaga de forma circuncêntrica apenas em meios materiais, ou seja, não se propaga no vácuo. Ele é produzido pela existência de pressão em um meio como, por exemplo, o ar. Com a existência dessa pressão, criam-se as vibrações e a combinação dessas caracterizam o som, representadas pela soma de diversas freqüências diferentes:

$$SOM = F1 + F2 + F3 + F4 + ...+ Fn \qquad [3.1]$$

Cada termo da equação 3.1 é uma freqüência múltipla da primeira, sendo o seu conjunto conhecido como serie harmônica. O primeiro termo (F1) é chamado de fundamental ou harmônico de ordem 0 (zero) e os demais termos são os harmônicos de ordem 1 (um), 2 (dois) e assim sucessivamente.

O som é, normalmente, representado por uma forma de onda senoidal, ou seja, tendo como principais características:

- amplitude: distância do ponto mais elevado que a onda possui até a reta

de origem;

- freqüência: quantidade de ciclos que a onda é capaz de fazer em um intervalo de 1 (um) segundo;

- período: é o inverso da freqüência e representa a quantidade de tempo que a onda leva para fazer 1 (um) ciclo completo.

Um exemplo de onda senoidal possuindo uma amplitude igual a 1 (um) e freqüência igual a 1rad/s (um radiano por segundo) pode ser visto na **Figura 2.**

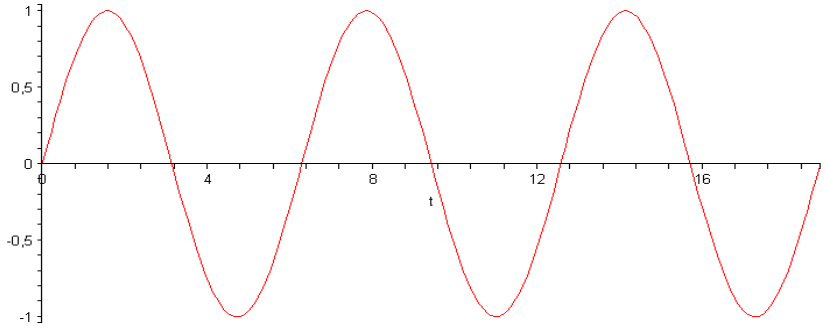

Figura 2 - Onda Senoidal

Outra característica das ondas é que elas transportam energia sem transportar matéria. Isso fica claro com a propagação da perturbação das ondas criadas quando se joga um objeto qualquer em um lago de água parada.

Muitos animais percebem os sons com o sentido da audição, o que permite saber a posição e distância da fonte do som. É a que chamamos de audição estereofônica. Para os humanos, este sentido está limitado entre 12 Hz e 20KHz, embora estes limites não sejam absolutos. Com o passar dos anos, o limite superior decresce por conta da dificuldade auditiva dos seres humanos mais idosos.

Algumas partes do corpo também sentem sons de baixa freqüência.

3.2. A voz

Desde o momento do seu nascimento, o ser humano utiliza a voz para,

principalmente, transmitir mensagens e manifestar seus sentimentos, desejos e apelos. Ainda na forma de sons inarticulados, ela é a primeira ferramenta que um recém-nascido possui para exprimir suas necessidades e, assim, tentar garantir a sua sobrevivência. Quando a criança começa a aprender a falar, ela passa a ser capaz de se expressar mais diretamente e, a partir daí, a voz será a sua principal forma de se comunicar com o mundo a seu redor.

3.2.1. Produção

O aparelho fonador, constituído por órgãos dos aparelhos digestivo e respiratório, é o responsável por produzir a voz no corpo humano. A voz humana é o som produzido pela vibração que o ar, quando sai dos pulmões, causa nas pregas vocais localizadas na laringe, sendo modificados pela língua, lábios e até dentes.

A coluna de ar que posteriormente será transformada em voz se origina nos pulmões, onde o diafragma e os músculos respiratórios pressionam o ar a ser expelido. A partir daí, passa pela laringe, onde estão localizadas as cordas vocais, responsáveis por produzir o som fundamental através das suas vibrações. A ressonância é produzida na faringe, na boca e na cavidade nasal, resultando na amplificação do som, e os lábios, língua, palato mole, palato duro e mandíbula finalizam a voz ao articular e dar sentido ao som.

Além dos órgãos do aparelho digestório e do respiratório, outros dois são bastante importantes também para uma melhor interação com a voz. O ouvido é responsável por captar, localizar e conduzir o som, enquanto que o cérebro analisa, registra e processa o som, tentando interpretar o que o ouvido capta.

3.2.2. Características

A voz que o ser humano é capaz de produzir pode variar em freqüência, intensidade, timbre, tom, ressonância, articulação, inflexão e em muitas outras características. Algumas delas serão mais bem explicadas neste capítulo.

A faixa de freqüência da voz humana pode se estender entre 20Hz e 10KHz, embora a faixa de maior energia esteja concentrada principalmente entre 100Hz e 3400Hz. Vale ressaltar que os sons mais graves possuem maior riqueza harmônica do que os muito agudos, uma vez que, como os harmônicos são múltiplos da freqüência fundamental e a fundamental de um agudo já é uma freqüência bastante alta, o indivíduo de voz muito aguda não será capaz de gerar tantos harmônicos quanto o de voz mais grave.

Já o timbre é como uma assinatura pessoal, uma vez que varia de pessoa para pessoa. Ele depende tanto de fatores fisiológicos, como a musculatura e as cavidades ósseas e nasais, quanto até mesmo do próprio temperamento de quem está falando. Na música, há diversos parâmetros que ajudam a caracterizar o timbre de um cantor, tais como volume, espessura, mordente e cor.

Para o desenvolvimento de um sistema que trabalhe com a voz humana, a faixa de freqüência é uma característica em que se deve ter muita atenção, pois ela é um parâmetro obrigatório para a construção de filtros. Se o sistema a ser desenvolvido tiver também como objetivo a identificação do interlocutor, ele trabalhará igualmente com o timbre da voz, pois é pelo timbre que é possível discernir entre as vozes de diferentes pessoas.

3.2.3. Perigos e cuidados

Devido a sua grande importância, a voz requer cuidados para evitar a deterioração ou até mesmo doenças que restrinjam a capacidade de produzi-la. Naturalmente, a voz já sofre alterações com os hormônios e até mesmo com o estado emocional da pessoa, mas outros fatores contribuem para o seu desgaste.

A fumaça produzida pelo cigarro, por exemplo, colabora para uma maior produção de muco, alterando a faringe e irritando as cordas vocais. E, para isso, não é necessário que o indivíduo seja fumante, bastando ele se encontrar em um ambiente fechado com fumaça presente. Poeira e pó de giz, quando aspirados, se depositam sobre as cordas vocais e também as irritam ao aumentar o atrito entre elas. Ambientes com ar-condicionado são igualmente prejudiciais, pois ressecam o ar e as cordas vocais e faz com que haja maior esforço na produção da voz. Nesses casos, o recomendável é manter a garganta hidratada aumentando a freqüência de ingestão de água.

Outro ponto que se deve ter cuidado é com a alimentação. Comidas e bebidas muito geladas devem ser evitadas, assim como variações de temperatura, pois podem provocar inflamações e alergias. Alimentos muito condimentados podem dificultar a digestão e atrapalhar a movimentação do diafragma. Bebidas gasosas podem provocar gases e o leite e o chocolate aumentam a produção de muco e devem ser evitados por pessoas que necessitam fazer uso constante e intenso da voz. Por outro lado, alimentos leves, frutas e verduras, quando bem mastigados, ajudam a relaxar a mandíbula e melhorar a dicção.

Algumas substâncias, como o álcool e certas pastilhas e *sprays*, dão uma aparente sensação de relaxamento muscular e alívio ao desgaste da voz, mas na verdade apenas mascaram o problema, pois atuam como anestésico e, quando seu

efeito passa, o cansaço e a deterioração vocal se tornam perceptíveis novamente.

Maus hábitos como tosse e pigarros podem provocar alterações nas cordas vocais devido ao atrito constante e brusco e por isso devem ser corrigidos. Também a postura e até mesmo a forma de se vestir podem prejudicar a produção da voz e por isso deve-se cuidar para manter uma postura mais relaxada e usar roupas mais confortáveis.

4. Sistemas de Reconhecimento de Voz

O rápido avanço das tecnologias utilizadas nas interfaces homem-máquina tem auxiliado o desenvolvimento de sistemas de reconhecimento de voz cada vez mais complexos e eficazes visando substituir sistemas tradicionais que utilizam dispositivos como teclados, painéis e, até mesmo, leituras da impressão digital na intenção, sempre, de evitar o contato físico entre usuário e sistema.

O reconhecimento automático de voz é um processo composto pela extração de características únicas contidas num sinal sonoro e seu posterior reconhecimento comparado a um outro conjunto de características previamente armazenado. Este processo de reconhecimento pode ser dividido nas seguintes etapas:

- Criação do banco de dados: gravações das elocuções que devem ser reconhecidas são armazenadas em um banco de dados.

- Digitalização do sinal de voz: etapa em que o sinal analógico é capturado, digitalizado e tratado.

- Extração de parâmetros do sinal de voz: etapa que extrai parâmetros específicos do sinal para armazenamento e comparação de padrões entre sinais.

- Reconhecimento de padrões em sinais de voz: etapa em que ocorre o processo de comparação e entre o sinal adquirido e um sinal armazenado.

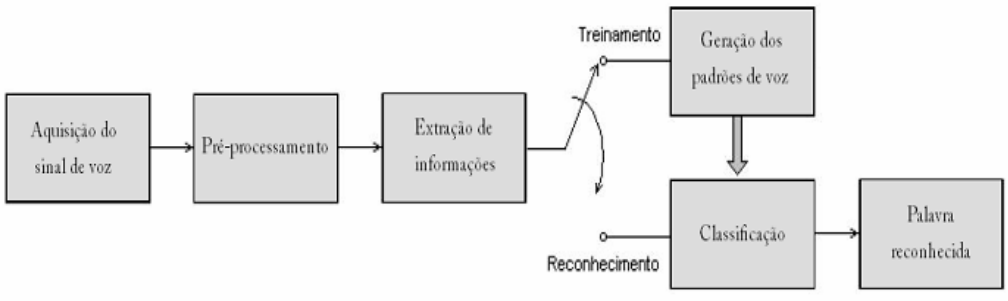

Figura 3 - Sistema de reconhecimento de voz padrão

Na **Figura 3** podemos ver o diagrama de blocos de um típico sistema de reconhecimento de voz. A fase de treinamento do sistema pode ser substituída pela fase de armazenamento do padrão de voz em uma base de dados qualquer para comparações futuras.

Devido a sua interdisciplinaridade, é composto por diversas etapas que englobam várias áreas do conhecimento. Nos próximos capítulos, detalharemos cada etapa de um típico sistema desta natureza, da aquisição de sinal até seu reconhecimento de fato, e algumas das técnicas possíveis a serem utilizadas em cada etapa.

4.1 Classificação

Pode-se caracterizar um sistema de reconhecimento de voz de várias maneiras. Abaixo se encontram as principais formas de classificação[21].

4.1.1 Quanto à pronúncia

- Palavras isoladas: sistemas capazes de reconhecer a pronuncia de apenas uma palavra. São os sistemas de reconhecimento mais simples de serem desenvolvidos.

- Palavras conectadas: sistemas capazes de reconhecer uma sentença completa natural e bem pronunciada. São mais complexos que os primeiros.

- Voz contínua: sistemas capazes de lidar com vícios e entonações da linguagem falada. São os sistemas mais difíceis de serem desenvolvidos.

4.1.2 Quanto à dependência de locutor

- Dependente do locutor: sistema de reconhecimento de locutor depende do locutor que utiliza o sistema.

- Independente do locutor: sistemas de reconhecimento de palavras não dependem do locutor que utiliza o sistema, são mais fáceis de serem desenvolvidos que o primeiro.

4.1.3 Quanto ao tamanho do vocabulário

- Vocabulário de tamanho pequeno: possui até cem palavras distintas em sua base de dados.

- Vocabulário de tamanho médio: possui de cem a dez mil palavras distintas em sua base de dados.

- Vocabulário de tamanho grande: mais de dez mil palavras distintas em sua base de dados.

4.2 Abordagens utilizadas

São muitas as abordagens utilizadas para a o desenvolvimento de sistemas de reconhecimento biométrico. No entanto, duas delas se destacam pelo desempenho e, conseqüentemente, pela utilização: uma de reconhecimento por unidade fonética do sinal de fala e outra por reconhecimento do padrão da base do sinal emitido.

Na primeira abordagem, o sinal da fala é considerado um fluxo contínuo de

informação de unidades fonéticas. Portanto, o sinal é fragmentado em unidades fonéticas características do idioma utilizado produzindo uma seqüência característica do sinal. Alguns fatores influenciam diretamente este método de reconhecimento como as pausas feitas durante a fala porque são elas que determinam os fonemas contidos em uma determinada palavra.

Na segunda abordagem, a base do sinal é modelada baseando-se no vocabulário utilizado sendo treinada para reconhecer apenas um número finito de palavras. Esta abordagem é a que consegue uma maior performance no reconhecimento de palavras.

4.3 Dificuldades e restrições

Sistemas dessa natureza apresentam inúmeras dificuldades de implementação. Grande parte dessas dificuldades é oriunda da combinação entre a imensa variação dos sinais vocais de um indivíduo para outro e os diversos fatores que influenciam este processo.

Estes fatores são de naturezas diferentes e influenciam o sinal de voz emitida por um determinado indivíduo castigando a performance de um sistema de identificação por voz, tais como:

- Fatores pessoais: temperamento, pausas durante a fala, estado emocional, envelhecimento, doenças, dificuldades de dicção, etc.

- Fatores culturais: gírias, idioma, acentuação, etc.

- Fatores externos: geralmente ruído ambiente.

Sendo este sistema afetado até por um possível simples resfriado que acometeu seu usuário, o desenvolvimento destes sistemas é bastante complexo até mesmo em suas mais simples formas.

Para as influências causadas por fatores externos, a literatura mostra que a precisão de um sistema de reconhecimento de voz cai de 96% para 73% na medida em que a relação sinal-ruído (SNR) é diminuída em 20dB, e cai para 31% a 10dB.

Por estas razões pode-se afirmar que jamais será desenvolvido um sistema de reconhecimento de voz contínua com vocabulário de tamanho grande e dependente do locutor que tenha uma taxa de acerto igual a um.

5. Digitalização do sinal de voz

A primeira etapa de um sistema de reconhecimento de voz é a etapa da digitalização do sinal de voz ou aquisição de dados. É nesta etapa que se encontra a forma de conversão da voz analógica para a forma digital e onde um processamento básico é efetuado, para, posteriormente, ser possível trabalhar de forma mais complexa sobre estes dados.

O processo de digitalização de um sinal analógico é composto por:

- **Amostragem**: fase responsável por captar amostras do sinal em intervalos de tempos pré-determinados.

- **Quantização**: fase em que valores válidos são atribuídos ao sinal amostrado.

- **Codificação**: fase em que o sinal é transcrito da forma mais adequada para processamento posterior.

- **Filtragem**: fase responsável por separar a região de interesse do sinal.

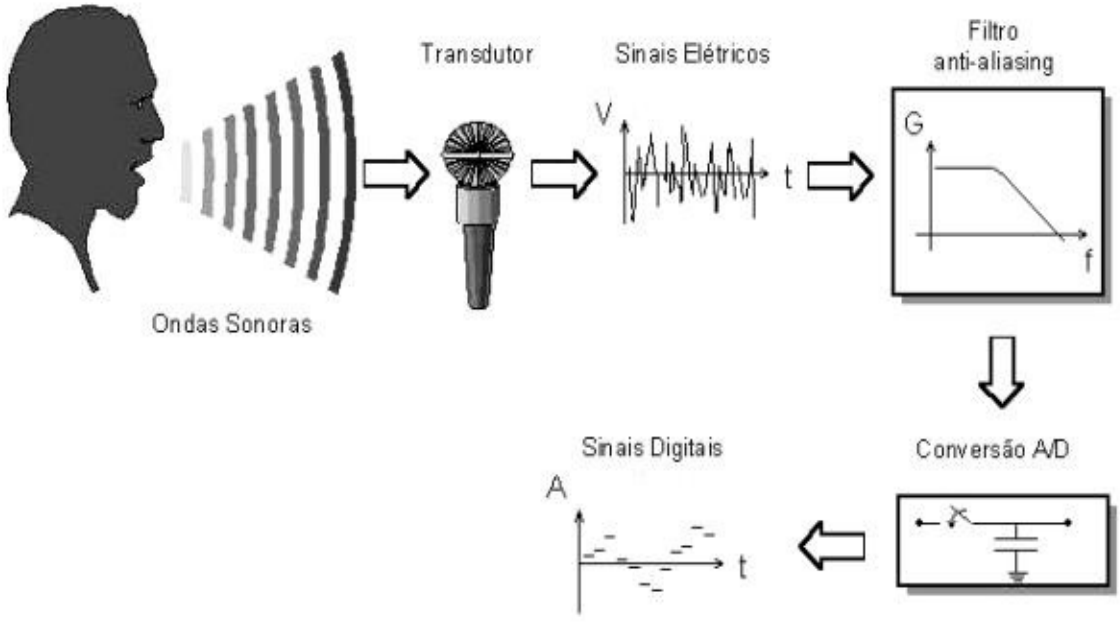

Figura 4 - Digitalização de um sinal de voz

Na **Figura 4** podemos ver a interação entre estas etapas. A seguir iremos detalhar cada uma destas etapas.

5.1 - Amostragem

Sempre que um sinal é amostrado, perdemos parte da informação contida nele. Para minimizar esta perda, é necessário seguir o Teorema de Nyquist que diz:

> *"A freqüência de amostragem deve ser, no mínimo, o dobro da maior freqüência contida no sinal".*

Seguindo este teorema, garantimos que em cada período da onda amostrada terá pelo menos dois pontos coletados pela amostragem garantindo, assim, uma maior fidelidade na recomposição do sinal.

Como foi dito no capitulo anterior, a faixa de freqüência onde se encontra a maior energia armazenada na voz esta entre 100 Hz e 3400 Hz. Logo a freqüência de amostragem deve ser no mínimo 6800 Hz, mesmo sabendo que a faixa audível se estende até 20000 Hz, para que possamos reproduzir o sinal sem o chamado erro de *aliasing*, parâmetro que será explicado a seguir. A metade dessa freqüência de amostragem é chamada de freqüência de *Nyquist* que indica a freqüência máxima do sinal que pode ser reproduzido.

No entanto, como não há como garantir que não existam freqüências no sinal fora dessa faixa, é necessário que ele, o sinal, passe por um processo de filtragem com freqüência de corte, no máximo, igual à freqüência de *Nyquist*, ou seja, um filtro passa-baixa ou filtro *anti-aliasing.*

Aliasing é a superposição de espectros do sinal no domínio da freqüência. Por conta dessa superposição, no momento de ser reproduzido, o sinal ficará

deformado. Podemos ver na **Figura *5*** que ocorreu o efeito de *aliasing*. Isso se deu pelo fato de o Teorema de *Nyquist* não ter sido respeitado, ou seja, não existem, pelo menos, duas amostras do sinal em um mesmo período. Logo, no momento de ser reproduzido, o sinal original sofreu distorções.

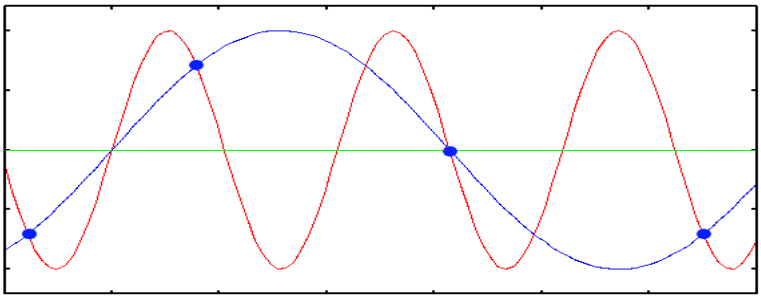

Figura 5 - Efeito aliasing

5.2 Quantização

Enquanto que a amostragem é o processo utilizado para discretizar um sinal, a quantização trata da atribuição de valores definidos às amostras resultantes desse processo. Por ser oriunda de um sinal analógico, a voz, essas amostras podem assumir um número infinito de valores, que evidentemente não serão repassados para a forma digital sem sofrerem aproximações.

A fidelidade dos valores atribuídos às amostras depende da quantidade de bits a serem utilizados no conversor analógico-digital (AD). Quanto maior essa quantidade, maior a resolução do conversor, o que implica em mais níveis de valores possíveis de serem utilizados e, conseqüentemente, em uma maior precisão do valor digital.

No entanto, por maior que seja a resolução de um conversor AD, sempre haverá uma diferença entre o sinal original e o sinal obtido após a quantização. Essa diferença é chamada de erro de quantização. Dependendo do nível do sinal e da

resolução do conversor, esse erro pode ser percebido como um simples ruído branco ou como distorções desagradáveis do sinal de áudio. Na **Figura 6**, fica claro o erro de quantização obtido no processo.

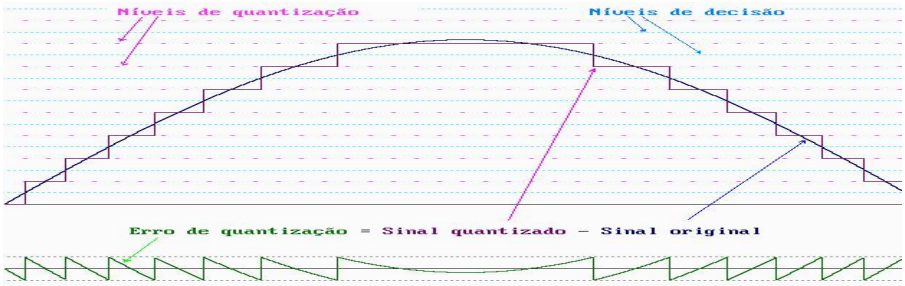

Figura 6 - Quantização de um sinal

Com o sinal devidamente quantizado, ou seja, com cada amostra sendo representada por um número discreto, surge problemas como representar da melhor forma mais os valores recém-amostrados. É nesse ponto que surge a codificação.

5.3 Codificação

A codificação é um processo que visa produzir a representação mais eficiente de uma série de números. Como estamos lidando com um sinal de voz que precisa ser processado, é necessário que os valores das amplitudes das amostras estejam na forma binária, pois é a linguagem compreendida pelo computador. Cada valor produzido pela codificação é chamado de símbolo.

Há diversos tipos de código que podem ser usados, tais como códigos de comprimento fixo, de comprimento variável, de prefixo livre, códigos distintos, etc. Logicamente, há um tipo de codificação mais adequado para cada caso, pois cada tipo de código possui vantagens que favorece o seu uso em determinadas aplicações em detrimentos de outras.

Um dos principais usos da codificação é buscar a menor representação

possível para cada valor de entrada do codificador, de modo que um possível armazenamento dos símbolos gerados ocupe menos espaço na memória e que a transmissão dos mesmos possa ocorrer mais rapidamente. Dessa forma é feita a compactação do sinal. Mas é preciso atentar para o fato de que o gasto computacional para tratar com dados comprimidos é maior.

Para a fase de transmissão do sinal, é importante se preocupar não só com o tamanho dos símbolos, já que muitas vezes é necessário o uso de códigos de tamanho fixo, mas também com a quantidade de energia gasta nesse processo. Tendo em vista essa finalidade, deve-se atentar para que aos símbolos mais freqüentes sejam atribuídos os valores binários de menor energia, o que torna o código mais eficiente.

5.4 Filtragem

Após o sinal estar amostrado é preciso filtrá-lo. Um filtro é um dispositivo que desempenha o papel de permitir ou rejeitar a passagem de uma determinada faixa de freqüência de um sinal.

A resposta em freqüência de um filtro possui uma faixa de passagem, uma faixa de rejeição e uma faixa de transição. Como dito anteriormente, a faixa de freqüência onde se encontra a maior energia armazenada na voz esta entre 100 Hz e 3400 Hz. Logo, um filtro do tipo passa-baixa pode ser utilizado.

Os filtros podem ser:

- analógicos: processam sinais analógicos e são compostos por resistores, capacitores e indutores.

- digitais: processam sinais digitais e são compostos por um processador digital.

São várias as vantagens obtidas através da utilização de filtros digitais. A seguir iremos expor algumas:

- como o filtro digital é programável, estando presente na memória de um processador, a alteração do projeto de um filtro digital, ao contrário do que ocorre com o analógico, não necessita de alterações de hardware.

- a fase de teste dos filtros digitais é mais simples e direta, uma vez que engloba apenas o uso de um processador e uma entrada disponível para o sinal.

- as características de funcionamento dos filtros digitais, ao contrario do que ocorre com os filtros analógicos, não estão sujeitas a alterações devido à variação de temperatura e variações dos valores nominais dos componentes utilizados para construção do filtro tornando-os, assim, mais precisos e estáveis.

Os filtros digitais se dividem em dois tipos:

- resposta ao impulso de duração infinita (IIR);

- resposta ao impulso de duração finita (FIR).

Os filtros digitais de resposta ao impulso de duração infinita ou IIR são filtros que podem ser implementados de forma digital ou analógica e são de natureza recursiva, o que fica claro nas equações lineares de diferenças que as regem que são do tipo:

$$y[n] = \frac{1}{a_0}\left(b_0 x[n] + b_1 x[n-1] + ... + b_i x[n-I] - a_1 y[n-1] - a_2 y[n-2] - ... - a_j y[n-J]\right) \quad [5.1]$$

Onde:

- I é a ordem do filtro direto;

- b_i são os coeficientes do filtro direto;

- J é a ordem do filtro recursivo;

- a_j são os coeficientes do filtro recursivo;

- x[n] é o sinal de entrada;

- y[n] é o sinal de saída do filtro.

Desenvolvendo a equação acima e considerando que a maioria dos filtros IIR possui a_0=1, chegamos a seguinte equação de transferência:

$$H(z) = \frac{\sum_{I=0}^{I} b_i z^{-i}}{1 + \sum_{J=1}^{J} a_j z^{-j}} \qquad [5.2]$$

Como podemos perceber pela equação 5.2, ela é racional e em função de z^{-1}, conseqüentemente, o uso de um filtro IIR resulta em um filtro de menor ordem que o uso de um filtro FIR. No entanto, essa melhoria é obtida em troca de fatores negativos para um filtro como a instabilidade e uma resposta em fase não linear.

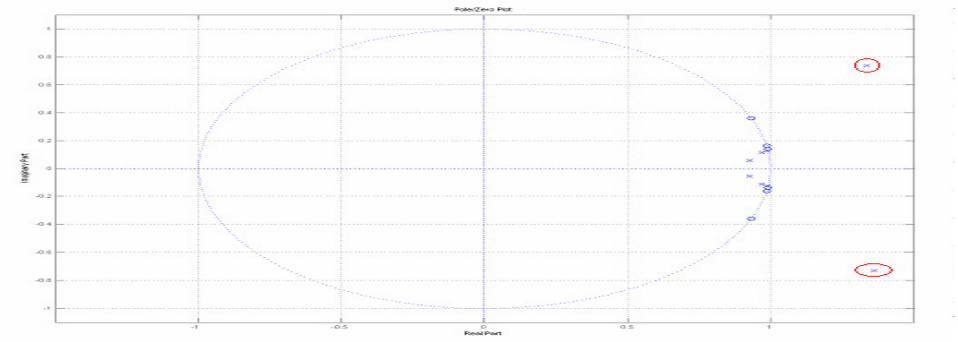

Figura 7 - Diagrama de pólos e zeros de um sistema instável

Para um sistema ser estável, é necessário que todos os pólos, que são os valores que zeram o denominador da equação de transferência, do sistema estejam localizados no círculo de raio unitário no plano-z.

Na **Figura 7** podemos ver o diagrama de pólos e zeros de um sistema instável, pois possui dois pólos fora do círculo de raio unitário.

A **Figura 8** ilustra um típico diagrama de blocos de um filtro IIR. Os blocos Z^{-1} correspondem aos atrasos inseridos no sinal. Também é visível a realimentação feita com estes atrasos o que provoca um acúmulo de erros e, conseqüentemente, sua instabilidade. Mostrando, assim, que não se trata de um sistema BIBO

(*bounded input, bounded output*).

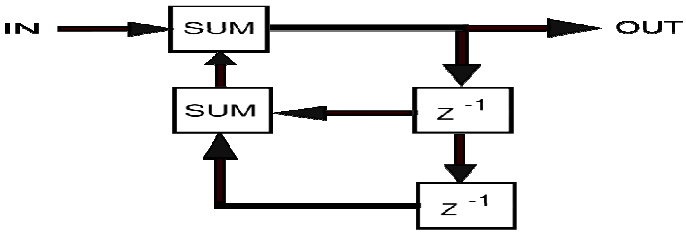

Figura 8 - Diagrama de blocos de um filtro IIR

Uma metodologia muito utilizada nos projetos de filtros IIR é projetar inicialmente um filtro analógico recursivo com suas especificações devido a grande quantidade de métodos que podem ser utilizados para implementação destes. Por isso, normalmente, quando um filtro IIR vai ser implementado, um filtro analógico do tipo Chebyshev, Butterworth ou Elíptico é implementado inicialmente e depois convertido em filtro digital IIR.

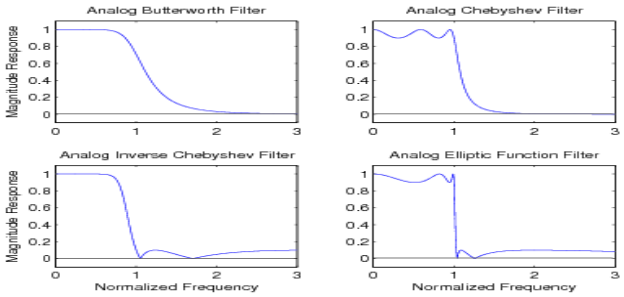

Figura 9 - Tipos de filtros analógicos

Na **Figura 9**, podemos ver o comportamento padrão dos filtros analógicos mencionados. Observe que todos são do tipo passa-baixa.

Já os filtros digitais de resposta ao impulso de duração finita ou FIR contrastam com os filtros IIR porque são caracterizados por sua resposta ao impulso se tornar nula após um determinado período de tempo.

Um filtro FIR é regido pela equação:

$$y[n] = h_0 x[n] + h_1 x[n-1] + ... + h_J x[n-P] \qquad [5.3]$$

Onde:

- P é a ordem do filtro;

- h_j são os coeficientes do filtro;

- x[n] é o sinal de entrada;

- y[n] é o sinal de saída do filtro.

A partir da equação 5.3, chegamos a seguinte equação de transferência:

$$H(z) = \sum_{n=0}^{N} b_n z^{-n} \qquad [5.4]$$

Sua estabilidade fica clara na equação 5.4. Por ser um filtro que não utiliza realimentação em seu modelo, ver **Figura 10**, ele não possui pólos. Logo a transformada inversa não contribuirá com um termo exponencial crescente lateral direito.

Sendo assim, fica claro que os filtros FIR se tratam de um sistema BIBO (*bounded input, bounded output*) porque sua saída é limitada a um múltiplo do maior valor da entrada.

A principal desvantagem dos filtros FIR está na necessidade de um poder computacional maior que os IIR de mesmas especificações fazendo com que sejam necessárias simplificações em seu projeto para que atendam de forma satisfatória aos requisitos a eles empregados.

A **Figura 10** mostra a estrutura básica de um filtro FIR. Podemos perceber que, ao contrario dos filtros IIR, não existe realimentação fazendo com que os erros não se acumulem.

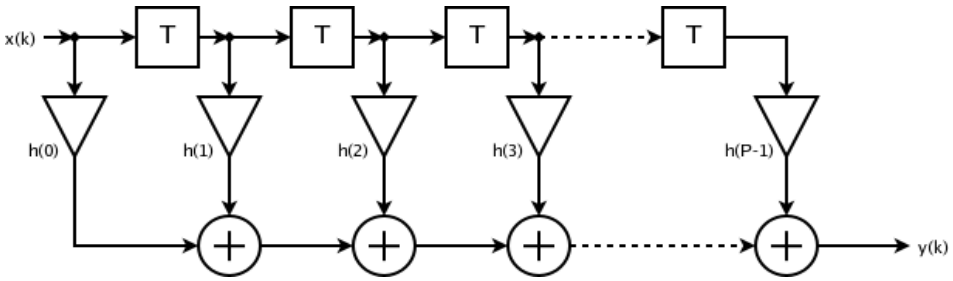

Figura 10 - Diagrama de blocos de um filtro FIR

6. Extração de parâmetros do sinal de voz

Após o processo de digitalização do sinal de voz, é preciso extrair os parâmetros deste sinal porque o processo de reconhecimento é feito em cima de características únicas contidas nele.

Este processo é fundamental para qualquer sistema destinado a reconhecimento de fala, uma vez que é no sinal de voz que estão armazenados todo o dado necessário para o reconhecimento. Considerando que parte do sinal contém informação redundante e insignificante para a operação de reconhecimento, é preciso fazer um rigoroso tratamento para que apenas dados significativos sejam tratados. Caso dados insignificantes sejam extraídos do sinal e considerado no momento do reconhecimento, o classificador dificilmente conseguirá classificar corretamente.

O princípio básico de qualquer extrator de parâmetros de um sinal de voz consiste em dividir o sinal em diversos grupos de *frames*, fonemas, segmentos ou qualquer unidade suficiente para que os dados extraídos representem os sinal a ser reconhecido da melhor forma possível. Mesmo que o classificador possua um alto percentual de acerto, ele não conseguirá bons resultados caso os parâmetros extraídos do sinal sejam incoerentes.

Existem várias métodos de análise espectral utilizados para a extração de parâmetros dos sinais de voz, entre eles estão: a transformada rápida de Fourier (*Fast Fourier Transform* ou FFT) e o de codificação preditiva linear (*Linear Predictive Coding* ou LPC). A seguir, mostraremos as técnicas utilizadas por cada um destes métodos e suas dificuldades de desenvolvimento.

6.1 Codificação Preditiva Linear (LPC)

A codificação preditiva linear, ou simplesmente LPC, é umas das técnicas com maior relação entre custo e benefício utilizados. Isso se dá pelo fato de que esta técnica consegue analisar e codificar um determinado sinal de voz com uma boa qualidade a uma pequena taxa de bits fornecendo parâmetros precisos do sinal analisado com um baixo esforço computacional[22].

O algoritmo LPC utiliza-se do funcionamento do sistema vocal para embasar sua funcionalidade. Ele encara a voz como um conjunto de um som caracterizado pela intensidade e freqüência (som produzido pelas cordas vocais) e suas ressonâncias (causadas pela garganta e pela boca). Essas ressonâncias são chamadas de formantes.

Estes formantes são retirados do sinal através de um processo chamado filtragem inversa, mostrada na equação abaixo. Após este processo, o som produzido pelas cordas vocais é retirado do sinal e são chamados de resíduos. Desta forma os coeficientes que descrevem os resíduos e os formantes podem ser armazenados.

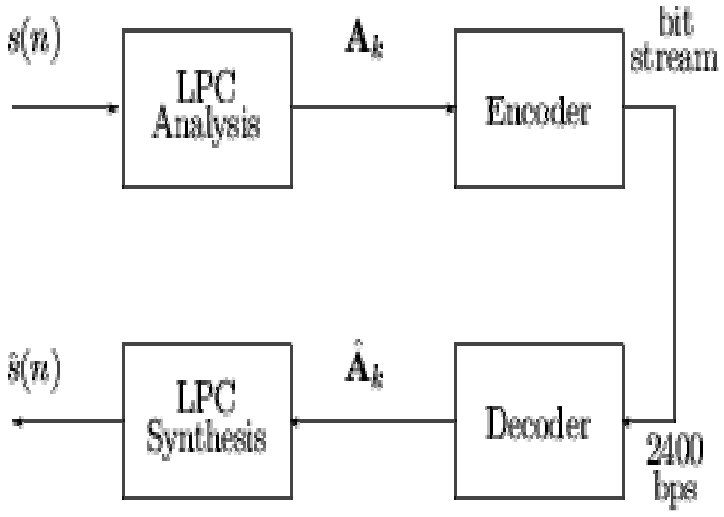

Figura 11 - Análise e síntese de sinal por LPC

Como mostrado na **Figura 11**, para recuperar o sinal de voz, o LPC sintetiza o sinal para inverter o processo utilizando os resíduos para criar um sinal característico e os formantes para criar um filtro. O resultado disso é a reprodução da voz gravada anteriormente. Para executar todo este processo, o LPC fragmenta o sinal em inúmeras partes chamadas frames. Utilizando entre 30 e 60 *frames* por segundo é possível reconstruir a voz de maneira inteligível.

O problema básico a ser resolvido pelo algoritmo LPC é estimar corretamente estes formantes. A equação 6.1 é chamada de preditor linear, razão pelo qual o algoritmo possui esse nome, é a solução padrão para a equação diferencial que expressa cada janela do sinal como uma combinação linear das demais janelas analisadas para reconstruir um valor de amostra do sinal digital.

$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + e[n] \qquad [6.1]$$

Onde:

- a_k são os coeficientes de predição;

- s[n-k] são as amostras passadas;

- p é a ordem da predição;

- e[n] é o erro de predição ou o resíduo;

- s[n] é o valor de amostra a ser reconstruído.

Os coeficientes ou pesos a_k caracterizam os formantes do sinal. Para estimá-los com maior precisão é preciso levar e[n] ao menor valor possível. Isto é feito minimizando o erro quadrático médio entre o sinal real e o sinal previsto dado por:

$$E_m = \sum_n e^2[n] = \sum_n \left(s[n] - \sum_{k=1}^{p} a_k s[n-k] \right)^2 \qquad [6.2]$$

Vários métodos, como a auto-correlação e a covariância, podem definir o valor do somatório, levando-o para o menor valor possível e, assim, elevar as

probabilidades de acerto na reconstrução e reconhecimento do sinal.

O reconhecimento de sons nasais, apenas representados por consoantes, requer um algoritmo de reconhecimento de maior complexidade porque, para um som nasal, uma cavidade nasal é aberta como um tubo lateral introduzindo sons e, conseqüentemente, ruídos, tornando o sinal aperiódico. As vogais, que são compostas por combinações de sons periódicos e não-nasais, possuem um melhor desempenho no processo de reconhecimento de fala. Na prática, utiliza-se o mesmo algoritmo para reconhecer ambos os tipos de sons considerando uma pequena margem de erro.

Caso os coeficientes sejam calculados corretamente, o som original pode ser reconstruído através de uma filtragem inversa e, sendo assim, o esforço para extrair e codificar todas as informações (freqüência e amplitude) deste sinal se torna bastante pequeno. Na **Figura 12** podemos ver um sinal reconstruído utilizando LPC.

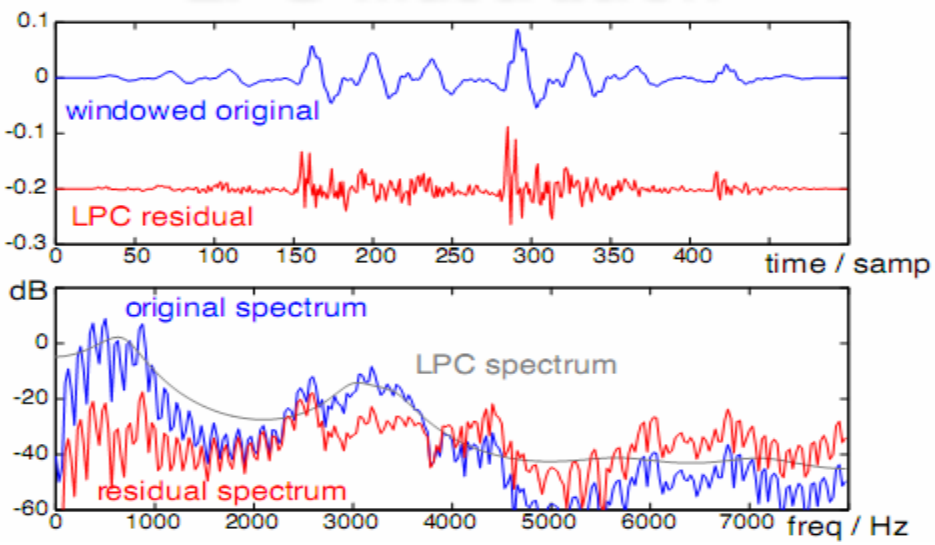

Figura 12 - Reconstrução de sinal por LPC

Assim, o algoritmo LPC deve decidir se cada frame analisado é composto por sons periódicos ou aperiódicos e, com isso, estimar a freqüência e a intensidade para reconstruir o sinal através de um processo inverso.

Um dos fatores que mais prejudicam o desempenho do LPC na reconstrução

de um determinado sinal é o fato de que alguns sons da fala são compostos pela combinação de sons periódicos e aperiódicos. Esta categoria de som não pode ser reconstruída por um algoritmo LPC dos mais simples.

Aliado a isso, está o fato de que qualquer imprecisão na estimativa dos coeficientes dos formantes significa que a informação foi agregada aos resíduos desequilibrando a equação e, conseqüentemente, denegrindo a probabilidade de reconstrução do sinal. Portanto, o resíduo contém importantes informações do sinal e, caso estas informações não sejam consideradas em todo o processo, o resultado será um sinal de má qualidade. Mas, como uma das conseqüências do LPC é a compressão do sinal, é importante avaliar até que ponto as informações armazenadas nos resíduos devem ser consideradas porque para se chegar ao nível máximo de qualidade na reconstrução do sinal não existirá compressão.

Várias são as formas para se codificar os resíduos de forma eficiente para aumentar a qualidade do sinal reconstruído sem aumentar a quantidade de bits utilizados para descrevê-lo. Um dos métodos mais bem sucedidos é o que utiliza um *codebook*, termo que será explicado no próximo capítulo[22].

Desta forma podemos concluir que o LPC é uma poderosa técnica utilizada para analisar e representar sinais provenientes do sistema vocal humano a uma pequena taxa de bits provocando uma baixa latência em transmissões. Sendo assim, bastante utilizado em sistemas destinados a reconhecer usuários através do sinal e voz.

6.2 Análise utilizando técnicas de *Fourier*

O estudo de sinais que utiliza representações senoidais para descrever o

comportamento de qualquer sinal é chamado de análise de *Fourier* em homenagem a Joseph Fourier (1768 – 1830) por sua contribuição à teoria publicada em 1822 que garante a representação de funções como uma superposição ponderada de senóides.

Se um sinal, considerando a teoria de *Fourier*, for aplicado a um sistema de comportamento linear terá como resultado uma superposição ponderada das respostas do sistema a cada senóide complexa.

Esta teoria tem aplicação difundida além das fronteiras dos estudos e analise de sinais e comportamentos de sistemas, sendo muito utilizado em todos os ramos da ciência e da engenharia.

Existem as seguintes representações de *Fourier* para os sinais: contínuos (definidos em todos os pontos do intervalo) e discretos (definidos apenas em pontos específicos).

Tempo	**Periódica**	**Não Periódica**
Contínuo	Série de *Fourier* (FS)	Transformada de *Fourier* (FT)
Discreto	Série de *Fourier* de tempo Discreto (DTFS)	Transformada de *Fourier* de tempo Discreto (DTFT)

Como os sistemas computacionais não trabalham com sinais contínuos, a FT é uma aproximação da FS e a DTFT é uma aproximação da DTFS.

As aplicações mais comuns que utilizam estas representações são as de análise da interação entre sinais e sistemas e avaliação matemática das propriedades do sinal ou do comportamento do sistema. A FT e a DTFT são mais utilizadas no primeiro caso enquanto a DTFS e a FS são mais utilizadas no segundo caso.

6.2.1 A série de *Fourier*

A série de Fourier nos leva as seguintes conclusões para uma função periódica **x(t)** com período T_0:

- a freqüência fundamental da função é:

$$f_0 = 1/T_0 \qquad [6.3]$$

- x(t) pode ser representado por uma superposição ponderada de funções senoidais do tipo:

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos(2\pi f_0 nt) + b_n sen(2\pi f_0 nt)) \qquad [6.4]$$

- $2\pi f_0$ é considerado como a primeira harmônica ou freqüência angular fundamental do sinal sendo chamado de ω_0:

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos(\omega_0 nt) + b_n sen(\omega_0 nt)) \qquad [6.5]$$

- o somatório é chamado de série de *Fourier* com a_n e b_n sendo os coeficientes da série sendo definidos como:

$$a_n = \frac{2}{T} \int_{\frac{-T}{2}}^{\frac{T}{2}} x(t) \cos(\omega nt) dt \quad \text{e} \quad b_n = \frac{2}{T} \int_{\frac{-T}{2}}^{\frac{T}{2}} x(t) sen(\omega nt) dt \qquad [6.6]$$

Estas considerações são válidas para funções tanto discretas como contínuas. Se tivermos uma função com taxa de amostragem $\mathbf{F_s}$ e uma série de **N** amostras no período, teremos:

- a menor freqüência será:

$$f_0 = \frac{F_s}{N} \qquad [6.7]$$

- o somatório de funções senoidais será:

$$x(nT_s) = a_0 + \sum_{k=1}^{\infty}\left(a_k \cos\left(2k\pi\frac{n}{N}\right) + b_k sen\left(2k\pi\frac{n}{N}\right)\right) \quad [6.8]$$

- a_0 corresponde a componente contínua na série de *Fourier*:

$$x(t) = a_0 + \sum_{k=1}^{\infty}\left(a_k \cos(k\omega_0 t + \theta_k)\right) \quad [6.9]$$

6.2.2 A transformada de *Fourier*

A transformada de *Fourier* é uma generalização da série de *Fourier* complexa capaz de representar sinais do domínio do tempo no domínio da freqüência e, sua transformada inversa, representam sinais do domínio da freqüência no domínio do tempo tornando possível uma melhor análise das características inerentes a cada sinal estudado.

Figura 13 - Decomposição de onda pela FT

Se a função existir em toda a sua extensão, de $-\infty$ a $+\infty$, a integral a seguir é definida como transformada de *Fourier* para tempo contínuo:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt \quad [6.10]$$

Quando tratamos em termos de tempo discreto, com o mesmo requisito de existência em toda a extensão, a integral a seguir é definida como transformada de *Fourier* para tempo discreto:

$$X(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega n} \qquad [6.11]$$

Cada transformada possui o seu par inverso. Sendo definidas como:

- transformada inversa no tempo contínuo:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{j\omega t}d\omega \qquad [6.12]$$

- transformada inversa no tempo discreto:

$$x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\Omega})e^{j\Omega n}d\Omega \qquad [6.13]$$

6.2.3 Transformada Rápida de *Fourier*

Outro método bastante utilizado para extração de características de um sinal, principalmente para o reconhecimento de palavras isoladas, é a transformada rápida de *Fourier* ou FFT (*Fast Fourier Transform*).

Este método transfere a abordagem do estudo do sinal em função do tempo para o estudo o sinal em função das freqüências presentes no nele, tornando-se eficaz no reconhecimento de sinais independentes de locutor.

Este algoritmo se baseia na divisão da DTFS em várias séries de DTFS de menor ordem e utiliza-se das propriedades de periodicidade e simetria da senóide complexa $e^{jk2\pi n}$. Menos computação é necessária para efetuar a análise e a combinação de fatores da DTFS de pequenas ordens.

Pela transformada discreta de *Fourier*, são necessárias $\mathbf{N^2}$ operações de multiplicação para a obtenção do espectro do sinal. No entanto, com o uso da FFT, este cálculo cai para $\mathbf{N.log_2N}$ o que significa que o número de multiplicações realizadas da forma direta cresce exponencialmente em relação a ao número de

pontos de uma amostra. Enquanto que com a FFT, o número de multiplicações é bem menor como pode ser visto na **Figura 14**.

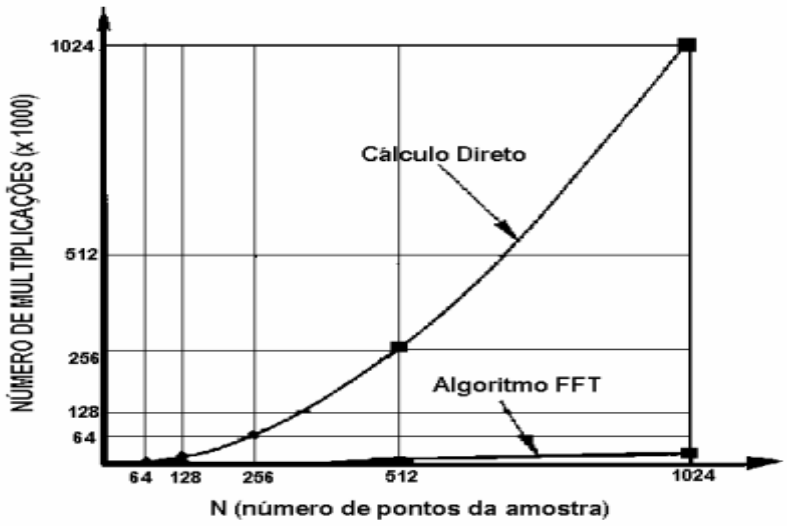

Figura 14 - Diferença entre operações com cálculo direto e com FFT

6.2.3.1 A operação *Butterfly*

A operação *butterfly* é um algoritmo de decomposição da DTFT para que o número de operações necessárias para se computar o resultado final seja diminuído em até a metade. Para demonstrar como isto é possível vamos tomar a DTFS supondo um *N* par:

$$x[n] = \sum_{k=0}^{N-1} X[k] e^{jkn\Omega_0} \qquad [6.14]$$

Dividimos X[k], 0 ≤ k ≤ *N*-1 em sinais de índices par (subscrito p) e ímpar (subscrito i) obtendo:

$$X_p = X[2k] \text{ e } X_i[k] = X[2k+1] \text{ com } 0 \le k \le N'-1 \qquad [6.15]$$

Onde N' = N/2, sendo $x_p[n] \xleftarrow{DTFS;\Omega_0'} X_p[k]$ e $x_i[n] \xleftarrow{DTFS;\Omega_0'} X_i[n]$, com $\Omega_0' = 2\pi/N'$. Agora se expressa a equação 6.14 como uma combinação dos coeficientes N' de $X_p[k]$ e $X_i[k]$ da DTFS:

$$x[n] = \sum_{kpar} X[k] e^{jkn\Omega_0} + \sum_{kimpar} X[k] e^{jkn\Omega_o} \qquad [6.16]$$

Substituindo dos índices par (p) e ímpar (i) por 2m e 2m+1, respectivamente, e aplicando as definições de $X_p[k]$, $X_i[k]$ e $\Omega_0^{'} = 2\Omega_0$ nos resultados obtidos pelas substituições anteriores, obtemos:

$$x[n] = x_p[n] + e^{jn\Omega_0} x_i[n] \text{ , } 0 \leq n \leq N\text{-}1 \qquad [6.17]$$

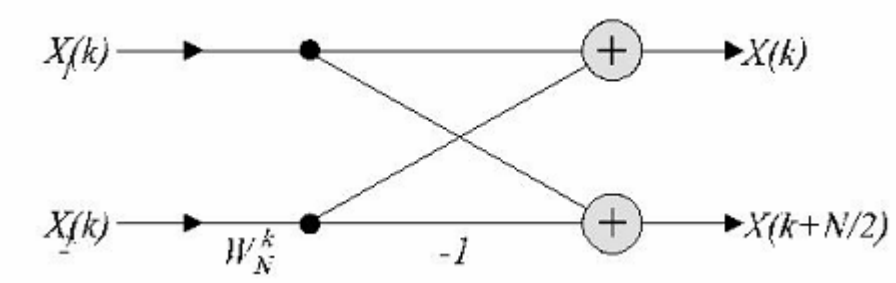

Figura 15 - Operação *Butterfly* (DFTS de 2 pontos)

Isto mostra que x[n] é uma combinação ponderada de suas componentes pares e ímpares. Simplificando ainda mais as equações utilizando a propriedade de periodicidade obtemos:

- Para os primeiros valores de N' da DTFS:

$$x[n] = x_p[n] + e^{jn\Omega_0} x_i[n] \text{ , } 0 \leq n \leq N'\text{-}1 \qquad [6.18]$$

- Para os segundos valores de N' da DTFS:

$$x[n+N'] = x_p[n] - e^{jn\Omega_0} x_i[n] \text{ , } 0 \leq n \leq N'\text{-}1 \qquad [6.19]$$

O esforço computacional para as equações 6.18 e 6.19 são $N^2/2 + N/2$ multiplicações complexas. Se levarmos o N para um valor muito grande, quantidade de multiplicações complexas se aproximará proporcionalmente ao fator $N^2/2$ que é exatamente metade da quantidade de multiplicações requeridas para calcular x[n] de forma direta.

É possível reduzir ainda mais a quantidade de operações computacionais necessárias de dividirmos um maior número de vezes os valores $X_p[k]$ e $X_i[k]$ em

seqüências de índices par e ímpar. Uma maior otimização pode ser encontrada quando N é uma potência de 2 porque, neste caso, podemos subdividir a DTFS até que o tamanho de sua inversa seja 2 de acordo com a **Figura 15**.

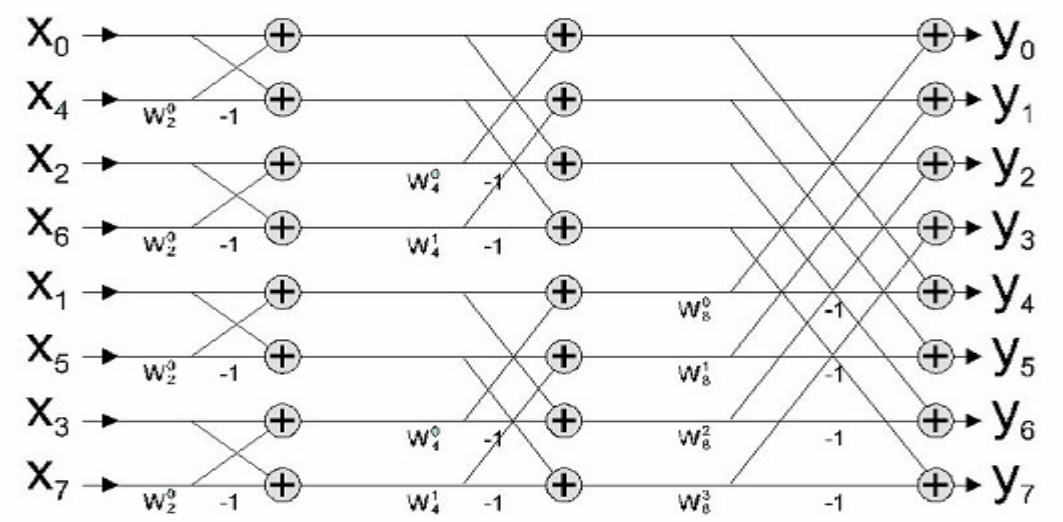

Figura 16 - Operação *Butterfly* para FFT de 8 pontos

A **Figura 16** ilustra a operação *butterfly* da FFT para um N=8. A partição das seqüências de índices da entrada par e ímpar acaba por permutar a ordem dos coeficientes da DTFS e isso é chamado de inversão de bits, uma vez que X[k] pode ser alterado de local invertendo-se os bits de uma representação binária para o índice k, como mostrado na tabela a seguir:

Seqüência da operação butterfly	*Endereço binário*	*Bit Reverso*	*Elementos correspondentes a seqüência original*
X_0	*000*	*000*	X_0
X_4	*100*	*001*	X_1
X_2	*010*	*010*	X_2
X_6	*110*	*011*	X_3
X_1	*001*	*100*	X_4
X_5	*101*	*101*	X_5
X_3	*011*	*110*	X_6
X_7	*111*	*111*	X_7

Um exemplo em que a economia computacional é extremamente grande pode ser visto caso o valor de *N* seja muito grande, por exemplo:

- Se $N = 8192$ ou 2^{13}, a quantidade de operações aritméticas requeridas pelo método direto seria, aproximadamente, 630 vezes maior.

6.3 Transformada *Wavelet*

A FFT é uma eficiente técnica utilizada na análise de sinais, no entanto, uma de suas principais desvantagens está no fato de não haver um compromisso entre a resolução de tempo e a resolução de freqüência, ou seja, a FFT não trabalha de forma muito eficiente para sinais não-estacionários como o sinal de voz porque ela identifica exatamente quais são as freqüências presentes num determinado sinal, mas não informa nada a respeito das localizações destas freqüências identificadas no sinal. Para que isto seja feito, é necessária a utilização de técnicas complementares para análise de sinais. Um desses métodos é conhecido como Transformada *Wavelet* (WT).

As principais aplicações práticas para uso da WT são:

- Detectar características em sinais;
- Identificar características em sinais;
- Filtrar ruídos em sons e imagens;
- Estudos em tremores sísmicos;
- Estudos em imagens médicas;
- Compressão de imagens e sons.

A *Wavelet*, traduzida como ondaleta ou pequena onda, é uma onda de duração limitada com valor médio igual a zero. A **Figura 17** mostra a diferença entre

uma onda senoidal e uma forma de onda *wavelet.*

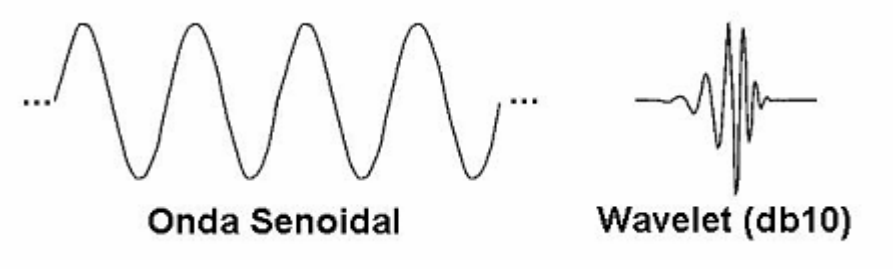

Figura 17 - Onda senoidal e Wavelet

De forma semelhante à FT, a WT é definida como a soma de todo o sinal no domínio do tempo multiplicado pela função wavelet.

Dado um sinal não-estacionário x(t), a WT é o produto interno de x(t) com a família de funções básicas de dois parâmetros denotadas por:

$$\psi_{\tau,a}(t) = |a|^{-1/2} \psi\left(\frac{t-\tau}{a}\right) \qquad [6.20]$$

em que *a* é um fator de escala (conhecido como parâmetro de dilatação) e τ é um fator de posição ou retardo no tempo. Assim, matematicamente, a WT de x(t) é:

$$W_x(\tau, a) = |a|^{-1/2} \int_{-\infty}^{+\infty} x(t) \psi^* \left(\frac{t-\tau}{a}\right) dt \qquad [6.21]$$

A **Figura 18** mostra um sinal decomposto pela WT.

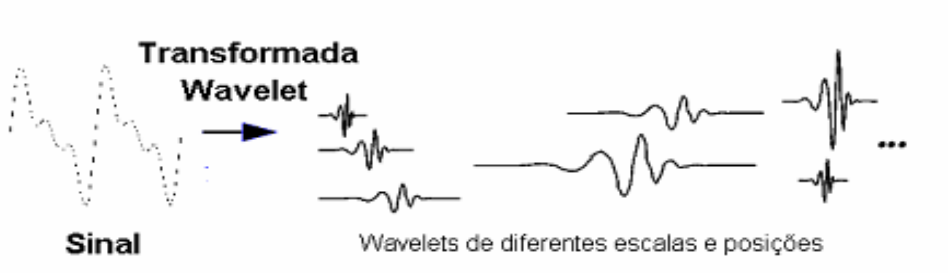

Figura 18 - Decomposição de onda pela WT

A função básica $\psi_{\tau,a}(t)$, como vimos, é chamada de wavelet. Ela constitui os blocos de construção da análise wavelet como podemos ver pela equação 6.20 onde vemos que ela é uma versão, multiplicada por fatores de escala e deslocada

no tempo, de um protótipo $\psi(t)$, chamado *wavelet básica* ou *wavelet mãe*. O parâmetro τ dá a posição enquanto o parâmetro *a* é responsável por controlar o conteúdo de freqüência da wavelet.

Para um valor de *a*<<1, a wavelet $\psi_{\tau,a}(t)$ é uma versão altamente concentrada e contraída da wavelet básica $\psi(t)$, com conteúdo de freqüência concentrado em sua maioria na faixa de alta freqüência. Por outro lado, para um valor de *a*>>1, teremos uma wavelet $\psi_{\tau,a}(t)$ expandida em relação a wavelet básica $\psi(t)$, com conteúdo de freqüência concentrado nas freqüências mais baixas.

A **Figura 19** exibe duas formas de wavelets: a forma de *Haar* concentrada nas altas freqüências e a forma de *Daubechies* concentrada nas baixas freqüências. Ambas possuem um suporte finito, no entanto a forma de *Daubechies* é contínua e tem melhor resolução de freqüência.

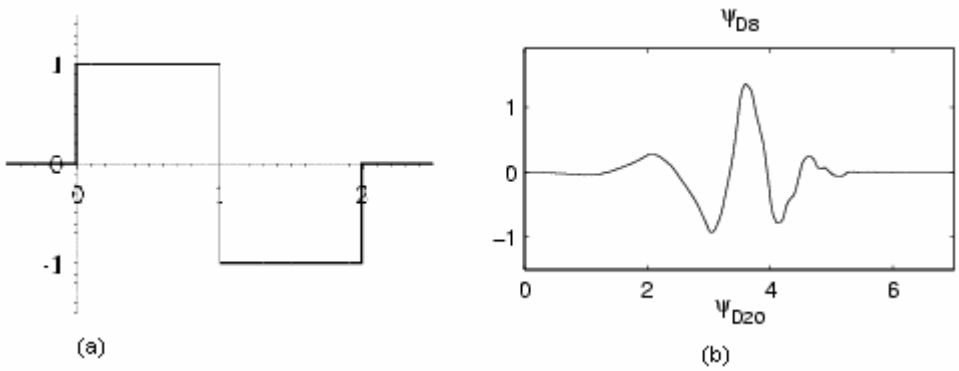

Figura 19 - (a) Wavelet Haar. (b) Wavelet Daubechies

7. Algoritmos de reconhecimento de voz

Como vimos no capítulo anterior, características do sinal são extraídas para que o reconhecimento seja feito em uma etapa posterior. Neste capítulo vamos tratar exatamente desta etapa do processo de reconhecimento de voz: a classificação dos dados.

O processo de classificação das características extraídas do sinal é, necessariamente, dependente do processo de extração dessas porque se esta etapa for executada utilizando-se de dados incoerentes, imprecisos ou insuficientes, a qualidade do método utilizado para reconhecer e associar o sinal ao seu correspondente de forma correta será completamente comprometida, de tal forma que todo o sistema será classificado como ineficiente.

Vários são os algoritmos citados pela literatura para efetuar este tipo de reconhecimento. Alguns recorrem a meios mais tradicionais como a correlação entre sinais. Outros utilizam meios desenvolvidos e aprimorados mais recentemente como redes neurais com várias camadas.

Para modelos de reconhecimento de fala dependente de texto os métodos mais usados são os modelos estocásticos *Dinamic Time Warping* (DTW) e *Hidden Markov Models* (HMMs). Para modelos de sistemas independentes de texto, os métodos mais usados são métodos baseados em quantização vetorial (VQ) e métodos baseados em *Gaussian mixture models* (GMMs). Algumas técnicas fazem uso de *codebooks* para alinhar sinais e compará-los.

Não é objetivo deste trabalho detalhar cada algoritmo utilizado. Nós vamos realizar uma avaliação de alguns métodos e diferenciá-los pelas vantagens e desvantagens de cada um, assim como a complexidade da implementação e o

desempenho computacional de cada um aqui descrito.

7.1 *Codebooks*

Codebook, como o próprio nome sugere, é um conjunto de possíveis códigos com codificações aperfeiçoadas a serem utilizados que são armazenados em uma lista para uma comparação no momento do reconhecimento. Esta lista de códigos é preenchida no momento da criação do sistema ou é preenchida sob demanda com o sistema já em uso.

No processo, o sinal compara o resíduo a cada código armazenado na lista. Após a identificação, o sistema apenas envia o código cadastrado nesta lista ao invés de codificar o resíduo e, assim, aumentar sua taxa de perdas ou desempenho por conta de uma eventual grande quantidade de bits necessários. O sistema recebe este código e, a partir dele, reconstrói o resíduo correspondente para estimular o filtro formado pelos formantes.

Para que esta forma de trabalho alcance seu principal objetivo, diminuir a quantidade de bits utilizados para codificar o resíduo, a lista de códigos deve ser grande o bastante para incluir todos os tipos possíveis de resíduos. No entanto, se esta lista for muito grande, o tempo gasto para buscar o código a ser utilizado também será.

Uma solução utilizada para este problema é a utilização de dois *codebooks*, um preenchido no momento da criação do sistema e o outro sendo preenchido de acordo com a necessidade de reconhecimento.

Para que esta solução seja suficiente, a primeira lista citada deve ser bastante otimizada, contendo apenas os códigos suficientes para representar um

período do resíduo do sinal. A segunda lista é preenchida com o decorrer do uso do sistema podendo até possuir um tamanho limite, desde de que um algoritmo possa rejeitar as amostras mais redundantes da lista, para que o sistema possa alcançar um bom desempenho.

7.2 *Dynamic Time Warping* (DTW)

O algoritmo DTW possui este nome por ser baseado em programação dinâmica, comparando padrões de sinais, não obrigatoriamente do mesmo tamanho, baseada em suas formas, que podem variar com o tempo como podemos ver na **Figura 20**.

Este algoritmo alinha quadros de sinais de fala pronunciados em diferentes velocidades e encontra a distância (similaridade) entre dois sinais analisados sendo interpretado como: quanto menor a distância entre os sinais, maior a semelhança entre eles. Como conseqüência, fica claro que o DTW faz uso de um *codebook* para alinhar um sinal previamente gravado a um sinal a ser reconhecido.

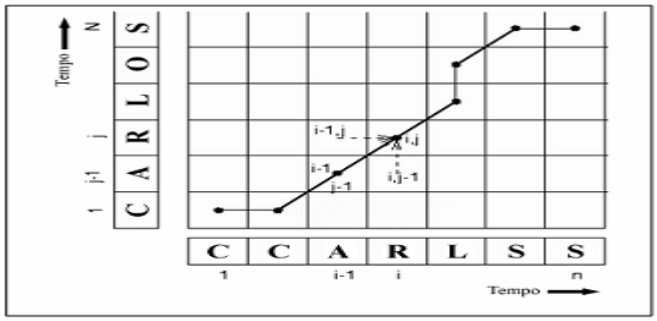

Figura 20 - Funcionamento do DTW clássico

O objetivo do método DTW é aperfeiçoar a função de alinhamento o que corresponde a minimizar a distância entre os parâmetros extraídos dos sinais. Para tanto, as funções que regem o comportamento dos parâmetros devem ser corretamente determinadas, o que pode ser conseguido através da utilização da

programação dinâmica aliada a um processo de decisão.

A principal desvantagem do método DTW é seu caráter determinístico implicando em um padrão de referência correspondente a cada palavra pronunciada, o que conduz a necessidade de se ter vários padrões de uma mesma palavra, ou seja, este algoritmo necessita de um *codebook* de tamanho considerável para que o grau de reconhecimento seja alto na intenção de cobrir todas as variações possíveis na sua elocução, por um ou mais locutores.

7.3 Hidden Markov Models – HMMs

Os processos de Markov têm aplicações em diversas áreas e tem como uma das principais características ser um processo sem memória, ou seja, todos os acontecimentos do passado estão completamente resumidos no estado atual. A teoria básica deste modelo foi publicada nos anos 60 sendo utilizada pela primeira vez para processamento de um sinal de voz nos anos 70. Mas apenas recentemente este modelo foi aperfeiçoado para ser utilizado em sistemas de reconhecimento de fala.

Um processo ou cadeia de Markov é um conjunto finito de estados ligados entre si por transições ligadas a processos estocásticos formando uma máquina de estados. Se somente as informações de saída do modelo forem visíveis a um observador externo os estados são denominados ocultos originando o nome do modelo *Hidden Markov Models*.

Diversos fenômenos são modeláveis por máquinas de estados finitas. Se a natureza do problema possui como característica processos estocásticos, então pode ser modelada utilizando as HMMs. A **Figura 21** mostra uma cadeia de Markov

com 3 símbolos.

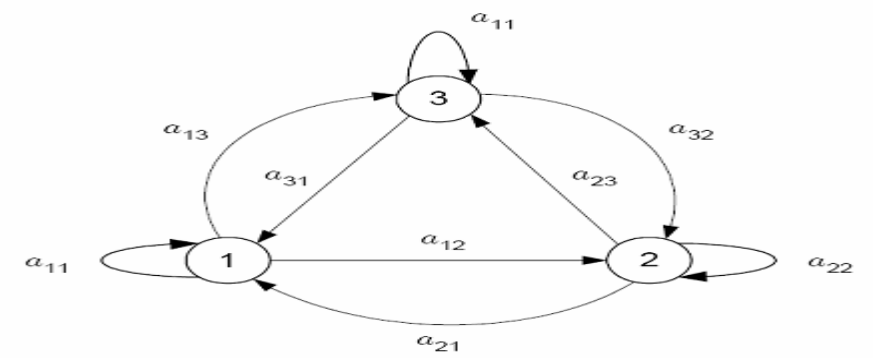

Figura 21 - Cadeia de Markov com 3 símbolos

Um HMM tem como características:

- Os estados individuais são rotulados como $S=\{S_1, S_2, ..., S_N\}$, sendo N o número de estados do modelo e o estado t definido como q_t.

- M sendo o número de símbolos de observação distintos por estado correspondendo à saída do sistema modelado sendo denotados por $V=\{v_1,v_2,..., v_M\}$.

- A distribuição de probabilidade de transição do estado $A=\{a_{ij}\}$, onde:

$$a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i),\ 1 \leq i, j \leq N \qquad [7.1]$$

Para o caso onde qualquer estado pode transitar para qualquer outro estado em apenas um passo, tem-se $a_{ij} > 0$ para todo i e j.

- A distribuição de observações no estado j, $B=\{b_j(k)\}$ onde:

$$b_j(k) = P(O_t = v_k \mid q_t = S_j),\ 1 \leq j, k \leq N \qquad [7.2]$$

- A distribuição no estado inicial $\pi=\{ \pi_i\}$, onde:

$$\pi_i = P(q_1 = S_j),\ 1 \leq i \leq N \qquad [7.3]$$

Dessa forma podemos perceber que para definir completamente um processo HMM precisa-se definir os parâmetros M, N, uma série de observações e três conjuntos de medidas de probabilidade A, B e π. Sendo assim podemos escrever um processo HMM como:

$$\lambda = (A, B, \pi) \qquad [7.4]$$

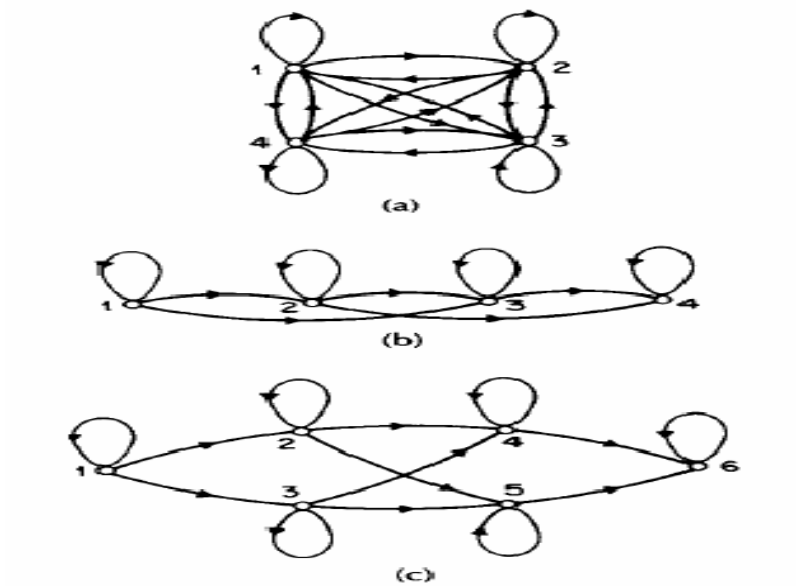

Figura 22 - Topologias de HMM. a) modelo ergótico. b) modelo esquerda-direita. c) modelo esquerda-direita paralelo

Os processos de HMM possuem as seguintes topologias possíveis:

- O modelo ergótico completamente conectado onde é possível transitar livremente de qualquer estado para qualquer estado, **Figura 22a**;

- O modelo esquerda-direita, onde não é possível transitar para um número de estado apenas evolui ou permanece o mesmo com o decorrer do tempo, ou seja, nunca será possível retornar para um estado já assumido em um tempo passado, conforme **Figura 22b e 22c**.

Os algoritmos mais comumente utilizados nas etapas de determinação da seqüência de estados são: Viterbi utilizado para decodificação, *Backward* e *Forward*, utilizados para a avaliação e Baum-Welch utilizado para a fase de treinamento. Estes algoritmos são definidos a seguir.

7.3.1 Viterbi

O algoritmo de Viterbi tem como objetivo encontrar a seqüência de estados

que tem maior probabilidade de ter gerado uma seqüência de estados observados. Neste caso, se assemelha com o DTW porque ambos procuram a menor distância que possa ter gerado a saída. Assim é definida a maior probabilidade ao longo de um caminho no instante *t*, que considera todas as observações até este instante:

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P(q_1 q_2 \ldots q_t = i, O_1 O_2 \ldots O_T \mid \lambda) \qquad [7.5]$$

Para observar corretamente o caminho de transições de estados é necessário maximizar a expressão 7.5

7.3.2 Forward-Backward

Dado um conjunto de observações {O}, estes algoritmos têm como objetivo verificar qual a probabilidade deste conjunto ter sido gerado pelo modelo de probabilidades da equação 7.4 onde:

$$A = a_{ij} = P(s_{t+1} = j \mid s_t = i); B = b_j(O_t) = P(O_t \mid s_t = j); \pi = \pi_i = P(s_1 = i) \qquad [7.6]$$

A aplicação destes algoritmos resulta em:

- *Forward*:

$$P(O \mid \lambda) = \sum_{ieS_p} \alpha_T(i) \text{ com } \alpha_t(i) = P(O_1, O_2, \ldots, O_t, s_t = i \mid \lambda) \qquad [7.7]$$

- *Backward*:

$$P(O \mid \lambda) = \sum_{ieS_p} \pi_i b_i O_1 \beta_1(i) \text{ com } \beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots, O_T, s_t = i \mid \lambda) \qquad [7.8]$$

7.3.3 Baum-Welch

Com uma seqüência finita de observações utilizadas para o treinamento, não existe maneira de maximizar a probabilidade da seqüência de observações

resolvendo de forma analítica o conjunto de parâmetros para um dado modelo. No entanto, é possível escolher um λ = (A, B, π) que maximize localmente a expressão P(O| λ) utilizando procedimentos iterativos como o algoritmo *Baum-Welch* (também conhecido como EM (*expectation-maximization*)). A seguir apresentamos este algoritmo em termos das variáveis α_t e β_t dos algoritmos *forward* e *backward*, respectivamente.

Para uma seqüência de observações O = {O_1, O_2, ..., O_T}, a re-estimação da probabilidade de transição do estado *i* para o estado *j* da matriz de transição é:

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(j)} \qquad [7.9]$$

cuja propriedades são:

$$\bar{b}_i(k) \geq 0 \;;\; 1 \leq i \leq N \;;\; 1 \leq k \leq M \;;\; \sum_{k=1}^{M} \bar{b}_i(k) = 1 \qquad [7.10]$$

7.4 Correlação

A medida para o grau de correção entre duas variáveis, ***x*** e ***y***, é chamada de coeficiente de correlação, representada por ***r***, definida por:

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{\left(n\sum x^2 - \left(\sum x\right)^2\right)\left(n\sum y^2 - \left(\sum y\right)^2\right)}} \qquad [7.11]$$

O valor de ***r*** irá sempre variar entre -1 e 1. Caso seja negativo, dizemos que a correção é negativa, ou seja, ***x*** e ***y*** crescem em sentidos opostos. Sendo positivo, dizemos que a correlação é positiva, ou seja, ***x*** e ***y*** crescem no mesmo sentido.

8. Wavelets vs. LPC

Neste capítulo vamos analisar e comparar o algoritmo utilizado e o desempenho alcançado por um sistema de reconhecimento de voz utilizando o algoritmo LPC (denominado sistema LPC) e utilizando a transformada *wavelet* (denominado sistema *wavelet*), ambas definidas no capítulo 6.

Para cada etapa do processo, vamos descrever os algoritmos utilizados para ambos os sistemas. Ambas implementações foram desenvolvidas no MATLAB VERSÃO 7.4.0.287 (R2007a) para Windows XP.

Vale ressaltar que o objetivo dos sistemas é relacionar uma palavra captada pela entrada do sistema com a palavra mais semelhante salva em seu *codebook*. O sistema desenvolvido possui a mesma estrutura da **Figura 3**, no capítulo 4.

8.1 Digitalização do sinal de voz

Tanto no sistema LPC como no sistema *wavelet* o processo de digitalização do sinal de voz foi idêntico no intuito de reproduzir o mesmo ambiente para avaliar o desempenho das duas técnicas utilizadas. A aquisição do sinal foi feita por placa de áudio comum em microcomputadores.

Para a captação do sinal de áudio utilizamos o código:

```
ai=analoginput('winsound');          % cria um objeto para entrada analógica
canal=addchannel(ai,1:1);            % adiciona um canal de hardware
set(ai,'SampleRate', 22050);         % taxa de amostragem em 22050 Hz
tx=get(ai,'SampleRate');             % tx = o valor da taxa  de amostragem
set(ai,'TriggerChannel',canal);      % define o canal para a aquisição
```

```
set(ai, 'SamplesPerTrigger', tx*1.5); % define duração da amostragem em 1,5s
set(ai,'TriggerType','Software');       % define o controle através do software
set(ai,'TriggerCondition','Rising');    % define aquisição com voltagem >0 Volts
set(ai,'TriggerConditionValue',0.05); % define voltagem >0.05Volts em aquisição
start(ai);                              % inicia a gravação
wait(ai,10);                            % encerra a execução após 10 s
 [dados] = getdata(ai);                 % retorna os dados adquiridos
```

Vimos no capítulo 5 que temos que seguir o Teorema de *Nyquist* para evitar efeito *aliasing*. Utilizamos uma freqüência Fs de 22050Hz.

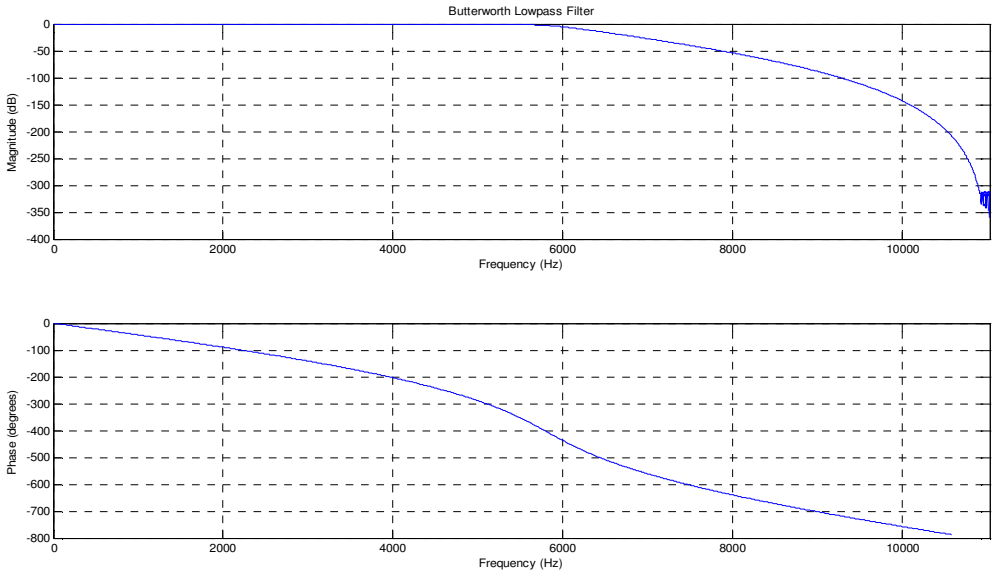

Figura 23 - Fase e magnitude do filtro

No processo de filtragem foi utilizado o filtro de *Butterworth* com o código:

```
Wp = 5500/Fs;       % frequência de corte
Ws = 7500/Fs;       % frequência onde a atenuação é de 40dB
rs = 40;            % atenuação na frequência de 7500 Hz
rp = 3;             % máximo ripple
```

```
[n,Wn] = buttord(Wp,Ws,rp,rs);
[b,a] = butter(n,Wn); % b, a: vetores com os coeficientes do filtro
```

Os parâmetros **n** e **Wn** são a ordem do filtro e o escalar da freqüência de corte, respectivamente. A **Figura 23** mostra o gráfico do filtro utilizado.

8.2 Extração de parâmetros do sinal de voz

No intuito de criar o mesmo ambiente e igualar os algoritmos ao máximo, é a partir desta etapa do reconhecimento de voz em que os sistemas diferem entre si.

8.2.1 Extração de parâmetros no sistema LPC

O código utilizado para o sistema LPC foi:

f = lpc(signal_filter, n);

Esta função determina os coeficientes de um preditor linear, minimizando o erro de previsão no sentido dos mínimos quadrados tendo aplicações em projetos de filtragem e de codificação de fala. Tem como parâmetros:

- signal_filter: uma matriz contendo os dados do sinal filtrado.

- n: ordem do preditor linear.

Utilizamos um preditor linear de ordem dez.

8.2.2 Extração de parâmetros no sistema *wavelet*

O código utilizado para o sistema *wavelet* foi:

[C, L] = wavedec(dados_f,n,'wname');

Esta função decompõe o sinal armazenado em **dados_f** em sua transformada *wavelet* de acordo com a **Figura 24**. Os dados que modelam o comportamento desta função são:

- dados_f: matriz com o sinal filtrado.

- n: quantidade de níveis de decomposição da transformada.

- 'wname': especifica de que família será a transformada *wavelet* utilizada. Pode ser *Haar*(wname = db1 ou haar) ou *Daubechies* (wname = dbX, com X>1).

Como vimos no capítulo 6, a forma *Daubechies* possui uma melhor resolução em freqüência e está concentrada nas baixas freqüências, ou seja, é mais adequada ao sistema que queremos desenvolver. Assim, utilizado um valor de 'wname' igual a db10. Os parâmetros C e L estão especificados na **Figura 24**.

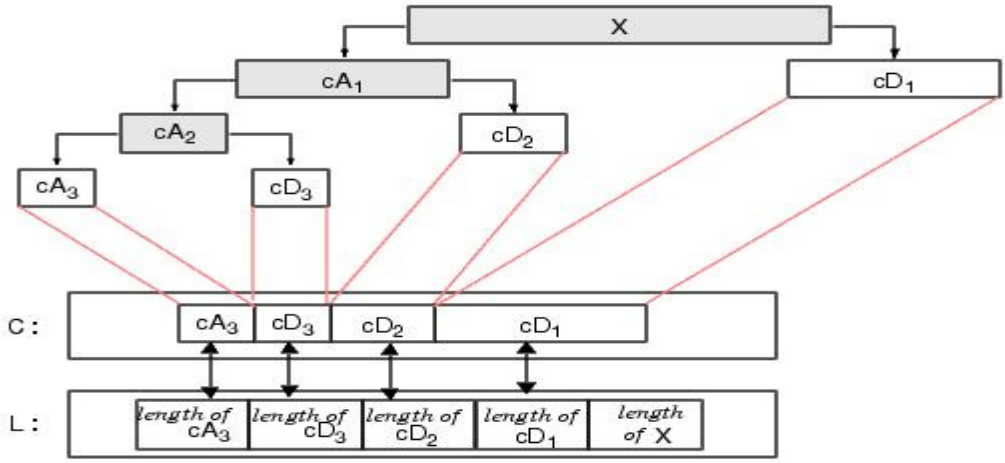

Figura 24 - Decomposição do sinal pela *wavedec*

8.3 Algoritmo de reconhecimento de voz

Utilizamos a correlação, descrita no capítulo 7, em ambos os sistemas para reconhecimento. Para decidir qual a palavra mais semelhante no *codebook* correlacionamos cada escala C do sinal de entrada com sua respectiva contida no

sinal armazenado, ou seja, correlacionamos, de acordo com a **Figura 24**, o campo $\mathbf{cA_3}$ do sinal de entrada com o campo $\mathbf{cA_3}$ do sinal salvo anteriormente no *codebook* e o mesmo foi feito com para os demais campos.

A função utilizada para correlacionas as escalas segue:

corr(A', B');

8.4 Implementações adicionais

Como um dos objetivos deste trabalho é aumentar a eficiência do algoritmo *wavelet* para reconhecimento de voz, implementações adicionais foram incluídas com este fim. A seguir detalharemos cada uma delas:

1. Normalização do sinal: foi feito um trabalho de normalização do sinal para que a discrepância existente devido à diferença das intensidades de sinais não fosse determinante na hora do reconhecimento. Para isso dividimos todo o sinal pelo valor mais elevado de amostra existente nele. O código a seguir foi utilizado:

array_sinal = array_sinal/max(array_sinal);

2. Retirada do silêncio: este passo foi adicionado para não compararmos sinais semelhantes com uma defasagem causada por conta de uma pausa durante a fala. Na **Figura 25**, por exemplo, podemos afirmar que os sinais A, B e C seriam idênticos se não fosse apenas sua diferença de defasagem.

Para isso removemos toda amostra do sinal inferior a 1% da amostra de maior intensidade do sinal.

3. Interpolação do sinal: este passo foi necessário porque, após a retirada do silêncio do sinal, nunca teríamos dois sinais de mesmo tamanho para comparar. Para que isto não acontecesse, inserimos ou retiramos amostras do sinal a ser

reconhecido de forma interpolada para que este não perdesse muito da sua identidade ao ser feita a operação de reconhecimento.

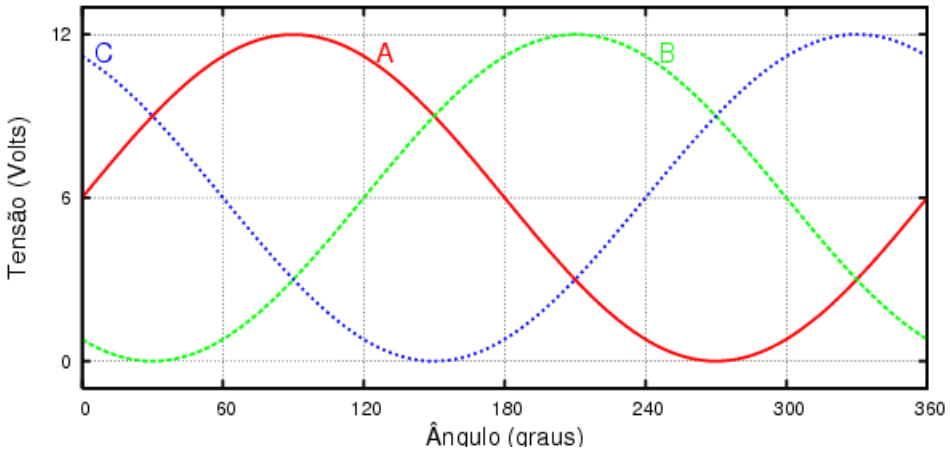

Figura 25 - Sinais defasados

8.5 Análise comparativa de desempenho

Utilizamos para ambos os algoritmos algumas combinações de *codebooks* com vozes masculinas e femininas totalizando uma população igual a sete indivíduos. Os testes foram feitos da seguinte forma:

- Para geração do *codebook*, o usuário pronuncia, separadamente, quatro amostras de sua voz com as seguintes palavras: esquerda, direita, alto, baixo gravando-as no *codebook*.

- Para reconhecimento, o usuário pronuncia uma das palavras ao captador de som (microfone).

- O sistema reproduzirá o palavra do *codebook* mais semelhante a palavra pronunciada.

Não houve diferenças significativas entre o processo de reconhecimento de voz com os pares masculino-masculino e feminino-feminino. Os resultados dos testes podem ser vistos a seguir.

8.5.1 Resultados com algoritmo LPC

Resultado para par masculino-feminino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	40%
Direita	30%
Alto	45%
Baixo	55%

Resultado para par masculino-feminino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	45%
Direita	40%
Alto	45%
Baixo	60%

Resultado para par masculino-masculino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	80%
Direita	65%
Alto	70%
Baixo	80%

Resultado para par masculino-masculino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	90%
Direita	85%
Alto	85%
Baixo	95%

8.5.2 Resultados com algoritmo *Wavelet* sem implementações adicionais

Resultado para par masculino-feminino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	45%
Direita	30%
Alto	40%
Baixo	45%

Resultado para par masculino-feminino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	40%
Direita	40%
Alto	45%
Baixo	55%

Resultado para par masculino-masculino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	75%
Direita	60%
Alto	75%
Baixo	80%

Resultado para par masculino-masculino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	75%
Direita	75%
Alto	80%
Baixo	90%

8.5.3 Resultados com algoritmo *Wavelet* com implementações adicionais

Resultado para par masculino-feminino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	45%
Direita	30%
Alto	40%
Baixo	45%

Resultado para par masculino-feminino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	40%
Direita	40%
Alto	45%
Baixo	55%

Resultado para par masculino-masculino com apenas uma amostra por palavra gravada no *codebook*.

Som	Taxa de acerto
Esquerda	90%
Direita	65%
Alto	75%
Baixo	95%

Resultado para par masculino-masculino com duas amostras por palavra gravadas no *codebook*.

Som	Taxa de acerto
Esquerda	95%
Direita	90%
Alto	95%
Baixo	100%

9. Conclusão e trabalhos futuros

O trabalho realizado teve como principais objetivos comparar implementações de métodos desenvolvidos para viabilizar um reconhecimento de voz satisfatório e demonstrar alguns dos métodos biométricos de reconhecimento viáveis e desenvolvidos.

Percorremos os conceitos teóricos de sistemas de reconhecimento biométricos e apresentamos os mais utilizados e complexos. A partir disso, apresentamos como é feito o reconhecimento de voz, suas vantagens, suas desvantagens e suas principais dificuldades que o tornam um dos métodos mais confiáveis e de maior desafio de implementação.

Analisamos o processo de reconhecimento de modo global e discorremos sobre cada etapa, citando os algoritmos mais utilizados em cada uma delas e suas dificuldades de implementação e utilização. Por fim, comparamos resultados de dois diferentes métodos utilizados na extração de parâmetros de um sinal e auxiliamos na melhoria de um deles.

Apresentamos as soluções propostas em MATLAB o que nos dá espaço para implementar, futuramente, o mesmo código para alguma plataforma com menor recurso computacional como DSPs e microcontroladores de poder computacional bem menor que os computadores mais utilizados.

Com os resultados obtidos, que são apresentados no capítulo 8, podemos concluir:

- A diferença entre as taxas de acerto dos algoritmos apresentados (LPC e *wavelet*) para reconhecer um locutor de um sexo A com amostras de um locutor de um sexo B, não é significativa.

- Dobrando o tamanho do *codebook*, a diferença entre taxas de acerto para reconhecer um locutor de um sexo A com amostras de um locutor de um sexo B, não é significativa se comparada ao tamanho original do *codebook*.

- O percentual de acerto aumentou de forma semelhante para ambos os algoritmos analisados (LPC e *wavelet*) quando dobramos o tamanho do *codebook*.

- O algoritmo *wavelet* obteve uma taxa de acerto superior ao algoritmo LPC em todos os aspectos quando o locutor a ser reconhecido é do mesmo sexo do locutor que gravou as amostras.

- As implementações adicionais feitas no algoritmo *wavelet* aumentou, em média, 9% as taxas de acerto quando o locutor a ser reconhecido é do mesmo sexo do locutor com uma amostra e aumentou, em média, 15% com duas amostras.

- A baixa taxa de reconhecimento para locutores de sexo distintos se dá pelo fato dos métodos LPC e *wavelet* utilizarem uma análise de freqüência nos sinais e, como homens e mulheres possuem grandes diferenças neste ponto, as taxas de acerto são baixas.

Mesmo atingindo boas taxas de reconhecimento, os estudos mostraram que o reconhecimento de voz ainda tem muito a evoluir. Utilizar meios como inteligência computacional pode aumentar a eficiência e, com isso, este tipo de reconhecimento tem um grande potencial de atingir um de seus principais objetivos que é melhorar a qualidade de vida de pessoas com deficiências físicas. Aliando-se a robótica, por exemplo, é possível realizar muitas das tarefas caseiras mais árduas com simples comando de voz.

10. Referências Bibliográficas

[1] Oppenhein, A. V., Schafer, R. W. Discrete Time Signal Processing, 2.ed. New York: Prentice Hall, 2002.

[2] Haykin, Simon, Sinais e Sistemas, Porto Alegre: Bookman, 2001.

[3] Tolba, O'Shaughnessy. Speech Recognition by Intelligent Machines, IEEE Press, 20-23, 2001.

[4] Bergadano, F., Gunetti, D., and Picardi, C. User authentication through keystroke dynamics. ACM Transactions on Information and System Security, 2002.

[5] Leniski, A. C., Skinner, R. C., McGann, S. F., and Elliott, S. J. Securing the biometric model. In IEEE 37th Annual 2003 International Carnahan Conference on Security Technology, 2003.

[5] Przybocki, M. and Martin, A. (2004). NIST speaker recognition evaluation chronicles. Technical report, Speech Group, Information Access Division, Information Technology Laboratory National Institute of Standards and Te chnology, USA. Published in the Odissey 2004 Conference.

[6] J. Campbell, "Speaker Recognition: A Tutorial," Proc. IEEE, vol. 85, pp. 1437–1462, 1997.

[7] D. A. Reynolds, L. P. Heck, "Automatic Speaker Recognition - Recent Progress, Current Application, and Future Trends". AAAS 2000 Meeting Humans, Computer and Speech Symposium, 2000.

[8] Picone, J. Signal Modeling Techniques in Speech Recognition, Proceedings of IEEE, 81, nr. 9, 1993.

[9] Rabiner, L. R., Juang B. H. Fundamentals of speech recognition. Prentice Hall, 1993.

[10] Petry, A. Reconhecimento Automático de Locutor Utilizando medidas de invariantes dinâmicas não-lineares. Tese de doutorado. UFRS, 2002.

[11] Rabiner, L. R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, Vol. 77, no.2, 1989.

[12] Furui, Sadaoki. Digital Speech Processing, Synthesis, and Recognition. New York, Marcel Dekker, 1989.

[13] Maciel, R.C.V. - Melhoria da Qualidade de Sinais de Fala Degradados por Ruído através da Utilização de Sinais Sintetizados. Escola Politécnica da Universidade de São Paulo, Dissertação de Mestrado. São Paulo, 2003.

[15] A. Piyush Shanker and A.N. Rajagopalan, Off-line signature verification using DTW, Pattern Recognition Letters, 2007.

[16] Chau, F.T., Liang, Y.Z., GAO, J., SHAO, J., SHAO, X.G., Chemometrics: From basics to Wavelet Transform. Nova Jersey: Wiley-Interscience, 2004.

[17] Donoho, D.L., Johntone, I.M., Ideal spatial adaptation via wavelet shrinkage, 1994.

[18] Tanyer, S., Ozer, H. Voice activity detection in nonstationary noise. IEEE Transactions on Speech and Audio Processing, v. 8, n. 4, 2000.

[19] Liu, S.S., "A Practical Guide to Biometric Security Technology", IEEE Computer Society, 2001.

[20] Rowley, H., Baluja, S., Kanade, T. Rotation invariant neural network-based face detection. IEEE Computer Society, 1998.

[21] Martins, J. A. Avaliação de diferentes técnicas para reconhecimento de fala. Tese de doutorado. UNICAMP, SP, 1997.

[22] Grassi, S. Efficient algorithm to compute lsp parameters from 10th-order lpc coefficients. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Washington, DC, USA: IEEE Computer Society, 1997.