UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

RECONHECIMENTO DE VOZ PARA PALAVRAS ISOLADAS

TRABALHO DE GRADUAÇÃO

Aluno: Anderson Gomes da Silva {ags@cin.ufpe.br}

Orientador: Tsang Ing Ren {tir@cin.ufpe.br}

Recife, Dezembro de 2009

UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

RECONHECIMENTO DE VOZ PARA PALAVRAS ISOLADAS

ANDERSON GOMES DA SILVA

Monografia apresentada ao Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do título de engenheiro da computação.

Orientador: Tsang Ing Ren

Recife, Dezembro de 2009

“Um passo à frente, e você não está mais no mesmo lugar.”

Chico Science

Agradecimentos

Em primeiro lugar, agradeço a Deus em Quem confio muito e está sempre presente em minha vida, pois sem Ele eu não teria conseguido chegar até aqui.

Agradeço a meus pais, Geraldo e Fátima, que sempre investiram e depositaram muita confiança em mim, sendo os grandes responsáveis por tudo o que já conquistei. Ao meu irmão, André, que acima de tudo é um amigo.

Ao meu orientador, Tsang, pela confiança depositada em mim e pelo apoio no desenvolvimento deste e de outros trabalhos, mas principalmente pela oportunidade.

A minha namorada, Rebeca, que me incentivou bastante durante o período de desenvolvimento deste projeto e soube compreender a minha ausência devido à dedicação ao mesmo.

Aos meus amigos que conviveram comigo durante estes anos de graduação e a todos os outros que desejam o meu sucesso e que eu posso sempre confiar.

Aos professores do Centro de Informática que contribuíram bastante para minha formação.

Por fim, gostaria de agradecer a todos que acreditaram em mim e que de alguma forma me ajudaram.

Resumo

O reconhecimento automático da fala tem sido objeto de estudo dos pesquisadores por mais de quatro décadas e já tem sua importância e espaço garantidos no mundo atual. Uma das maiores dificuldades encontradas nesta área é a sua natureza interdisciplinar, além de que os sistemas implementados, em geral, devem estar aptos a funcionar em condições de ruído de fundo, o que exige um estudo de técnicas para conseguir a robustez do sistema.

Nos sistemas que constituem o estado da arte na área de reconhecimento de voz predominam os modelos estatísticos, especialmente os baseados nos modelos ocultos de Marcov ou HMM (*Hidden Markov Models*). Os HMM's são estruturas poderosas, pois são capazes de modelar ao mesmo tempo as variabilidades acústicas e temporais do sinal de voz.

O objetivo deste trabalho é o desenvolvimento de um sistema de reconhecimento de palavras isoladas baseado em HMM. Ele deve reconhecer os dígitos de 0 a 9 e as palavras “sim” e “não”, porém deverá ser possível expandi-lo facilmente para conseguir reconhecer outras palavras. O sistema é formado por quatro blocos principais: aquisição do sinal de fala, pré-processamento, extração de parâmetros e HMM. Estes blocos são descritos no decorrer deste relatório.

Sumário

Lista de figuras

Lista de tabelas

1. Introdução

A linguagem oral é o modo natural de comunicação do ser humano e também o mais rápido [5]. Isto motiva o estudo de sistemas de reconhecimento e síntese de voz a fim de se criar uma interface homem-máquina mais amigável e simples de usar a partir da comunicação oral, permitindo assim o uso de computadores e outros aparelhos eletrônicos por mais pessoas. Em virtude desse fato, grandes esforços têm sido realizados para a obtenção de sistemas capazes de entender de se comunicar através da fala. Deste modo diversas técnicas têm sido desenvolvidas e aprimoradas com o objetivo de obter melhores resultados nessa categoria de algoritmos. Com o progresso do poder de processamento dos computadores e sistemas embarcados esta área cresce a cada dia. Porém, apesar desses esforços, está longe de existir um sistema capaz de compreender um discurso sobre qualquer assunto, falado de forma natural, por qualquer pessoa, em qualquer ambiente.

Devido à maturidade e eficiência dos algoritmos e métodos de reconhecimento de voz, o seu estudo tem importância também para outras áreas como biometria, visão computacional e reconhecimento de padrões em geral.

1.1. Definindo o problema

Reconhecimento automático da fala funciona a partir da conversão de um sinal acústico produzido pelo homem em um sinal digital de áudio através de um hardware associado a um software que a partir de uma base de dados identificará o conjunto de palavras faladas. As palavras reconhecidas podem ser o resultado final do sistema como no caso de aplicações de comandos de controle ou servir de entrada a outros sistemas.

Uma das maiores dificuldades da área de reconhecimento de voz é a sua natureza interdisciplinar [1]. Dentre as áreas envolvidas estão processamento de sinais, ciências da computação, reconhecimento de padrões, inteligência artificial, neurofisiologia, lingüística, teoria das comunicações, fonética articulatória e acústica. Além disto, os sistemas de reconhecimento da fala, em geral, devem ser aptos a funcionar em condições de ruído de fundo, o que exige um estudo de técnicas para conseguir a robustez do sistema.

Durante os últimos anos tem se observado uma grande evolução na área de reconhecimento de voz. Já existem sistemas bastante eficientes, entretanto nenhum deles é independente de limitações e nem funcionam com uma taxa de 100% de acerto. Dentre as razões para este significativo avanço podem ser citadas: desenvolvimento de novas técnicas de processamento digital de sinais, a disponibilidade de computadores rápidos e mais baratos, a instituição de padrões para avaliação de desempenho e uma maturidade alcançada em algumas técnicas como Modelos Ocultos de Markov (HMM), Modelos de Mistura Gaussiana (GMM) e Redes Neurais Artificiais (RNA).

Certamente, a variabilidade dos sinais de fala é o principal limitador de desempenho dos sistemas de reconhecimento. Esta variabilidade se deve a diversos fatores, dentre eles:

- A variabilidade dos sons para um único locutor e entre locutores diferentes;
- A variabilidade do transdutor e do canal, como microfones, telefones fixos e celulares;
- A variabilidade do ruído de fundo gerado a partir de outras vozes, carros, ar-condicionado, dentre outros;
- A variabilidade na produção da fala incluindo barulhos resultantes de movimentos da boca, ruídos de respiração, hesitações ao falar, etc.

Em geral, estas fontes de variabilidade não podem ser eliminadas, devendo, portanto, ser modeladas diretamente pela tecnologia de reconhecimento de fala.

1.2. Aplicações do reconhecimento automático da fala

Sistemas de reconhecimento de voz têm aplicações em diversas áreas. Na realidade qualquer atividade que envolva interação humano-máquina pode potencialmente utilizar estes sistemas. Atualmente várias aplicações já estão sendo concebidas com um sistema de reconhecimento de fala incorporado. Dentre as áreas mais comuns encontra-se:

- Sistemas de controle e comando: estes sistemas utilizam a fala para realizar determinadas funções;
- Sistemas de telefonia: o usuário pode utilizar a voz para fazer uma chamada, ao invés de discar o número;

- Sistemas de transcrição: textos falados pelo usuário podem ser transcritos automaticamente por estes sistemas;
- Centrais de atendimento ao cliente: uma atendente virtual pode ser utilizada a fim de realizar o atendimento ao cliente;
- Robótica: robôs podem se comunicar pela fala com seus donos.

1.3. Visão geral do trabalho

Dentro desse contexto, buscou-se neste trabalho, desenvolver e implementar um sistema de reconhecimento de voz que obtivesse uma alta taxa de acerto, voltado para uma aplicação prática. A aplicação de reconhecimento de palavras isoladas com vocabulário pequeno é a opção ideal para se iniciar estudos nessa área, pois permite desenvolver a base de conhecimento necessária para se trabalhar em aplicações mais complexas, como o reconhecimento de fala contínua, além de ter muitas aplicações como em sistemas de controle de comando. Desenvolveu-se então um sistema desse tipo baseado nos modelos ocultos de Markov, também conhecidos como HMM (*Hidden Markov Models*).

O trabalho está dividido em sete capítulos descritos a seguir.

O capítulo 2 tem por finalidade discorrer sobre o histórico e as características mais comuns dos sistemas de reconhecimento de fala.

No capítulo 3 apresentam-se as principais etapas do processamento inicial do sinal de fala, o qual objetiva a extração de parâmetros do sinal capazes de diferenciar de forma eficiente, os eventos da fala.

No capítulo 4 é apresentada a teoria dos modelos ocultos de Markov, juntamente com aspectos relacionados com a implementação de um sistema de reconhecimento de padrões baseado nestes modelos. São apresentados algoritmos que possibilitam calcular a probabilidade de um modelo HMM gerar uma sequência de observações e determinar os parâmetros de um HMM para modelar uma sequência de observações.

O capítulo 5 descreve o desenvolvimento de um sistema de reconhecimento de palavras isoladas baseado em HMM. O principal objetivo deste capítulo é mostrar como a teoria básica

apresentada nos capítulos anteriores pode ser utilizada, de forma simples, para o desenvolvimento de um sistema de reconhecimento de fala.

No capítulo 6 são apresentados experimentos realizados e os resultados obtidos a partir deles.

No capítulo 7 é apresentada uma conclusão a respeito do trabalho realizado e possibilidades de trabalhos futuros que possam ter como ponto de partida os resultados deste trabalho.

2. Sistemas de reconhecimento de fala

Reconhecimento da fala consiste no processo de conversão de um sinal acústico produzido pelo homem em um sinal digital de áudio através de um hardware associado a um software que a partir de uma base de dados identificará o conjunto de palavras faladas. As palavras reconhecidas podem ser o resultado final do sistema como no caso de aplicações de comandos de controle ou servir de entrada a outros sistemas.

2.1. Um breve histórico

O reconhecimento automático da fala tem sido objeto de estudo dos pesquisadores por mais de quatro décadas [1] e já tem sua importância e espaço garantidos no mundo atual. A literatura reporta que o primeiro estudo nesta área foi realizado no ano de 1952, por pesquisadores dos laboratórios Bell, os quais criaram o primeiro sistema de reconhecimento de dígitos isolados adaptado para um único locutor. Este sistema era baseado nas medidas de ressonâncias espectrais das vogais de cada dígito [3].

Nas décadas de 50 e 60, as principais estratégias de reconhecimento de voz baseavam-se na segmentação do sinal acústico em fonemas (unidades básicas da pronúncia), identificar os fonemas, baseados em análise espectral e transcrever o fonema reconhecido. No final dos anos 60 pesquisadores do NTT Labs formularam os conceitos da técnica Linear Predictive Coding (LPC), o que simplificou bastante a análise de voz [4].

Em meados da década de 70, a técnica LPC foi introduzida no reconhecimento da fala (até então só tinha sido utilizada na codificação), por Rabiner e Levinson, Itakura e outros. Nesta época, o paradigma dominante para o reconhecimento da fala com vocabulário pequeno era o Dynamic Time Warping (DTW), proposto por Vintsyuk, como um método para calcular a similaridade entre duas sequências temporais (e.g. sentenças faladas) [5]. Com a técnica DTW, bons resultados para reconhecimento de palavras isoladas com vocabulário pequeno foram alcançados e apareceram os primeiros sistemas de reconhecimento de voz comerciais.

Na década de 80 surgiram os métodos estatísticos para reconhecimento de fala. O mais utilizado destes métodos é o baseado em Modelos Ocultos de Markov (*Hidden Markov Models* ou HMM), cujo a teoria já era conhecida há vários anos, porém ainda não

tinha sido utilizada em reconhecimento de voz. HMM tornou-se desde então a técnica mais utilizada no reconhecimento de fala. Ainda nesta década foi introduzida a técnica de redes neurais para o reconhecimento de voz.

Da década de 90 aos dias atuais, os estudos estão mais voltados para o reconhecimento de fala contínua irrestrita, com vocabulário ilimitado e independente do locutor. A fim de alcançar este objetivo, muitas pesquisas buscam a resolução de problemas como robustez ao ruído de fundo, adaptação ao locutor, diferença de pronúncias, distorções introduzidas pelo canal de transmissão, como o telefone, etc.

2.2. Tipos de reconhecimento de fala

Um sistema de reconhecimento automático da fala pode ser caracterizado de várias maneiras. As mais importantes são estão relacionadas ao estilo de pronúncia que aceita, ao tamanho do vocabulário e à dependência ou independência do locutor [6]. A precisão dos sistemas é fortemente influenciada por estes fatores. As classificações dos sistemas quanto a estas características encontram-se abaixo.

Quanto à dependência de locutor:

- Dependente de locutor: reconhece a fala das pessoas cujas vozes foram utilizadas para treinar o sistema.
- Independente de locutor: reconhece a fala de qualquer pessoa com uma taxa de acerto aceitável. Neste caso é necessário realizar o treino do sistema com uma base que inclua diferentes pessoas com diferentes idades, sexo, sotaques, etc.

Quanto ao tamanho do vocabulário:

- Vocabulário pequeno: reconhecem até 20 palavras.
- Vocabulário médio: reconhecem entre 20 e 100 palavras.
- Vocabulário grande: reconhecem entre 100 e 1000 palavras.
- Vocabulário muito grande: reconhecem mais de 1000 palavras.

Quanto ao modo de pronúncia:

- Reconhecedor de palavras isoladas: estes sistemas reconhecem palavras faladas isoladamente, isto é, entre cada palavra deve existir uma pausa mínima, para que seja detectado o início e o fim da mesma. Estes sistemas são os mais simples de serem implementados.
- Reconhecedor de palavras conectadas: são sistemas um pouco mais complexos que os de palavras isoladas e utilizam palavras como unidade fonética padrão. São capazes de reconhecer sentenças completas pronunciadas sem pausa entre as palavras, porém estas devem ser bem pronunciadas.
- Reconhecedor de fala contínua: são capazes de reconhecer a fala na comunicação natural, sem nenhuma particularidade quanto à pronúncia. Estes sistemas são bastante complexos, pois precisam lidar com todas as características e vícios da linguagem natural, como o sotaque, a duração das palavras, a pronúncia descuidada, etc.

2.3. O sistema de reconhecimento de fala

Um reconhecedor de voz tem como entrada um sinal de fala obtido a partir de um transdutor. A partir dessa entrada é realizado um mapeamento a fim de descobrir a palavra falada, ou seja, transcrever o que foi falado.

Na literatura é bastante comum representar o funcionamento de um sistema de reconhecimento automático da fala através de um diagrama de blocos, conforme o exibido na figura 1, facilitando assim o seu entendimento.

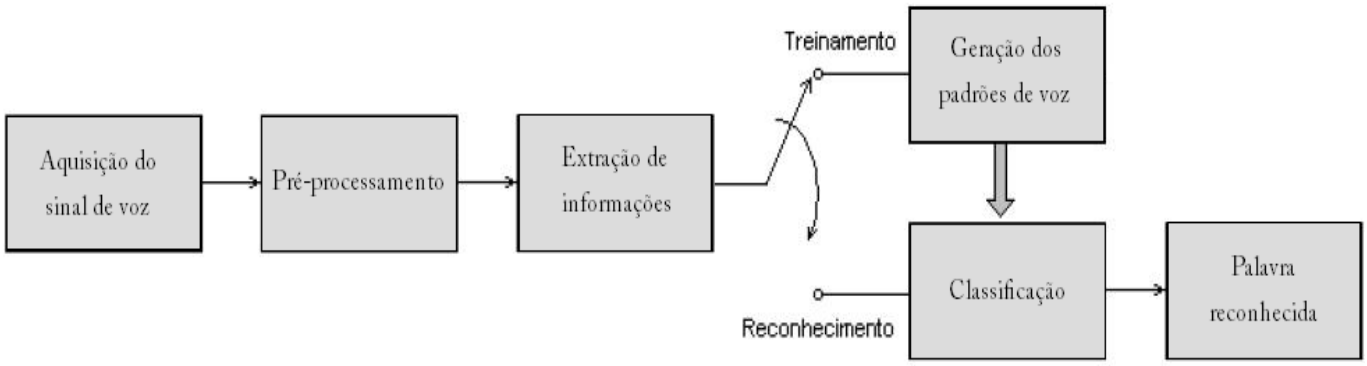

Figura 1 - Diagrama de blocos de um sistema de reconhecimento de voz

O sistema é dividido em quatro etapas, aquisição do sinal de voz, pré-processamento, extração de informações e a última que pode ser a geração dos padrões de voz, quando na fase de treinamento ou a classificação, quando na fase de reconhecimento, que utiliza os padrões de voz gerados na fase de treinamento.

3. Processamento do sinal acústico

O processamento do sinal acústico é a parte básica do sistema de reconhecimento de voz, compreendendo desde a etapa de aquisição do sinal de voz até a etapa de extração das características do sinal relevantes ao reconhecimento, que servirão de entrada para a próxima etapa, onde é realizado de fato o reconhecimento.

3.1. Aquisição da fala

A primeira etapa consiste em realizar a aquisição do sinal de voz através da conversão das ondas sonoras em sinais elétricos a partir de um transdutor, filtragem desse sinal e conversão analógico-digital do mesmo, como pode ser visto na figura 2 [7].

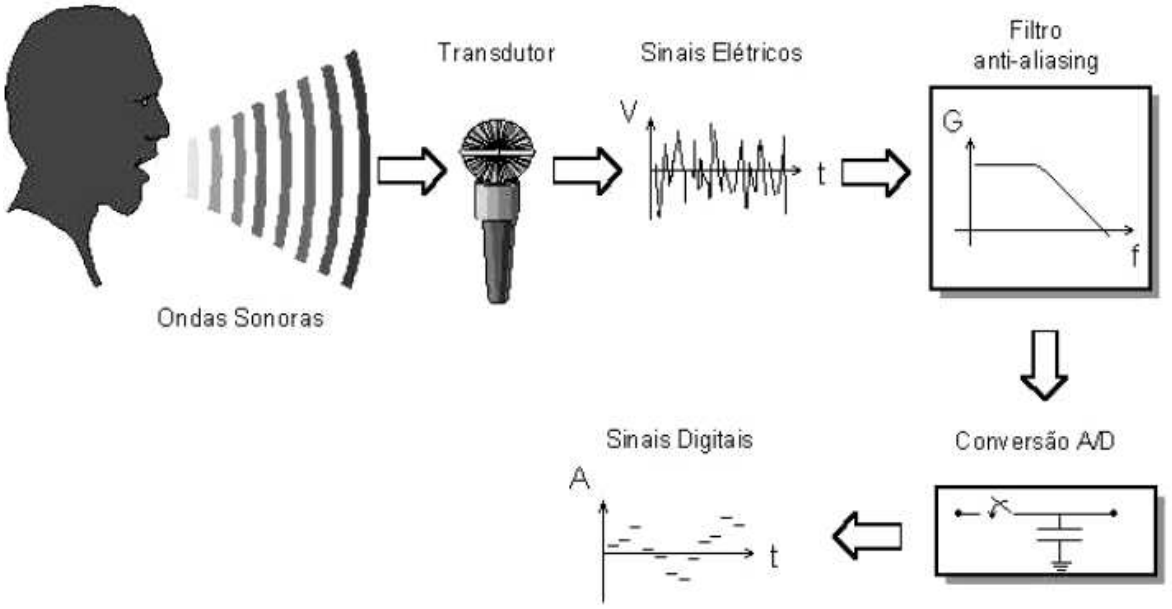

Figura 2 - Processo de aquisição do sinal da fala

A conversão do sinal acústico em ondas elétricas é realizada através de um transdutor, que normalmente é um microfone ou um telefone. Realizada esta conversão é necessário filtrar o sinal analógico resultante através de um filtro passa-baixas, chamado filtro *anti-aliasing*, antes de realizar a conversão analógico-digital [8]. Esta filtragem tem o intuito de suprimir as componentes de frequência superiores à metade da frequência de amostragem. Critério esse conhecido como critério de Nyquist [9, 12].

Por último é realizada a conversão do sinal de fala analógico em digital através de um amostrador, a fim de possibilitar o processamento digital do mesmo. É nesta etapa que são escolhidas a taxa de amostragem, de forma a assegurar a não ocorrência do efeito de *aliasing* [8, 10] e a precisão usada para a gravação do sinal, a partir do número de níveis que esse sinal poderá assumir, após ser amostrado. Esses níveis são representados por uma cadeia de bits e deve ser escolhido de forma a conseguir uma boa precisão. Quanto maior o número de níveis maior será a precisão. Uma cadeia de 16 bits por amostra, o que equivale a 65536 níveis, já é suficiente para sinais de voz.

3.2. Pré-processamento

O reconhecimento normalmente é atrapalhado por características que refletem o ambiente de gravação e o canal de comunicação, como ruídos de alta frequência, distância do microfone, períodos de silêncio, etc. Assim o sinal deve passar por um pré-processamento a fim de deixar o sinal mais próximo da fala pura. A metodologia a ser utilizada no pré-processamento pode ser vista no diagrama de blocos abaixo:

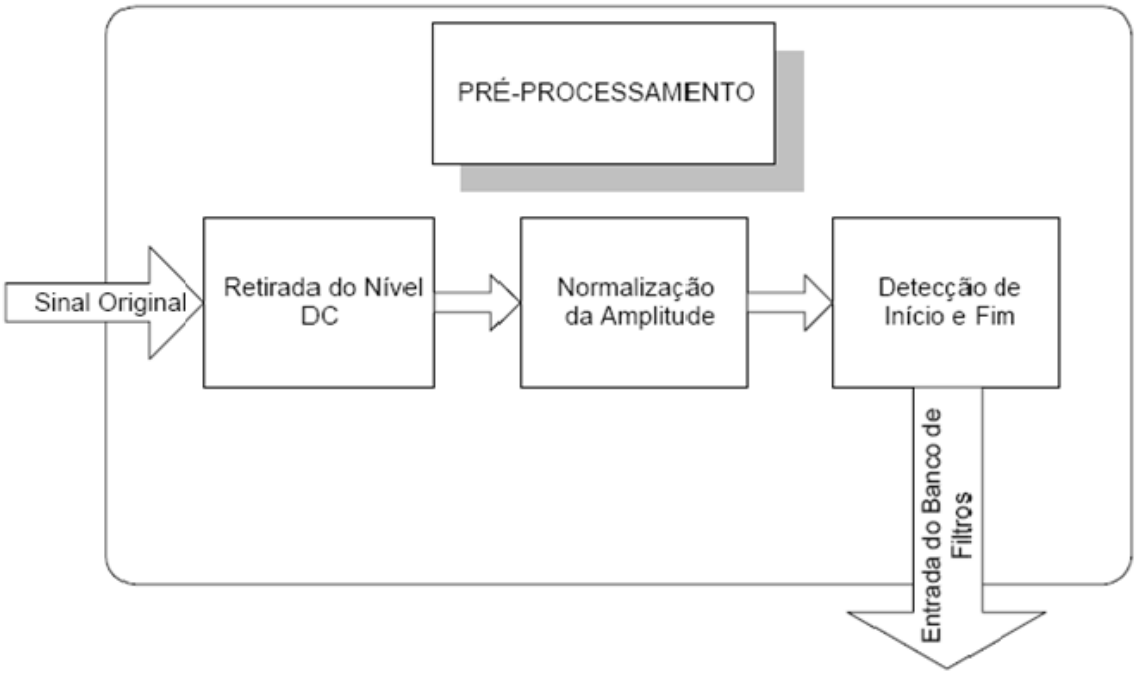

Figura 3 - Diagrama de blocos da fase de pré-processamento

Os sinais de voz apresentam, muitas vezes, uma componente contínua, o que atrapalha a comparação em valores absolutos, sendo necessário então a retirada desse nível DC, a fim de deixar

todas as amostras oscilando em torno do valor zero. Para realizar-se a retirada deste nível DC, calcula-se a média aritmética das amplitudes do sinal, e depois se subtrai de cada amplitude esta média.

A normalização da amplitude diz respeito à altura do som. Este pré-processamento do sinal faz com que todos os valores de amplitudes de todos os sinais estejam na mesma faixa de valores, que no caso deste trabalho é a faixa entre -1 e 1. Isto garante que todos os sinais sejam processados igualmente com relação ao volume da voz, ou seja, sons mais baixos e mais altos serão processados igualmente no algoritmo de reconhecimento. Para a realização desta normalização dividiu-se o valor de cada amostra do sinal pelo maior valor de amplitude do mesmo.

Por fim é realizada a detecção do início e fim da locução a fim de remover de forma precisa os períodos de silêncio existentes antes e após o sinal que além de não possuírem nenhuma informação relevante para o reconhecimento, podem conter ruídos, sinais indesejados e a duração dos mesmos pode ser variável, dificultando assim o reconhecimento. Esta detecção objetiva também a diminuir a carga computacional e economizar tempo, visto que o mesmo não terá que processar informações de trechos que não fazem parte da fala [11]. O extremo inicial é determinado pelo primeiro quadro onde realmente se inicia a fala, por enquanto que o extremo final é determinado pelo último quadro que ainda há fala. Neste trabalho foi utilizado o algoritmo proposto em [2], no qual a detecção dos extremos do sinal é realizada com base na energia e na taxa de cruzamentos por zero do mesmo. Este algoritmo é suficientemente robusto para operar em ambientes com taxa sinal-ruído (*signal-noise rate* ou SNR) de até 30 dB.

3.3. Extração de informações do sinal da fala

A etapa de extração de informações é de fundamental importância para o projeto de qualquer sistema de reconhecimento de fala, visto que o sinal digital possui uma grande quantidade de dados e consequentemente a sua análise direta além de exigir tempo e processamento consideráveis, provavelmente não apresentará resultados expressivos. Certamente muitas informações existentes no sinal digital puro são redundantes ou não possuem significância alguma para a distinção fonética. Sendo assim o classificador empregado dificilmente conseguirá diferenciar amostras de palavras distintas.

A idéia básica da extração de parâmetros é representar segmentos, fonemas ou qualquer outra unidade de fala com o menor número possível de parâmetros, de forma que estes contenham informações suficientes para caracterizar o sinal de fala. Por melhor que seja o classificador, este não apresentará bons resultados se os parâmetros utilizados durante o treinamento ou reconhecimento não contiverem informações relevantes. Uma redução no volume de dados de forma a fornecer apenas um conjunto pequeno de parâmetros, porém contendo informações suficientes para a caracterização do sinal, a um classificador viabilizará uma classificação robusta e confiável.

Os parâmetros tipicamente são obtidos a partir das seguintes técnicas de análise espectral: a transformada rápida de Fourier (*Fast Fourier Transform* ou FFT), os métodos de banco de filtros (*Filter Bank*), os de análise homomórfica ou análise cepstral (*mel-cepstrum*) e os de codificação por predição linear (*Linear Predictive Coding* ou LPC) [1, 8].

A técnica FFT, os métodos de banco de filtros e o LPC foram largamente utilizados para a análise espectral da fala, no entanto, elas possuem algumas restrições. A mais notável é a de não oferecer uma metodologia para a separação do sinal de excitação da resposta impulsiva do trato vocal, a qual é oferecida pela técnica de *mel-cepstrum*. Os coeficientes *mel-cepstrais* (*Mel-Frequency Cepstral Coefficients* ou MFCC), advindos desta técnica, são obtidos pela representação em freqüência na escala mel [1, 13], sendo considerada assim a técnica mais apropriada para ser utilizada no processo de reconhecimento de voz. Devido a isto, atualmente os coeficientes MFCC são os mais populares [14]. Com base nestas informações decidiu-se utilizar neste trabalho um modelo baseado na técnica de análise cepstral.

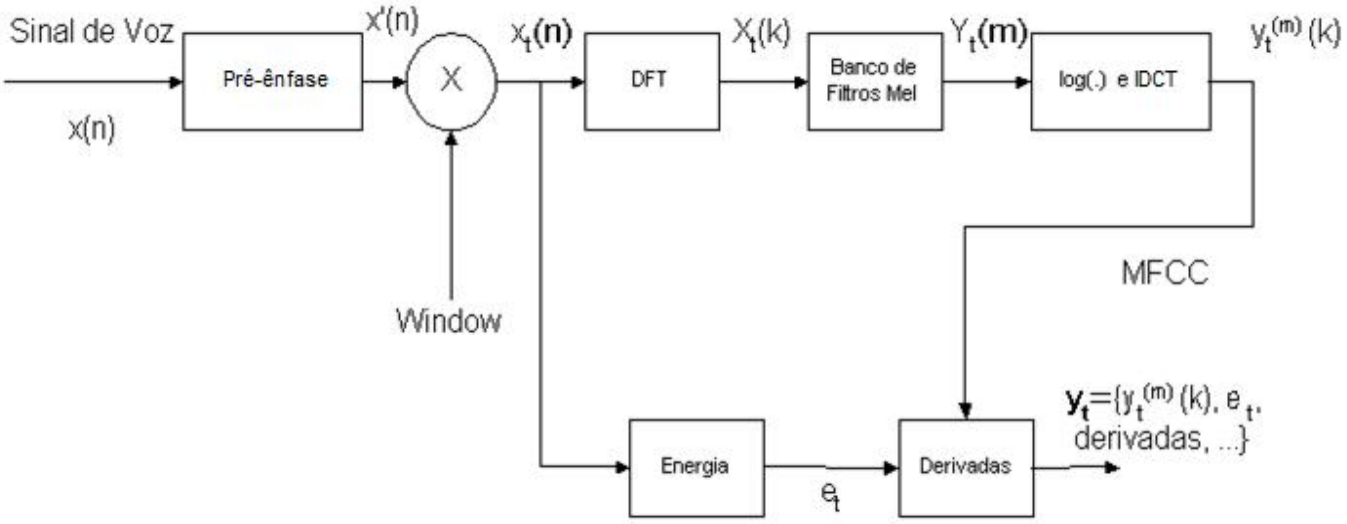

Figura 4 - Sistema de processamento de voz

A figura 4 mostra a arquitetura do sistema de extração de parâmetros utilizado. Alguns dos blocos serão melhores descritos a seguir.

3.3.1. Pré-ênfase

A finalidade da filtragem de pré-ênfase é compensar a atenuação de 6dB/oitava nas altas freqüências. Esta atenuação é ocasionada pelo efeito combinado do espectro decrescente dos pulsos glotais (-12dB/oitava) e pelo efeito de radiação dos lábios (+6dB/oitava) [6, 11]. Para a realização desta filtragem [1] utiliza-se um filtro passa-altas de primeira ordem com a seguinte função de transferência:

$$H(z) = 1 - \alpha z^{-1}$$

No domínio do tempo, o sinal de saída $x'(n)$ relaciona-se com o sinal de entrada $x(n)$ pela fórmula abaixo:

$$x'(n) = x(n) - \alpha x(n-1)$$

Neste trabalho utilizou-se $\alpha = 0.95$. Na figura 5 é ilustrada a resposta em frequência para o filtro de pré-ênfase.

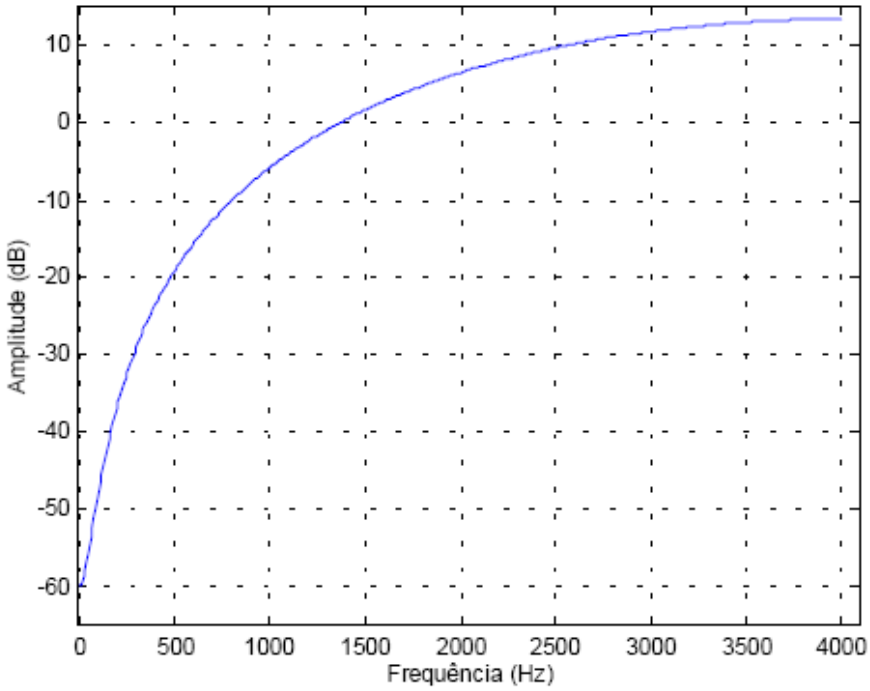

Figura 5 - Resposta em frequência do filtro de pré-ênfase para α = 0.95

A figura 6 mostra o espectro de um sinal de fala (a) sem antes de passar pelo filtro de pré-ênfase e (b) após passar pelo filtro de pré-ênfase [15].

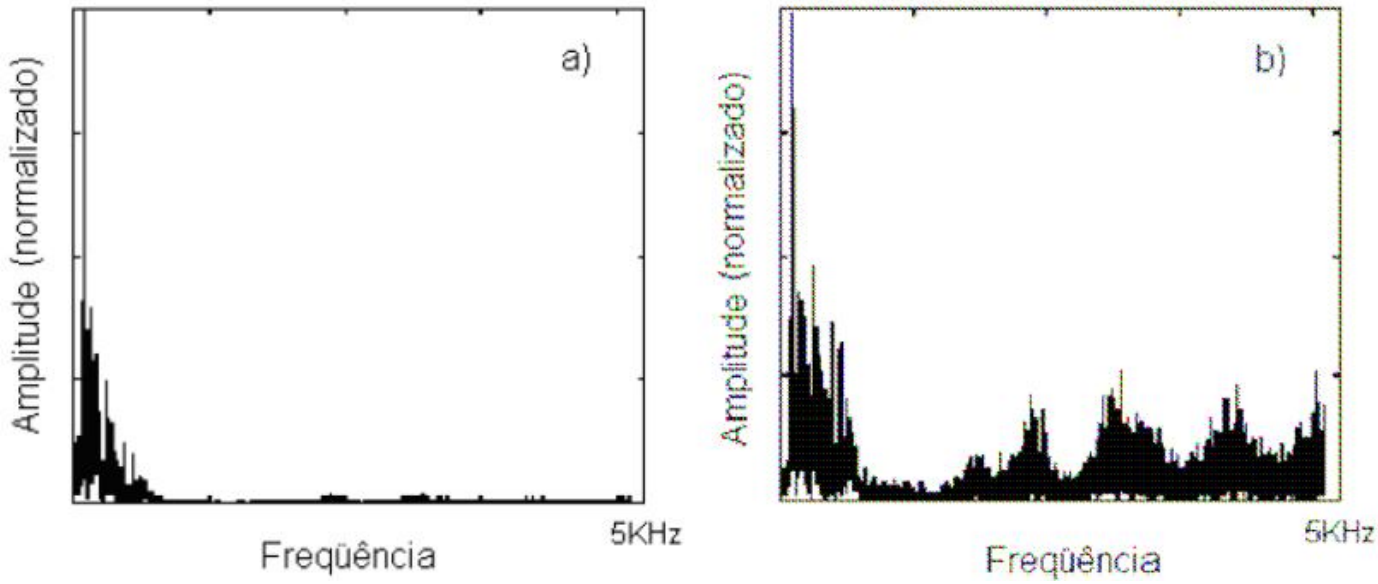

Figura 6 - Espectro de freqüências para um sinal de fala a) sem pré-ênfase e b) com pré-ênfase.

3.3.2. Divisão do sinal em frames e janelamento

As técnicas de extração de parâmetros conseguem bons resultados para sinais estacionários, ou seja, sinais cujas características são invariantes no tempo, o que não ocorre para o caso da voz. Por isso o sinal de entrada é dividido em frames (quadros) de N amostras, sendo os frames adjacentes separados por M amostras. Esta divisão varia de 10 a 25 ms, pois dentro deste intervalo o sinal é considerado quase estacionário [15].

Um frame de voz $x_t'(n)$ pode ser definido como sendo o resultado do produto de uma janela discreta $w(n)$ de tamanho N e terminando no tempo "l", com relação ao sinal de voz discreto (pré-enfatizado) $x'(n)$, como mostra a equação abaixo:

$$x_t'(n) = x'(n)w(l-n)$$

A divisão em frames do sinal é feita a partir do janelamento do mesmo. Realizar o janelamento no domínio do tempo é basicamente multiplicar o sinal pela função da janela utilizada. Uma vez que a multiplicação no domínio do tempo equivale a convolução no domínio da frequência, a aplicação da janela no domínio do tempo altera a forma do sinal também no domínio da frequência [1].

A divisão em frames é realizada através de janelas de Hamming, que são ilustradas na figura 7. São utilizadas janelas com períodos um pouco maior que o dos frames, gerando uma região de sobreposição, a fim de garantir que a variação dos parâmetros entre janelas adjacentes seja mais gradual e que a análise das informações localizadas nos extremos das janelas não seja prejudicada. A janela de Hamming é definida pela seguinte função:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), \qquad n = 0, 1, \dots, N-1$$
$$w(n) = 0, \qquad caso\ contrário$$

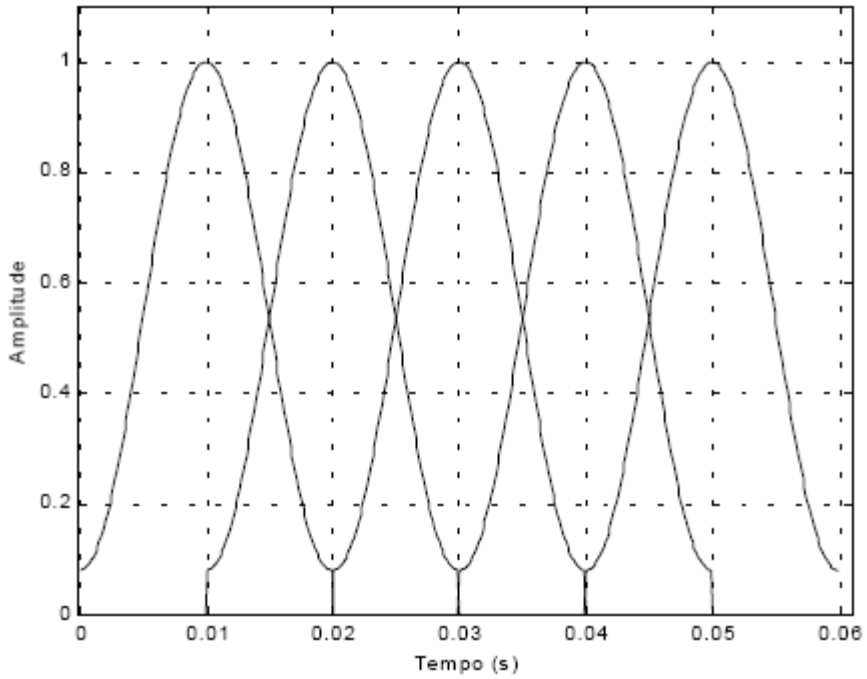

Figura 7 - Janelas de Hamming de 20 ms com superposição de 50 %.

3.3.3. Parâmetros MFCC

Estudos científicos comprovaram que a percepção humana para as frequências sonoras não segue uma escala linear. A técnica de extração de parâmetros MFCC baseia-se no uso do espectro da voz alterado segundo a escala Mel – uma escala perceptual amplamente utilizada em reconhecimento de fala - que procura se aproximar das características de sensitividade do ouvido humano [16].

Na escala Mel para cada tom com uma determinada frequência, medida em Hz, associa-se um valor medido em mel, que é a unidade de frequência dessa escala. Seja f uma frequência dada em Hz. O valor associado a essa frequência na escala Mel é denotado por $mel(f)$ e definido pela equação abaixo:

$$mel(f) = 1127 \ln\left(1 + \frac{f}{700}\right)$$

Nesta etapa realiza-se a extração dos coeficientes MFCC, que no caso deste trabalho utilizou-se um total de 39, sendo 12 parâmetros *mel-cepstrais*, 12 derivadas primeira (*delta-mel-*

cepstrais) e 12 derivadas segunda (*delta-delta-mel-cepstrais*) dos parâmetros *mel-cepstrais*, 1 parâmetro de energia, 1 derivada primeira (delta-energia) e 1 derivada segunda (delta-delta-energia) do parâmetro de energia. É necessário o uso das derivadas dos parâmetros, pelo motivo de que estes sozinhos não trazem informação sobre a evolução dinâmica da fala [6].

Para obter os coeficientes MFCC, calcula-se o quadrado do módulo da FFT das amostras pertencentes à janela em análise. Em seguida filtra-se esta janela por um banco de filtros triangulares na escala Mel. Geralmente utiliza-se 20 filtros no formato triangular passa-faixa, sendo 10 filtros uniformemente espaçados no eixo da frequência até 1 kHz e acima de 1 kHz as faixas são distribuídas segundo uma escala logarítmica, como mostrado na figura 8. Este tipo de distribuição simula o processo de audição humana.

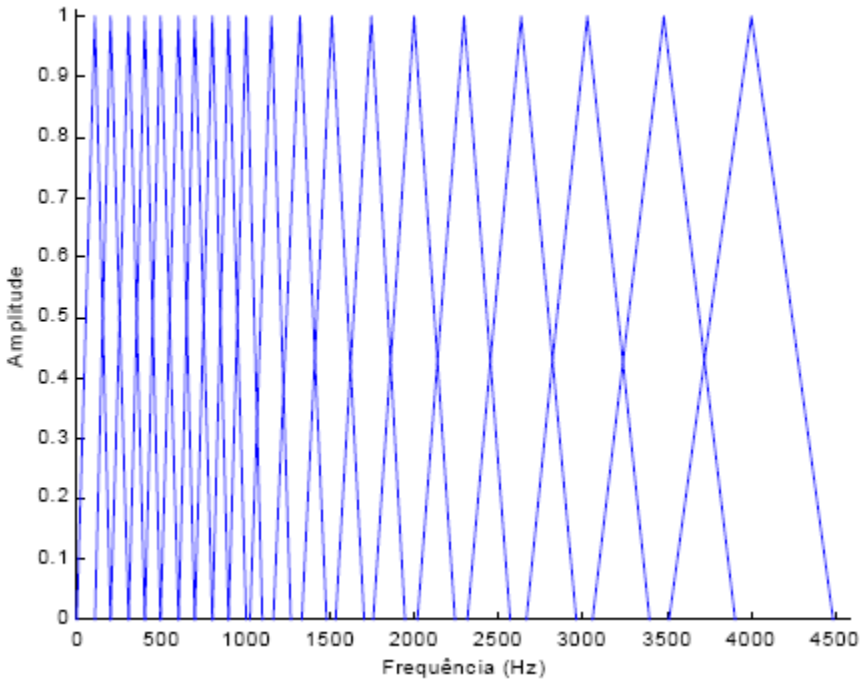

Figura 8 - Banco de 20 filtros na escala Mel

Após o banco de filtros, calcula-se o logaritmo da magnitude na saída dos filtros, a fim de obter os coeficientes cepstros. Posteriormente, calcula-se a transformada discreta inversa de cosseno (*Inverse Discrete Cosine Transform* ou IDCT) sobre estes valores, obtendo assim os coeficientes MFCC.

Os coeficientes MFCC são calculados a partir da seguinte equação:

$$c(i) = \sum_{k=1}^{M} (\log X(k)) \cos\left(\frac{i(k-0.5)\pi}{M}\right)$$

onde:

- i representa o índice do coeficiente MFCC;
- k o índice do filtro;
- M o número total de filtros e
- X(k) a energia de saída do filtro k.

Em reconhecimento de voz normalmente são descartados alguns dos últimos coeficientes MFCC, pois isto provoca uma suavização do sinal. Em geral são mantidos menos de 15 coeficientes. O coeficiente c(0) é função da soma das energias de todos os filtros e não costuma ser utilizado [16]. Com isto neste trabalho optou-se por utilizar do coeficiente de índice 1 ao coeficiente de índice 12.

A primeira e a segunda derivada dos coeficientes *mel-cepstrais* são obtidos pelas seguintes equações:

$$Dc_t^1(i) = \sum_{k=-K}^{K} \frac{kc_{t-k}(i)}{2K+1}$$

$$Dc_t^2(i) = \sum_{k=-K}^{K} \frac{kDc_{t-k}^1(i)}{2K+1}$$

onde:

- t representa o índice do frame em análise;
- i o índice do coeficiente MFCC e
- K o número de frames.

A energia é calculada utilizando-se a equação abaixo:

$$E = \log\left(\sum_{n=0}^{N-1} x_t^2(n)\right)$$

onde:

- N representa o número de amostras do frame em análise, cujo o índice é t e
- $x_t(n)$ representa o sinal de voz janelado.

A primeira e a segunda derivada da energia são calculadas a partir das mesmas equações de cálculo da primeira e segunda derivada dos coeficientes MFCC, substituindo apenas o vetor de coeficientes $c(i)$ pela energia do frame.

Os coeficientes MFCC e a energia e suas respectivas derivadas de primeira e segunda ordem são obtidos a partir de cada janela, de modo que é gerado um vetor de saída composto de 39 parâmetros para cada frame do sinal.

4. Modelos Ocultos de Markov

Os processos de Markov têm aplicações em diversas áreas e se caracterizam pelo fato de ser sem memória, isto é, toda a história passada está completamente resumida no valor atual do processo. A teoria básica dos modelos ocultos de Markov foi publicada por Baum juntamente com outros pesquisadores no final dos anos 60 e implementada pela primeira vez para aplicações em processamento de voz por Baker e por Jelinek com seus colegas nos anos 70 [17]. Porém, apenas recentemente os modelos de Markov tornaram-se a principal ferramenta utilizada em sistemas de reconhecimento automático de fala.

Um modelo de Markov, também chamado de cadeia de Markov, consiste em um conjunto finito de estados ligados entre si por transições, formando uma máquina de estados. Estas transições estão ligadas a um processo estocástico. Há ainda um outro processo estocástico associado a um modelo de Markov, que envolve as observações de saída de cada estado. Se somente as observações de saída forem visíveis a um observador externo ao processo, diz-se então que os estados estão ocultos, ou seja, o processo estocástico que envolve as transições de estados não é observável. Daí o nome Modelos Ocultos de Markov (*Hidden Markov Models* ou HMM) [1]. A figura 9 a seguir mostra um exemplo de um modelo de Markov com 3 símbolos.

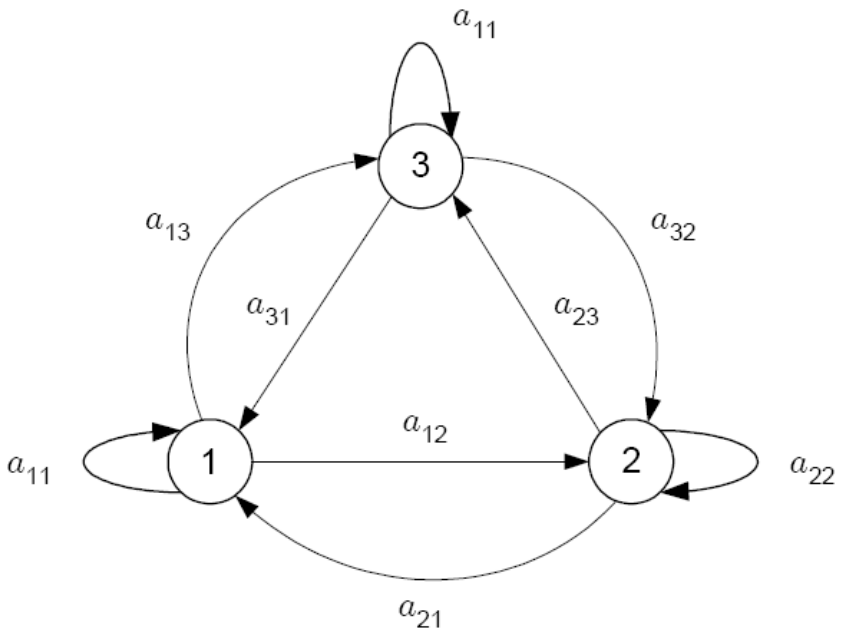

Figura 9 - Cadeia de Markov com 3 símbolos

Diversos fenômenos podem ser modelados por meio de uma máquina de estados finitos. Caso os fenômenos possuam características de processos estocásticos, pode-se então utilizar HMM para realizar a modelagem.

As observações de saída são manifestações do fenômeno sendo modelado e são descritas por funções de densidade de probabilidade (fdp), que podem ser obtidas de duas maneiras. A primeira delas que geralmente é utilizada no modelamento acústico da fala, está associada à emissão de um símbolo no instante de chegada a um estado. Já a segunda abordagem que costuma ser utilizada no processamento de linguagem, está associada à emissão de um símbolo durante a transição de um estado para outro. Os modelos associados à primeira abordagem são denominados de máquinas de Moore, enquanto que os associados à segunda abordagem são denominados de máquinas de Mealy [13].

Dependendo da fdp várias classes de HMM's podem ser definidas:

- Discreto: as observações são discretas por natureza ou discretizadas através de uma técnica de quantização vetorial, gerando assim *codebooks*.
- Contínuo: as observações são contínuas, com sua fdp contínua usualmente modelada como uma mistura finita de M Gaussianas multidimensionais.
- Semi-contínuo (híbrido): o modelo é um caso intermediário entre o contínuo e o discreto.

4.1. Elementos de um HMM

Um HMM é caracterizado pelo seguinte [17]:

- N, o número de estados do modelo. Os estados individuais são rotulados como $S = \{S_1, S_2, \dots, S_N\}$, e o estado em *t* como q_t.
- M, o número de símbolos de observação distintos por estado. Os símbolos de observação correspondem à saída do sistema sendo modelado. Os símbolos individuais são denotados como $V = \{v_1, v_2, \dots, v_M\}$.

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

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

Para o caso especial onde qualquer estado pode alcançar qualquer outro estado em um simples passo, tem-se $a_{ij} > 0$ para todo *i, j*. Para outros tipos de HMM, pode-se ter $a_{ij} = 0$ para um ou mais pares (*i, j*).

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

$$b_j(k) = P(O_t = v_k | q_t = S_j), \qquad 1 \leq j \leq N$$

$$1 \leq k \leq M$$

- A distribuição do estado inicial $\pi = \{\pi_i\}$, onde

$$\pi_i = P(q_1 = S_i), \qquad 1 \leq i \leq N$$

Percebe-se então que para uma definição completa de um HM M requer a especificação de dois parâmetros do modelo, *N* e *M*, a sequência de observações ($O = O_1, O_2, \ldots, O_T$, onde T é o número de observações na sequência) e a especificação de três conjuntos de medidas de probabilidade *A*, *B* e π. Seguindo o padrão da literatura [1, 17] será utilizada a notação compacta

$$\lambda = (A, B, \pi)$$

para indicar o conjunto de parâmetros completo do modelo.

4.2. Principais topologias de HMM

Em geral são definidas duas topologias de HMM [17]:

- Modelo ergótico, completamente conectado
- Modelo esquerda-direita

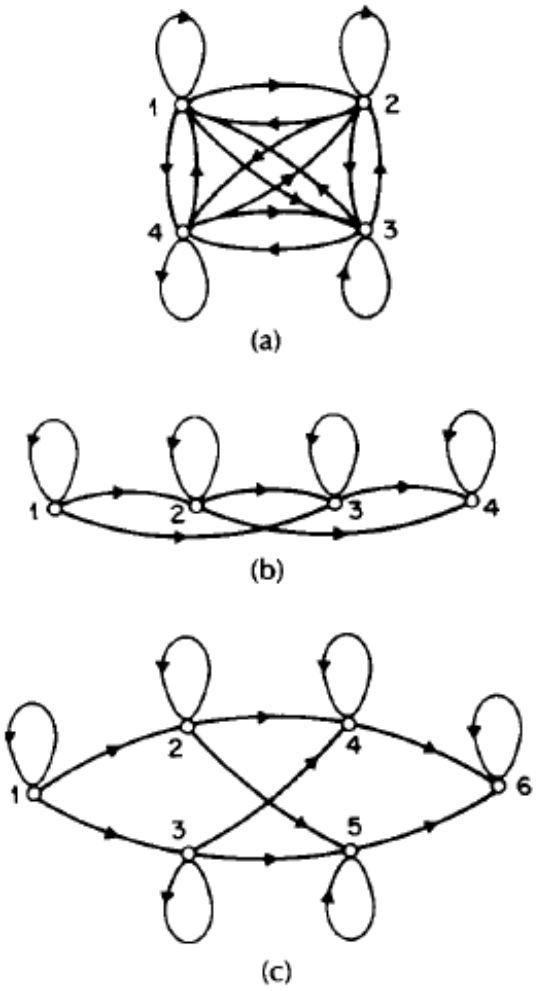

Figura 10 - Ilustração de 3 topologias de HMM distintas. a) Modelo ergótico. b) Modelo esquerda-direita. c) Modelo esquerda-direita paralelo.

No modelo ergótico, figura 10 (a), todos os estados são alcançáveis a partir de qualquer estado S_i em um número finito de passos.

O modelo esquerda-direita, figura 10 (b) possui este nome devido a propriedade de que a medida que o tempo aumenta o índice do estado aumenta ou permanece o mesmo, procedendo da esquerda para a direita. Existem algumas variações desse modelo, como pode ser visto na figura 10 (c) um modelo paralelo. A propriedade fundamental de todos os HMM's esquerda-direita é que os coeficientes de transição de estado tem a seguinte propriedade:

$$a_{ij} = 0, \qquad j < i$$

isto é, nenhuma transição para estados cujo o índice é menor do que o índice do estado atual é permitida. Além disto, a distribuição de probabilidades do estado inicial possui a seguinte propriedade:

$$\pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases}$$

ou seja a sequência de estados deve começar no estado S_1.

Todos os modelos ilustrados na figura 10 podem ser generalizados de modo a possuir um número arbitrário de estados. Se o parâmetro N for muito grande, uma determinação ótima das matrizes A e B vem a ser muito difícil. Não há meios teóricos para determinar o número de estados necessários no modelo devido ao motivo de que nem sempre os estados estão fisicamente ligados aos fenômenos observáveis.

Em geral, para o reconhecimento de fala, utiliza-se um modelo esquerda-direita simplificado conhecido como modelo de Bakis [13], o qual foi utilizado neste projeto. Neste modelo, exemplificado na figura 10 (b), são permitidos apenas transições para o mesmo estado, ou transições de um estado *i* para um estado *j*, mais à direita, onde

$$a_{ij} = 0, \qquad j > i + 2$$

4.3. Os três problemas básicos do HMM

Existem três problemas básicos encontrados no desenvolvimento de sistemas modelados por HMM's, descritos a seguir [17]:

- **Problema da avaliação**: dado um modelo $\lambda = (A, B, \pi)$ e uma seqüência de observações $O = O_1, O_2, \dots, O_T$, como calcular eficientemente $P(O|\lambda)$, a probabilidade da seqüência de observações, dado o modelo?
- **Problema da decodificação**: dado um modelo $\lambda = (A, B, \pi)$ e uma seqüência de observações $O = O_1, O_2, \dots, O_T$, qual a melhor seqüência dentro do modelo capaz de gerar essas observações?

- **Problema do treinamento**: dado um modelo $\lambda = (A, B, \pi)$ e uma seqüência de observações $O = O_1, O_2, \dots, O_T$, como ajustar os parâmetros do modelo $\lambda = (A, B, \pi)$ de modo a maximizar o valor $P(O|\lambda)$?

O problema de avaliação aparece quando se deseja selecionar dentre vários modelos aquele que mais provavelmente gerou uma dada sequência de observações A solução desta questão pode ser usada para o reconhecimento de palavras isoladas onde cada palavra é representada por um modelo. A determinação da palavra falada é realizada comparando-se as probabilidades de cada modelo ter gerado a dada seqüencia de observações.

O problema de decodificação tenta, a partir de uma sequência de observações, descobrir a parte escondida do modela, ou seja, determinar qual foi a sequência de estados que, mais provavelmente a gerou. Esta questão é encontrada no reconhecimento de fala conectada. Neste caso, existe um modelo para cada palavra, porém todas as palavras são colocadas em conjunto formando um modelo global. Desta forma, a estimação da sequência ótima de estados para uma dada sequência de observações (sequência de palavras faladas) é suficiente para determinar a sequência de palavras faladas. A determinação da sequência ótima de estados também pode ser aplicada na resolução de problemas de segmentação e rotulação de sinais de fala.

Com a solução das questões de avaliação e decodificação sabe-se como obter resultados a partir de sequências de observações e modelos com parâmetros determinados. Porém, precisa-se saber como criar um modelo de Markov para representar um dado fenômeno físico, ou seja, necessita-se de um algoritmo para inferir os parâmetros de um modelo a partir de observações de um dado fenômeno. A resposta vem com a solução do problema de treinamento, que é o mais difícil. A solução deste problema é crucial para a maioria das aplicações de HMM, pois permite o ajuste dos parâmetros de um modelo de forma a conseguir uma representação ótima para a sequência de observações. Com esta solução pode-se, por exemplo, a partir de locuções de uma determinada palavra, criar-se um modelo, o qual poderá ser utilizado para o reconhecimento de outras locuções da mesma palavra.

4.4. Algoritmos para solução dos problemas básicos

As aplicações do mundo real necessitam de uma solução para cada um dos três problemas básicos de HMM. Dentre os algoritmos que podem resolver estes problemas estão: o algoritmo

Forward ou o algoritmo Backward para o problema da avaliação, o algoritmo de Viterbi, que é próximo ao ótimo [18] para o problema da decodificação e o algoritmo Baum-Welch para o problema de do treinamento. Estes algoritmos estão definidos abaixo com base em [1].

4.4.1. Algoritmo Forward

A probabilidade da sequência de observações parcial $O_1, O_2, \dots, O_t$ (até o instante *t*) e estado S_i no instante *t*, dado o modelo λ é definida pela variável $\alpha_t(i)$, onde

$$\alpha_t(i) = P(O_1, O_2, \dots, O_t, q_t = S_i | \lambda)$$

A partir desta definição resolve-se $\alpha_t(i)$ recursivamente, como a seguir:

1) Inicialização:

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \leq i \leq N$$

2) Indução:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(O_{t+1}), \qquad 1 \leq t \leq T-1$$

$$1 \leq j \leq N$$

3) Finalização:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

A probabilidade da seqüência de observações, dado o modelo $P(O|\lambda)$ é calculada somando-se todas as variáveis $\alpha_t(i)$, para $t = T$ em todos os estados.

4.4.2. Algoritmo Backward

Uma outra opção para resolver o problema da avaliação é através do algoritmo Backward. De maneira similar, a variável $\beta_t(i)$ é definida como a probabilidade da seqüência de observações parciais de *t + 1* até o final, dado o estado S_i no instante *t* e o modelo λ, onde

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \dots, O_T | q_t = S_i, \lambda)$$

A partir desta definição resolve-se $\beta_t(i)$ recursivamente, como a seguir:

1) Inicialização:

$$\beta_T(i) = 1, 1 \leq i \leq N$$

2) Indução:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}\, b_j(O_{t+1})\beta_{t+1}(j), \qquad t = T-1, T-2, \dots, 1$$

$$1 \leq i \leq N$$

3) Finalização:

$$P(O|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1)\beta_1(i)$$

O algoritmo Backward possui recursão no sentido inverso ao do algoritmo Forward. Esses dois algoritmos são utilizados na solução do problema do treinamento. Para a solução do problema do problema de avaliação porém, apenas um deles é necessário.

4.4.3. Algoritmo de Viterbi

A fim de encontrar a melhor seqüência de estados, $Q = \{q_1, q_2, \dots, q_T\}$, para uma dada seqüência de observações $O = \{O_1, O_2, \dots, O_T\}$, define-se a quantidade

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

ou seja, $\delta_t(i)$ é a maior probabilidade ao longo de caminho no instante *t*, que considera as *t* primeiras observações e finaliza no estado S_i. Por indução tem-se:

$$\delta_{t+1}(j) = [\max_i \delta_t(i)a_{ij}]\, b_j(O_{t+1})$$

Para recuperar a sequência de estados, é necessário manter os argumentos que maximizam a expressão anterior, para cada *t* e *j*. Isto é realizado através do *array* $\psi_t(j)$. O procedimento completo para encontrar a melhor seqüência de estados é o seguinte:

1) Inicialização:

$$\delta_t(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N$$

$$\psi_1(i) = 0, \qquad 1 \le i \le N$$

2) Recursão:

$$\delta_t(j) = \max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}]b_j(O_t), \qquad 2 \le t \le T, \qquad 1 \le j \le N$$

$$\psi_t(j) = \operatorname{argmax}_{1\le i\le N}[\delta_{t-1}(i)a_{ij}], \qquad 2 \le t \le T, \qquad 1 \le j \le N$$

3) Finalização:

$$P^* = \max_{1\le i\le N}[\delta_T(i)]$$

$$q_T^* = \operatorname{argmax}_{1\le i\le N}[\delta_T(i)]$$

4) *Backtracking:*

$$q_T^* = \psi_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \ldots, 1$$

4.4.4. Algoritmo de Baum-Welch

Não existe uma maneira conhecida de resolver analiticamente o conjunto de parâmetros para um dado modelo de forma que seja maximizada a probabilidade da seqüência de observações. Na verdade, dada uma sequência finita de observações para se realizar o treinamento, não existe uma maneira ótima de estimar os parâmetros do modelo. Entretanto, pode-se escolher $\lambda = (A, B, \pi)$ tal que $P(O|\lambda)$ é localmente maximizada usando um procedimento iterativo tal como o método de Baum-Welch (também conhecido como método EM (*expectation-maximization*)), ou usando técnicas de gradiente. O algoritmo Baum-Welch, que é apresentado em termos das variáveis α_t e β_t dos algoritmos *forward* e *backward* respectivamente, é o mais indicado para a estimação dos parâmetros do HMM [17]. A re-estimação dos parâmetros a_{ij} e b_{ij} para uma dada sequência de observações através do método de Baum-Welch é descrita a seguir, baseado em [19].

Para uma única sequência de observações $O = \{O_1, O_2, \dots, O_T\}$, a re-estimação da probabilidade de transição do estado i para o estado j da matriz de transição de estados A é dada por:

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

Para os HMM's discretos, a quantidade de símbolos de saída é finita. Também para uma única elocução, a re-estimação da função de probabilidade para que um estado q_i emita um símbolo $O_t = v_k$ é obtida por

$$\bar{b}_i(k) = \frac{\sum_{t=1 \ t.q. \ O_t = v_k}^{T-1} \alpha_t(i) \quad \beta_t(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \beta_t(j)}$$

que possui as seguintes propriedades

$$\bar{b}_i(k) \geq 0, \qquad 1 \leq i \leq N, \qquad 1 \leq k \leq M$$

$$\sum_{k=1}^{M} \bar{b}_i(k) = 1, \qquad 1 \leq i \leq N$$

4.5. HMM para observações contínuas

Todas as definições de HMM realizadas neste trabalho até este ponto consideraram apenas o caso em que as observações foram caracterizadas como símbolos discretos escolhidos a partir de um alfabeto finito e por isso utilizavam densidades de probabilidade discretas em cada estado do modelo. Porém existem casos em que as observações são sinais contínuos (e.g. voz). Apesar de ser possível quantizar um sinal contínuo via *codebooks*, há grandes perdas associadas com essa quantização. Logo, a utilização de HMM's com densidades contínuas se mostra melhor para estes casos.

No intuito de se utilizar uma observação contínua, algumas restrições devem ser impostas à fdp do modelo de modo a garantir que os seus parâmetros possam ser re-estimados de forma consistente. A representação mais comum da fdp contínua, para a qual o procedimento de re-estimação foi formulado, apresenta-se na forma de misturas finitas, definida como:

$$b_j(O_t) = \sum_{m=1}^{M} c_{jm} \mathfrak{N}(O_t, \mu_{jm}, U_{jm}), \qquad 1 \leq j \leq N$$

onde:

- O_t é o vetor de observações no instante *t*;
- c_{jm} é o coeficiente de peso da *m*-ésima mistura do estado *j*;
- • *M* é o número de misturas ou regiões de um estado;
- • $\mathfrak{N}$ é qualquer densidade elipticamente simétrica ou log-côncava;
- • μ_{jm} é o vetor de média da *m*-ésima mistura do estado *j* e
- • U_{jm} é a matriz de covariância da *m*-ésima mistura do estado *j*.

Neste trabalho $\mathfrak{N}$ foi definido como uma função densidade de probabilidade Gaussiana multidimensional (*G*). Os coeficientes c_{jm} satisfazem às seguintes restrições:

$$\sum_{m=1}^{M} c_{jm} = 1, \qquad 1 \leq j \leq N$$

$$c_{jm} \geq 0, \qquad 1 \leq j \leq N, \qquad 1 \leq m \leq M$$

tal que a pdf é devidamente normalizada, isto é:

$$\int_{-\infty}^{\infty} b_j(x)\, dx = 1, \qquad 1 \leq j \leq N$$

No caso de HMM's contínuos se faz necessário a re-estimação também dos coeficientes de mistura c_{jm}, do vetor de média μ_{jm} e da matriz de covariância U_{jm} [17]. As fórmulas de re-estimação se encontram a seguir:

$$\bar{c}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m)}{\sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t(j,m)}$$

$$\bar{\mu}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) O_t}{\sum_{t=1}^{T} \gamma_t(j,m)}$$

$$\bar{U}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m)(O_t - \mu_{jm})(O_t - \mu_{jm})'}{\sum_{t=1}^{T} \gamma_t(j,m)}$$

para $1 \leq j \leq N$ e $1 \leq m \leq$ M. A variável denotada por γ é a probabilidade de estar no estado j no instante *t* com o *m*-ésimo componente de mistura ligado a observação O_t, isto é,

$$\gamma_t(j,m) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jm}\mathfrak{N}(O_t, \mu_{jm}, U_{jm})}{\sum_{k=1}^{M} c_{jk}\mathfrak{N}(O_t, \mu_{jk}, U_{jk})}\right]$$

4.6. HMM aplicado ao reconhecimento da voz

Os modelos ocultos de Markov são uma excelente representação da fala. As distribuições de probabilidade de saída modelam os eventos de fala, como o início de um fonema, por exemplo, enquanto que as probabilidades de transição modelam a duração destes eventos. Dessa maneira um HMM é capaz de absorver variações temporais entre diferentes amostras de uma mesma palavra. Esta é uma característica bastante desejável quando se deseja modelar o sinal de voz, visto que elocuções de uma mesma palavra possuem diferentes durações dependendo, do contexto no qual a

palavra está inserida, de características particulares do locutor, como o seu estado emocional, por exemplo, além de outros fatores.

O sinal de voz é contínuo no tempo e apesar de ser possível discretizá-lo com um algoritmo de quantização vetorial, uma melhor performance em reconhecimento de voz é obtida trabalhando diretamente com o sinal contínuo, pois evita-se o erro de quantização gerado durante a quantização vetorial dos vetores de parâmetros, mas, em compensação a estimação do modelo é bem mais complexa e possui um maior custo computacional. Utilizando-se a matriz de covariância U_{jm} diagonal reduz bastante o custo, porém a performance diminui, mas ainda assim possui melhores resultados que um HMM discreto. Segundo [1], é preferível utilizar a matriz de covariância diagonal com muitas misturas a utilizar poucas misturas com a matriz de covariância completa. Com base nestas informações, neste trabalho optou-se por utilizar o HMM contínuo com matriz de covariância diagonal.

Em aplicações práticas dos modelos ocultos de Markov se faz necessário a utilização de várias sequências de observações independentes para treinar um único modelo. No reconhecimento automático de fala, por exemplo, para se obter um bom modelo HMM para uma dada palavra deve-se treiná-lo com várias elocuções (sequências de observações) desta palavra, representando diferentes formas de elocução da mesma, a fim de ser mais robusto.

Cada elocução pode ser subdividida em frames, ou quadros, de igual duração que se sobrepõem, abrangendo amostras vizinhas. E a partir destes frames é possível extrair uma série de informações –parâmetros – do sinal de voz que sejam mais representativas do que puramente uma seqüência de amplitudes. Para cada frame é extraído um vetor de parâmetros. O encadeamento de variações de parâmetros no tempo pode ser modelado por uma máquina de estados finita, como o HMM. Usualmente a duração de um frame fica em torno de 20 milissegundos com sobreposição de 10 milissegundos, de modo que um determinado evento acústico ocorre em um período de alguns frames.

A partir de uma avaliação estatística do comportamento dos parâmetros pertencentes aos quadros de um dado fenômeno acústico (etapa de treinamento) é possível estimar os parâmetros de um modelo HMM - médias, covariâncias, coeficientes de ponderação e matriz de transição para um HMM contínuo [17] – que passarão a representar aquele fenômeno, caracterizando assim um estado do modelo de Markov. Assim, cada elocução pode passar a ser representada por uma seqüência de

estados. Durante esta etapa cria-se um HMM para cada palavra, isto é, estimam-se os parâmetros do modelo (A, B, π), de modo que cada elocução distinta de uma mesma palavra é utilizada na geração de um único modelo.

Na etapa de reconhecimento compara-se uma dada elocução com cada um dos modelos previamente treinados, verificando se a sua sequência de estados se assemelha à sequência do modelo, a fim de obter a verossimilhança entre os dois. Comparando-se os valores de verossimilhança entre a elocução e cada um dos modelos existentes é possível determinar qual o modelo que melhor representa esta palavra, realizando assim o reconhecimento.

5. Sistema de reconhecimento de palavras isoladas

Nos capítulos anteriores foram explanados os conceitos básicos de um sistema de reconhecimento de voz e dos modelos ocultos de Markov. No entanto, alguns pontos relacionados à implementação dos mesmos ainda não foram abordadas. Um ponto está relacionado à fase de treinamento dos modelos ocultos de Markov, que apresenta algumas peculiaridades. Neste trabalho, foi utilizada uma metodologia de treinamento por segmentação uniforme através do algoritmo LBG baseado em [19].

Neste capítulo será abordado o problema do reconhecimento automático de palavras isoladas utilizando HMM para o modelamento de sequências de frames. Assim, cada elocução é dividida em quadros de igual duração de tempo, extraindo-se de cada um deles os seus parâmetros, a fim de criar os modelos HMM's para cada palavra distinta da base de dados. Será abordado ainda os detalhes relativos à implementação das técnicas utilizadas. Todo o trabalho foi implementado no MATLAB.

5.1. Pré-processamento e extração de parâmetros

As fases iniciais que englobam o pré-processamento, a análise espectral e a extração de parâmetros do sinal de voz foram implementadas de acordo com o que foi explicado no Capítulo 3. Cada amostra da base de dados é representada por um conjunto de vetores, de tamanho $\rho = 39$ cada um, que contêm os parâmetros extraídos do sinal acústico da fala. Cada vetor é composto pelos parâmetros *mel-cepstrais*, energia e suas derivadas de primeira e segunda ordem, como foi explicado anteriormente no Capítulo 3. A quantidade de frames ou vetores varia para cada elocução de uma mesma palavra, dependendo da sua duração. O formato desse vetor pode ser visto na figura 11 [19].

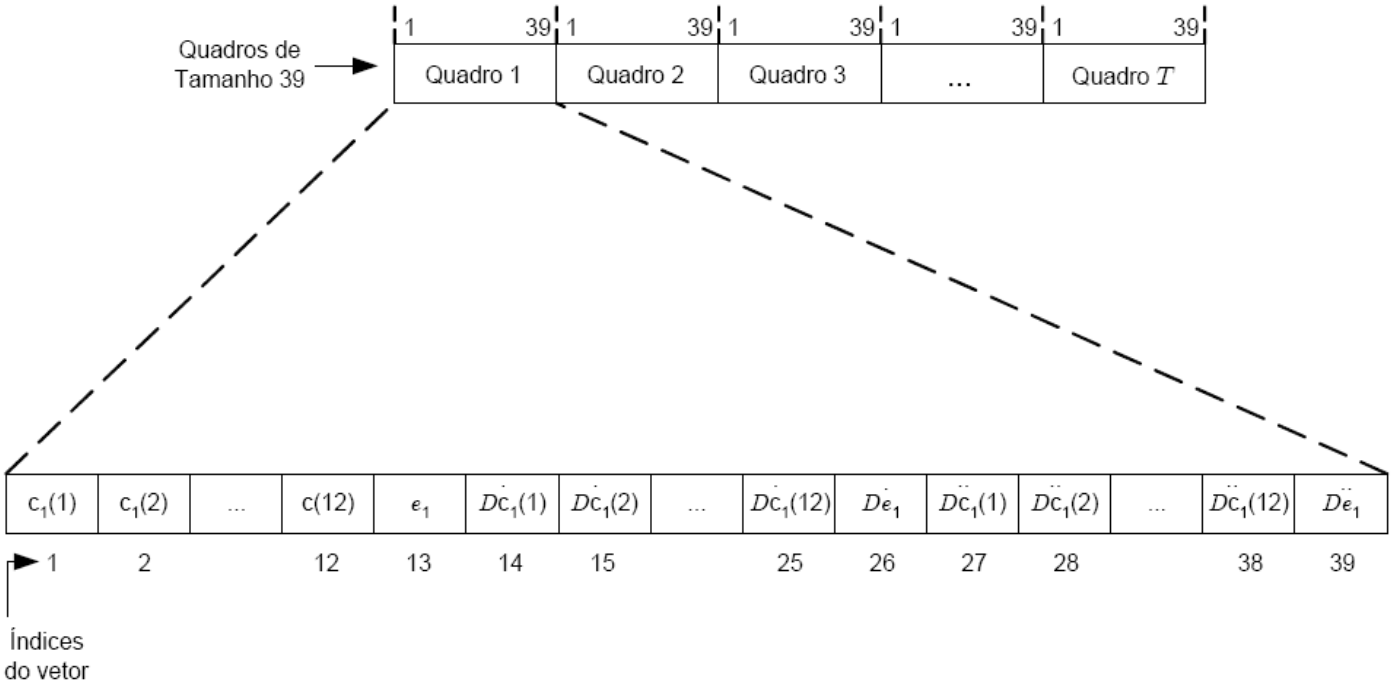

Figura 11 - Vetores de parâmetros do sinal de fala

5.2. Treinamento dos modelos ocultos de Markov

A etapa de treinamento é a peça chave do sistema de reconhecimento de voz independente de locutor e poderá decidir se o sistema apresentará bons resultados ou não. Nesta fase são estimados os modelos HMM's referentes a cada palavra do vocabulário utilizado.

Não existe uma regra padrão para a determinação do número de estados necessários para modelar-se uma dada palavra, assim como também não há para o número de misturas por estado. Estas decisões envolvem um pouco de intuição e familiaridade com os modelos de Markov. Além disso, há a necessidade de se realizar testes experimentais com diferentes valores a fim de encontrar um número ótimo. Neste projeto optou-se por utilizar, após alguns testes experimentais, 6 estados com 7 misturas gaussianas por estado.

5.2.1. Construção do *codebook*

O *codebook* ou dicionário de códigos é gerado a partir da base de dados de treinamento seguindo um critério de otimização. Um algoritmo muito eficiente para tal foi proposto por Linde, Buzo e Gray, conhecido como algoritmo LBG [20]. Com o intuito de formar o *codebook*,

particiona-se o espaço vetorial em células ou *clusters*. No caso do reconhecimento de voz para palavras isoladas, cada elocução é dividida em vetores (frames) com os parâmetros obtidos, que no caso deste trabalho possui tamanho ρ = 39.

Para cada um dos modelos HMM, todos os frames de cada elocução da palavra ele representa são distribuídos entre todos os estados do mesmo de maneira uniforme (isto é, mesma quantidade de vetores em cada estado). Após esta divisão calcula-se para cada estado um vetor centróide a partir de todos os vetores pertencentes a este estado. Maiores detalhes do algoritmo LBG pode ser visto em [19, 20].

Aplicando-se o algoritmo LBG em cada modelo gera-se um *codebook* de *M-codewords*, onde M é o número de *clusters*. Para cada célula de cada estado calcula-se um vetor média, um vetor variância e um coeficiente de peso da seguinte forma [19]:

- O vetor média μ_{jm}, com dimensão ρ, é a média amostral de todos os vetores classificados na região (*cluster*) *m* do estado *j* e corresponde ao próprio centróide gerado pelo algoritmo LBG.
- O componente *v* do vetor variância σ^2, também com dimensão ρ, da região *m* é calculado por:

$$\sigma_v^2 = \frac{1}{N_m - 1}\sum_{N_m}(p_v - \check{p}_v), \qquad 1 \leq v \leq \rho$$

onde:

- p_v é o componente *v* do vetor de parâmetros *P* que pertence a região *m*;
- N_m é o conjunto total de vetores classificados na região *m*;
- ρ é a dimensão do vetor variância.
- $\check{p}_v$ é o componente *v* do vetor média da região *m*.

Os componentes do vetor variância pertencente à célula *m* de um determinado estado *j* compõem a diagonal principal da matriz de covariância diagonal dos vetores classificados no *cluster m* do estado *j*. Desta forma a matriz de covariância, U_{jm} tem dimensão ρ x ρ.

- O coeficiente de peso da mistura c_{jm} é calculado dividindo-se o número de vetores de parâmetros classificados na célula *m* do estado *j* pelo número total de vetores de parâmetros pertencentes ao estado *j*.

5.2.2. Inicialização

Em especial para o caso de HMM's contínuos a inicialização (estimação inicial) dos parâmetros é decisiva para o alcance de um bom treinamento, maximizando $P(O|\lambda)$, influenciando significativamente no que diz respeito à convergência do treinamento [17]. Para isto a matriz A foi inicializada com distribuição uniforme, enquanto que a matriz B foi inicializada utilizando-se a segmentação uniforme das sequências de observação.

A técnica de segmentação uniforme consiste em distribuir todos os vetores de parâmetros de uma sequência de observação igualmente entre todos os estados de um determinado modelo. Caso a distribuição uniforme não seja possível devido ao número de vetores não ser múltiplo do número de estados, distribuem-se os frames restantes nos últimos estados, fazendo com que esses tenham um número maior de vetores. Como cada elocução possui um tamanho diferente e as janelas são de tamanho fixo, o número de frames gerados será diferente para cada elocução. Então para cada sequência de observações a distribuição irá adicionar um número diferente de vetores a cada estado.

A matriz B deve ser bem inicializada a fim de se obter uma rápida convergência das fórmulas de re-estimação. Os componentes de B são calculados para cada sequência de observações pela equação abaixo:

$$b_j(O_t) = \sum_{m=1}^{M} c_{jm} G(O_t, \mu_{jm}, U_{jm}), \qquad 1 \le j \le N$$

onde *G* é a função densidade de probabilidade Gaussiana multidimensional dada pela equação a seguir:

$$G(O_t, \mu_{jm}, U_{jm}) = \frac{1}{(2\pi)^{\rho/2}|U_{jm}|^{1/2}} \exp\left(-\frac{(O_t - \mu_{jm}){U_{jm}}^{-1}(O_t - \mu_{jm})'}{2}\right)$$

onde:

- ρ = 39 é a dimensão do vetor da sequência O_t;
- $|U_{jm}|$ é o determinante da matriz de covariância U_{jm} e
- ${U_{jm}}^{-1}$ é a matriz de covariância inversa.

A probabilidade de iniciar em um determinado estado π_i não é calculada, pois como já foi dito anteriormente na seção 4.2 neste trabalho usou-se o modelo de Barkis e neste tipo de modelo, o estado inicial é sempre o mais a esquerda.

5.2.3. Treinamento

Após a inicialização dos parâmetros do modelo HMM (*A, B, π)*, efetua-se o treinamento propriamente dito, pelo algoritmo de Baum-Welch. Os parâmetros de cada modelo são então re-estimados, com exceção de *π*, que não precisa ser re-estimado. Como o HMM utilizado é contínuo, é necessário re-estimar o vetor média, a matriz covariância e o coeficiente de peso, pois a matriz B depende desses parâmetros. Como os parâmetros re-estimados, gera-se então um novo modelo $\bar{\lambda} = (\bar{A}, \bar{B}, \pi)$. O processo de treinamento pode ser visualizado na figura 12, baseada em [19].

Após cada época de treinamento, verifica-se se houve convergência, calculando a distância (verossimilhança ou *Maximum Likelihood – ML*) entre o novo modelo e o anterior. No momento em que a diferença relativa entre a verossimilhança da época atual e da época anterior atingir um valor menor que 0,001 significa que a convergência foi alcançada e o treinamento é então finalizado. Porém, enquanto a convergência não ocorrer o processo é repetido substituindo-se o modelo antigo pelo atual ($\lambda = \bar{\lambda}$) e o modelo atual é mais uma vez re-estimado através do algoritmo Baum-Welch [17]. A inicialização dos parâmetros do modelo só se faz necessária na primeira interação, como pode ser visto na figura 12.

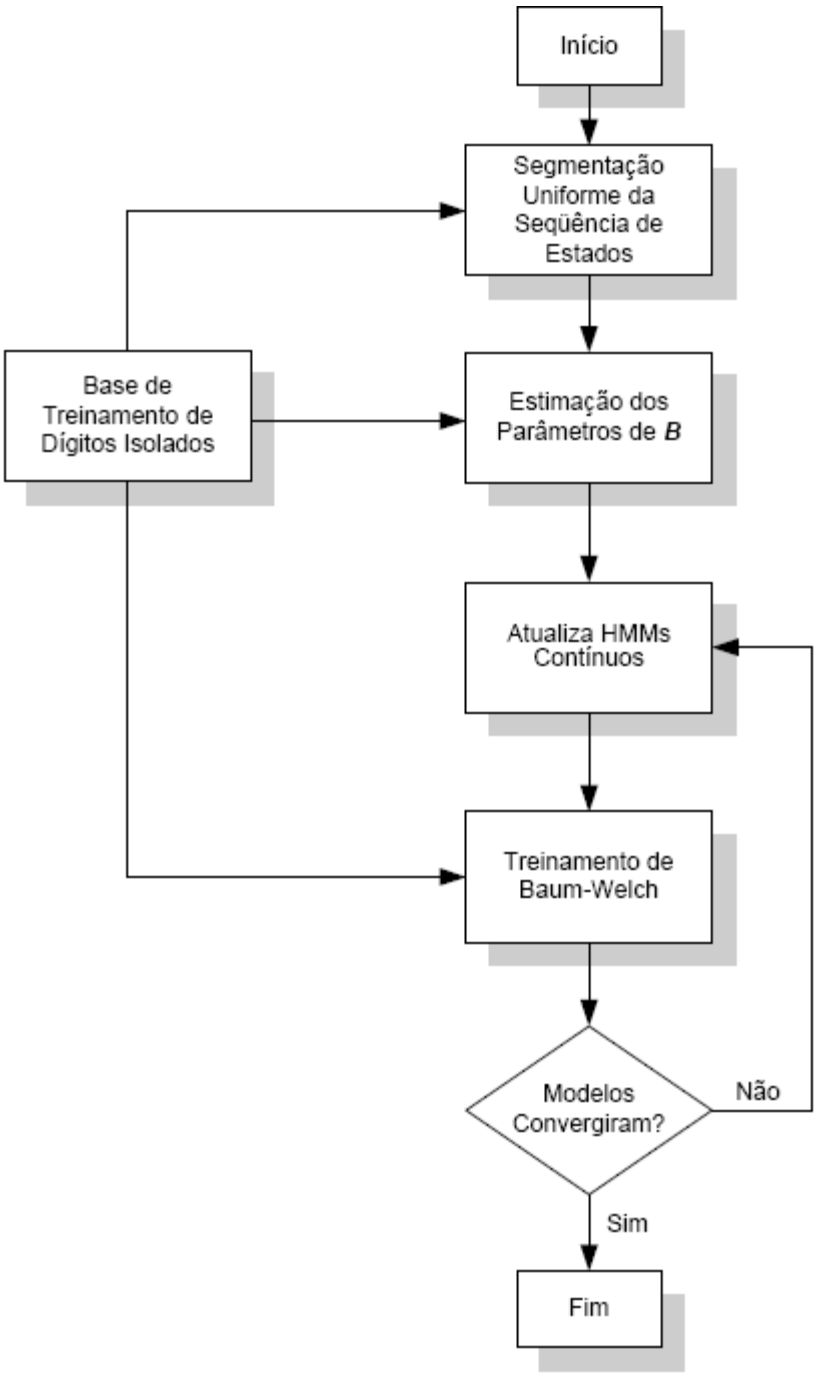

Figura 12 - Procedimento de treinamento

5.3. Reconhecimento

A fase de reconhecimento consiste em dada uma elocução, descobrir qual o modelo que tem a maior probabilidade de gerá-la. Nesta fase não é necessário cálculo de parâmetros HMM, nem da

geração de um modelo, apenas do pré-processamento e da extração de características que gera a sequência de observações a ser utilizada no algoritmo de reconhecimento. O procedimento de reconhecimento pode ser visualizado na figura 13, baseada em [17].

O reconhecimento pode ser obtido utilizando-se o algoritmo *forward* a fim de se determinar a probabilidade de cada modelo de palavra gerar uma dada elocução. O modelo com maior probabilidade é, então, escolhido como correspondendo à palavra falada, assumindo-se que, a priori, todas as palavras têm uma mesma probabilidade de ocorrência. O algoritmo de *Viterbi* também pode ser utilizado para esta classificação, embora este resulte apenas em uma aproximação para a probabilidade de uma dada sequência de observações. Os resultados utilizando-se um ou outro algoritmo são praticamente idênticos No sistema implementado, o reconhecimento das elocuções é baseado no algoritmo *forward*.

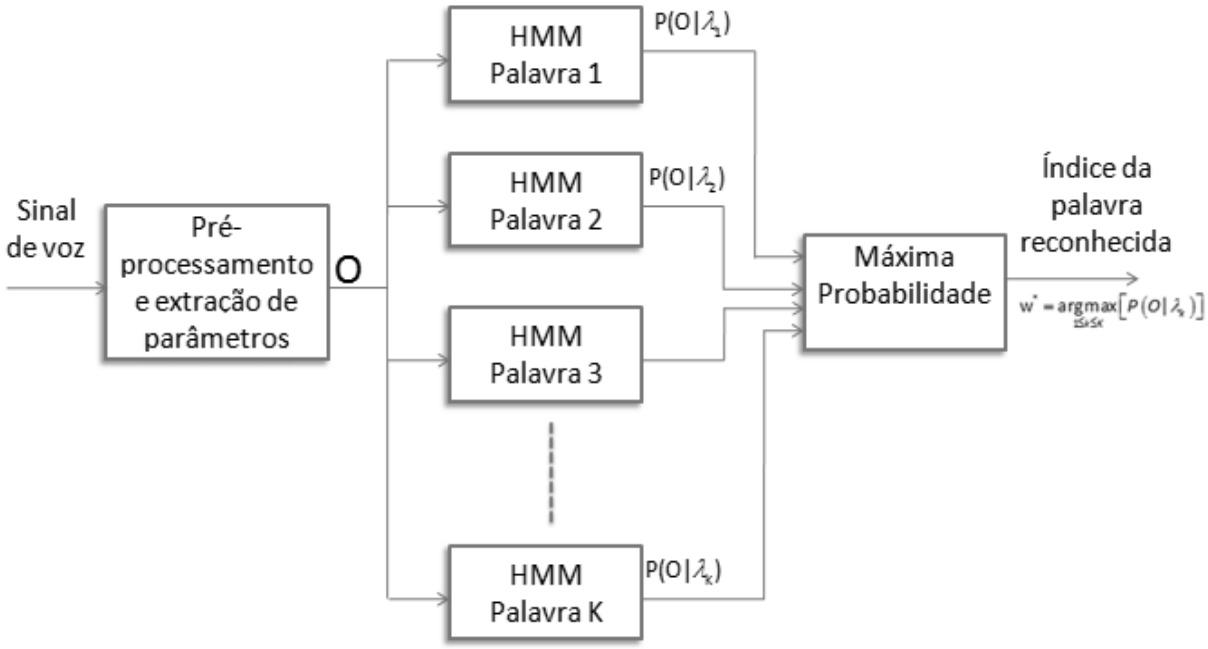

Figura 13 - Procedimento de reconhecimento

6. Experimentos e resultados

Neste capítulo apresentam-se os testes realizados e os resultados obtidos em cada uma das experimentações para um vocabulário pequeno formado pelos dígitos de 0 a 9 e pelas palavras “sim” e “não”.

Visando a obtenção de uma taxa de acertos aceitável, foram realizados vários experimentos alterando-se o número de estados e o número de misturas gaussianas por estado, a fim de obter-se o número ideal destes parâmetros. Uma vez encontrados estes valores, realizou-se experimentos para uma base de testes dependente de locutor e para uma independente, encontrando a taxa de acerto do sistema para cada um dos casos. Estes experimentos e os seus resultados poderão ser vistos no decorrer deste capítulo.

6.1. Base de dados

A maior dificuldade existente no reconhecimento de fala deve-se às diferenças existentes entre elocuções de uma mesma palavra, que são ainda maiores no caso destas elocuções serem produzidas por locutores distintos. Parâmetros como os descritos no Capítulo 3 assumem valores distintos quando extraídos de diferentes amostras de uma mesma palavra. Consequentemente, para a realização do treinamento dos modelos HMM’s, há a necessidade da apresentação de um número significante de diferentes elocuções de cada palavra, de forma que os HMM’s possam absorver as variações existentes nos parâmetros extraídos destas elocuções.

A criação da base de dados para a realização do treinamento representa, portanto, uma etapa determinante para o bom desempenho de um sistema de reconhecimento de voz. Para se conseguir um sistema independente de locutor faz-se necessário a utilização de uma base de dados contendo amostras de diferentes locutores, representando assim as mais variadas características dos possíveis usuários da aplicação desejada, tais como, sexo, idade, sotaque, timbre de voz, etc. Outro fator a ser levado em consideração é o ambiente (tipo de transdutor, canal de transmissão, quantidade de ruído, etc.) de gravação desta base. Um sistema de reconhecimento treinado com uma base de dados gravada em um determinado ambiente terá seu desempenho bastante degradado se avaliado com gravações de outro ambiente. Deve-se, portanto, escolher o ambiente de gravação da base de treinamento o mais próximo possível do ambiente no qual o sistema será utilizado. No caso de

variação do ambiente de utilização, deve-se gerar a base a partir de gravações nos diversos ambientes.

Para este trabalho, como o sistema não tinha uma aplicação definida, a base de dados foi gerada gravando-se as amostras a partir de diferentes ambientes com diferentes níveis de ruído, utilizando-se diferentes transdutores.

A base de dados é composta por elocuções de 13 locutores, dos quais 10 locutores são do sexo masculino e 3 do sexo feminino. Cada locutor gravou 10 amostras de cada palavra do vocabulário, totalizando 1560 elocuções.

6.2. Experimentos

Inicialmente a base foi dividida em dois grupos, sendo um para treinamento e o outro para testes, de forma que o grupo de treinamento foi formado por 1080 elocuções, sendo 90 de cada palavra, e o grupo de testes foi formado pelas outras 480 elocuções, sendo 40 de cada palavra. Realizou-se então os primeiros experimentos a fim de obter-se o melhor número de estados e o melhor número de misturas gaussianas por estado.

No experimento 1 fixou-se o número de misturas gaussianas por estados em 3, para que houvesse menos processamento e o tempo fosse menor e variou-se o número de estados. Já no experimento 2, o número de misturas gaussianas variou e o número de estados ficou fixo em 3, pelo mesmo motivo que fixou-se o número de misturas gaussianas em 3 no primeiro experimento. Os resultados destes dois experimentos podem ser visualizados respectivamente nas tabelas 1 e 2.

Número de estados	**Taxa de acerto(%)**
3	80
4	73,75
5	79,38
6	82,29
7	81,88
8	82,29
9	83,13
10	83,96

Tabela 1 - Experimento 1 (3 misturas gaussianas por estado)

Número de misturas gaussianas	Taxa de acerto(%)
3	80
4	76,25
5	80,63
6	85,42
7	86,46
9	82,29
12	87,5
15	82,71

Tabela 2 - Experimento 2 (3 estados)

A partir dos resultados obtidos nos experimentos 1 e 2, observou-se que a melhor opção para o número de estados é 6, enquanto que para o número de misturas gaussianas é 7, pois apesar de com 12 misturas o resultado ter sido melhor, o ganho é muito pouco se comparado ao custo computacional gerado por 5 misturas gaussianas a mais. Optou-se então por utilizar estes valores nos próximos experimentos.

A fim de verificar a eficácia do sistema desenvolvido para o caso dependente de locutor e para o caso independente de locutor, realizou-se um novo experimento para cada um dos casos. Para a realização destes experimentos foram realizadas novas divisões na base, respeitando a idéia de um grupo para treinamento e um para testes. Para o experimento dependente de locutor (experimento 3) os dois grupos foram formados pelos mesmos locutores e a divisão foi realizada de tal maneira que o grupo de treinamento ficou com 70% da base, ou seja, 1092 amostras e o grupo de testes com os outros 30% da base, ou seja, 468 amostras. Já para o experimento independente de locutor (experimento 4), os dois grupos foram divididos de forma que o grupo de testes não possuía nenhum dos locutores utilizados no grupo de treinamento, sendo o grupo de treinamento formado então por 9 locutores, contendo então 1080 elocuções, e o grupo de testes formado pelos outros 4 locutores, contendo então 480 elocuções. Os resultados podem ser visualizados respectivamente nas tabelas 3 e 4.

	Taxa de acerto(%)
0 (zero)	100
1 (um)	97,43
2 (dois)	94,43
3 (três)	82,05

4 (quatro)	100
5 (cinco)	89,74
6 (seis)	92,3
7 (sete)	94,87
8 (oito)	100
9 (nove)	100
Sim	94,87
Não	100
Média	95,72

Tabela 3 - Experimento 3 (dependente de locutor)

	Taxa de acerto(%)
0 (zero)	95
1 (um)	82,5
2 (dois)	87,5
3 (três)	67,5
4 (quatro)	100
5 (cinco)	87,5
6 (seis)	90
7 (sete)	90
8 (oito)	92,5
9 (nove)	100
Sim	77,5
Não	97,5
Média	88,96

Tabela 4 - Experimento 4 (independente de locutor)

A partir dos resultados obtidos no experimento 4 utilizou-se um limiar para a máxima verossimilhança, a fim de desconsiderar palavras não existentes no vocabulário. Isto é, as elocuções que não possuírem verossimilhança maior ou igual a este limiar com nenhum dos modelos HMM's serão consideradas palavras desconhecidas, o que gera uma maior robustez ao sistema. Este limiar foi escolhido verificando-se as verossimilhanças encontradas rodando cada palavra do grupo de teste. Foi então escolhido um valor razoável para que as palavras existentes pudessem ainda assim ser reconhecidas.

Dois últimos experimentos (experimentos 5 e 6) foram realizados a fim de validar o sistema, utilizando a técnica de validação chamada *K-fold cross-validation*. Nesta técnica o conjunto de dados é dividido igualmente em K subconjuntos exclusivos, ou seja, nenhum elemento

de um subconjunto se repete em um outro subconjunto. Na fase e treino utiliza-se K-1 subconjuntos concatenados, e na fase de validação utiliza-se o subconjunto restante. Estas fases são repetidas K vezes, permutando circularmente os subconjuntos. A precisão final é então calculada usando a média das precisões encontradas em cada permutação [21]. Esta técnica proporciona uma avaliação menos tendenciosa da precisão de um classificador à custa de um maior custo computacional. Para a validação do sistema implementado utilizou-se K = 4. No experimento 5 optou-se por não utilizar o limiar escolhido, por enquanto que no experimento 6, este limiar foi utilizado. Os resultados (precisão final para cada palavra e precisão final geral) destes experimentos podem ser visualizados nas tabelas 5 e 6 respectivamente.

	Taxa de acerto(%)
0 (zero)	100
1 (um)	98,48
2 (dois)	97,73
3 (três)	83,17
4 (quatro)	100
5 (cinco)	88,61
6 (seis)	90,13
7 (sete)	95,41
8 (oito)	99,22
9 (nove)	100
Sim	96,21
Não	99,24
Média	95,68

Tabela 5 - Experimento 5 (validação através da técnica *K-fold cross-validation, sem utilização do limiar*)

	Taxa de acerto(%)
0 (zero)	100
1 (um)	98,48
2 (dois)	97,73
3 (três)	82,39
4 (quatro)	99,24
5 (cinco)	87,86
6 (seis)	90,13
7 (sete)	95,41
8 (oito)	99,22
9 (nove)	100

Sim	96,21
Não	98,48
Média	95,43

Tabela 6 - Experimento 6 (validação através da técnica *K-fold cross-validation, com o limiar*)

Comparando-se os resultados obtidos nos experimentos 5 e 6 percebe-se que o limiar causa uma degradação mínima na taxa de acerto, sendo então aceitável.

7. Conclusão e trabalhos futuros

O trabalho realizado visou à pesquisa e ao desenvolvimento da tecnologia de reconhecimento de voz, tendo como enfoque principal a implementação de um sistema para reconhecimento de palavras isoladas. O enfoque adotado, além de resultar em um bom embasamento teórico e prático na área de reconhecimento de voz, possibilitou o desenvolvimento de uma plataforma inicial sobre a qual pesquisas e desenvolvimentos posteriores possam ser mais facilmente realizados.

Primeiramente, foi apresentada uma introdução à área de reconhecimento de voz, discutindo as suas aplicações e relatando o seu histórico e as características que um sistema desse tipo precisa ter. Depois, descreveu-se o sistema de pré-processamento e dos sinais de fala e a técnica de extração de parâmetros MFCC utilizada. Dentro deste contexto foram relatadas técnicas que diminuem os ruídos, técnicas de detecção do começo e fim da fala, de pré-ênfase e de janelamento do sinal em frames.

Discutiram-se ainda neste trabalho os modelos ocultos de Markov ou HMM, descrevendo os seus elementos, os tipos e as arquiteturas existentes, os seus problemas básicos e os algoritmos que tem por finalidade a resolução destes problemas.

Apresentou-se então o desenvolvimento em MATLAB de um sistema de reconhecimento de voz, independente de locutor para palavras isoladas utilizando HMM's contínuos. Foram levantadas então algumas informações relevantes ao desenvolvimento de um sistema desse tipo. E por último foram apresentados os experimentos e os resultados alcançados por este sistema, utilizando uma base de dados formada por 10 locutores do sexo masculino e 3 do sexo feminino, cada qual com 10 amostras de cada palavra gravada, gerando um total de 1560 amostras gravadas em diferentes ambientes.

Observando-se os resultados obtidos, algumas inferências podem ser feitas:

- Diferenças de ambientes de gravação possuem grandes influências sobre a precisão de um reconhecedor. Com uma base de dados gravada toda em um mesmo ambiente, provavelmente este sistema chegaria próximo a 100% de acerto. Conclui-

se então que se o objetivo do sistema é ser utilizado em ambiente controlado, é aconselhável que a base de dados seja gravada também neste ambiente.

- Uma outra causa de erros no reconhecimento deve-se ao sistema confundir algumas vezes as palavras “três”, “seis”, “sete”, “cinco” e “sim”, onde as palavras “três” , “seis” e “cinco” foram as que possuíram as maiores taxas de erro, sendo as palavras “cinco” e “sim” muitas vezes confundidas entre si e as palavra “três” e “seis” também muitas vezes confundidas entre si. Uma possível solução seria a implementação de um pós-processamento, que poderia, por exemplo, utilizar a taxa de cruzamentos por zero para estas palavras. Porém um pós-processamento só deve ser realizado em um sistema com um vocabulário já fixo, pois o pós-processamento complica a ampliação do vocabulário.
- O algoritmo de detecção de extremos da fala também pode ser melhorado, pois em alguns casos não funcionou muito bem. O erro muitas vezes estava ligado a um clique existente no final da palavra que ocorre devido ao fechamento dos lábios quando se encerra a pronúncia da palavra.

De modo geral os resultados foram satisfatórios, porém, o sistema pode ser melhorado de várias maneiras:

- Gerando-se uma base maior e gravada em um mesmo ambiente, ou em ambientes nos quais o sistema vá ser utilizado, caso tenha-se como objetivo uma aplicação real.
- Utilizando-se um vocabulário diferente.
- Utilizando-se técnicas derivadas do HMM, como o Type-2 Fuzzy HMM, já utilizado em [22] para reconhecimento de fonemas, o qual obteve uma taxa de acerto maior que o HMM. A superioridade foi ainda maior quando se tratando de sinais com altos níveis de ruídos.
- Pesquisando-se outras técnicas de pré-processamento e extração de parâmetros e/ou melhorando as usadas neste trabalho.

A área de reconhecimento de voz ainda é considerada nova, especialmente a nível de Brasil. Logo o desenvolvimento, a curto prazo, de sistemas práticos de reconhecimento de voz, principalmente para palavras isoladas, utilizando tecnologias básicas e novas tecnologias que estão

surgindo se mostram muito importante. Tais sistemas podem servir de base de pesquisa e desenvolvimento para outros sistemas mais elaborados que devem ser implementados a mais longo prazo, como um sistema de reconhecimento de fala contínua utilizando um grande vocabulário.

Conclui-se então que o presente trabalho representa o básico da tecnologia de reconhecimento de voz aplicada a uma situação próxima da real. Para a utilização do sistema desenvolvido em uma situação mais realista ainda é exigido novas pesquisas que poderão torná-lo mais realista e consistente.

Referências

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

[2] Rabiner, L. R., Sambur, M. R. An algorithm for determing the endpoints of isolated utterances, Bell System Technical Journal, vol. 54, pp. 297-315, 1975.

[3] Davis, K. H., Biddulph, R., Balashek, S. Automatic recognition of spoken digits. The Journal of the Acoustical Society of America, 1952.

[4] Itakura, F., Saito, S., A Statistical Method for Estimation of Speech Spectral Density and Formant Frequencies, Electronics and Communications in Japan, Vol. 53A, pp. 36-43, 1970.

[5] Rabiner, L. R., Juang B. H. Automatic speech recognition - a brief history of the technology development, Elsevier Encyclopedia of Language and Linguistics, 2005.

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

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

[8] Rabiner, L. R., Schafer, R.W. Digital processing of speech signals. Prentice Hall, 1978

[9] Proakis, J. G., Manolakis, D. G. Digital Signal Processing: principles, algorithms, and applications. New Jersey: Prentice Hall, 1996, 968 p.

[10] Chou, W., Juang, B.H. Pattern recognition in speech and language processing. CRC Press, 2003

[11] Chu, W. C. Speech coding algorithms. Wiley-Interscience, 2003.

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

[13] Deller, J. R., Proakis, J. G., Hansen, J. H. L. Discrete-time processing of speech signals. Macmillan Publishing Company, New York, 1993

[14] Bourouba, E-H., Bedda, M., Djemili, R. Isolated Words Recognition System Based on Hybrid Approach DTW/GHMM. Informatica, An International Journal of Computing and Informatics, Vol. 30, Number 3, pp. 373-384, 2006.

[15] Braga, P. L. Reconhecimento de voz dependente de locutor utilizando Redes Neurais Artificiais. Trabalho de graduação. UPE, 2006.

[16] Picone, J. Signal Modeling Techniques in Speech Recognition, Proceedings of IEEE, 81, nr. 9, 1215-47, 1993

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

[18] Lou, H.L. Implementing the Viterbi Algorithm, IEEE Signal Processing Magazine, pp. 42-52, 1995.

[19] Gonçalves, J. V. Estudo e implementação de um sistema de reconhecimento de dígitos conectados usando HMMs contínuos. Tese de Mestrado. UNICAMP, SP, 2005.

[20] Linde, Y., Buzo, A., Gray, R. M. An algorithm for vector quantizer design, IEEE Trans. Communications, vol. COM-28, issue 1, pp. 84-95, 1980

[21] Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95), pp. 1137–1143, 1995.

[22] Zeng, J., Liu, Z.-Q. Type-2 fuzzy hidden Markov models and their application to speech recognition, IEEE Trans. Fuzzy Syst. 14 (3) 454–467, 2006.

Assinaturas

Tsang Ing Ren
Orientador

Anderson Gomes da Silva
Aluno