UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de São José do Rio Preto

Rodrigo da Silva Barboza Lima

Voice Morphing com base em Aprendizado *Wavelet*

São José do Rio Preto
2023

Rodrigo da Silva Barboza Lima

Voice Morphing com base em Aprendizado *Wavelet*

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação junto ao Programa de Pós-Graduação Interunidades em Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho" - Campus de São José do Rio Preto.

Orientador: Prof. Dr. Rodrigo Capobianco Guido

São José do Rio Preto
2023

L732v

Lima, Rodrigo da Silva Barboza
Voice Morphing com base em Aprendizado Wavelet / Rodrigo da Silva Barboza Lima. -- São José do Rio Preto, 2023
59 p.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp), Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto
Orientador: Rodrigo Capobianco Guido

1. Processamento de sinais. 2. Processamento de sinais técnicas digitais. 3. Inteligência artificial. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

Rodrigo da Silva Barboza Lima

Voice Morphing com base em Aprendizado *Wavelet*

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação Interunidades em Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho" - Campus de São José do Rio Preto.

Comissão Examinadora

Prof. Dr. Rodrigo Capobianco Guido
UNESP - Campus de São José do Rio Preto
Orientador

Profª. Drª. Luciene Cavalcanti Rodrigues
FATEC - Campus de São José do Rio Preto

Prof. Dr. Lucimar Sasso Vieira
FATEC - Campus de São José do Rio Preto

São José do Rio Preto
06 de setembro 2023

AGRADECIMENTOS

Primeiramente, agradeço a Deus por todas as oportunidades e pessoas que cruzaram meu caminho.

Aos meus pais e meu irmão, pelo apoio constante e presença em todos os momentos, difíceis ou não.

Ao meu orientador e amigo Rodrigo Capobianco Guido, agradeço profundamente pelo incentivo, apoio e orientação que foram fundamentais para iniciar e concluir esta pós-graduação.

Ao meu amigo Jogi Suda, por estarmos juntos e nos apoiarmos mutuamente ao longo deste período.

A todos os amigos e professores que tive a honra de conhecer durante minha jornada na UNESP.

RESUMO

Algoritmos para conversão de voz, tradicionalmente conhecidos como *voice morphing algorithms*, têm tido aplicações diversas, tais como a substituição de falas de locutores falescidos ou com o sistema fonatório incapacitado, a possibilidade que determinada música originalmente cantada por um locutor surja com a voz de outro, e assim por diante. Recentemente, tais técnicas têm se tornado mais populares em função dos algoritmos de *deep fake*, entretanto, a maior desvantagem deles é a dificuldade em criar modelos interpretáveis, assim como ocorre com quaisquer estratégias baseadas em *deep learning*. Desse modo, neste trabalho, a intenção é a de explorar uma outra possibilidade: a conversão de voz baseada na Transformada *Wavelet* de Tempo Discreto (DTWT), trabalhando em associação com redes neurais artificiais rasas, que possuem maior possibilidade de gerar modelos interpretáveis. Particularmente, relacionam-se quais os melhores filtros *wavelet* para conversão de determinados padrões de voz. Testes são realizados com vozes da base de dados TIMIT do *Linguistic Data Consortium* (LDC) que permitem constatar a viabilidade da estratégia proposta considerando testes de preferência acústica e métricas de distância.

Palavras-chave: Processamento de sinais. *Voice morphing*. *Wavelets*. Redes neurais artificiais.

ABSTRACT

Algorithms for voice conversion, traditionally known as voice morphing algorithms, have been used in a number of applications, such as replacing deceased speakers or those with an incapacitated phonatory system, playing a song as if it had been sung by another speaker, and so on. Recently, such techniques have become more popular due to the deep fake algorithms, however, their biggest disadvantage is the difficulty in creating interpretable models, as happens with any strategies based on deep learning. Thus, in this work, the intention is to explore another possibility: speech conversion based on Discrete Time Wavelet Transform (DTWT) working in association with shallow artificial neural networks, which have greater possibility of generating interpretable models. Particularly, the best wavelet filters for converting certain speech patterns are determined. Tests are performed with voices from the Linguistic Data Consortium (LDC) TIMIT database, being assessed based on acoustic preference tests and distance metrics.

Keywords: Signal processing. Voice morphing. *Wavelets*. Artificial neural networks.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

INTRODUÇÃO

Na grande área de processamento digital de sinais de voz [1], o termo conversão de voz [2] [3], também referido como *voice morphing* (VM), *voice transformation* ou *voice conversion*, faz referência ao processo pelo qual uma locução pronunciada por um locutor, chamado de *source speaker* (SS), se transforma de forma que pareça ser pronunciada por um outro locutor, que é o *target speaker* (TS).

Diversas são as modalidades dos sistemas dedicados ao processo de VM: *one-to-one*, na qual somente a voz de um SS específico pode ser convertida de modo a se tornar perceptualmente similar à voz de um TS particular, *one-to-many*, na qual somente a voz de um SS específico pode ser convertida de modo a se tornar perceptualmente similar à voz de diversos TSs pré-matriculados no sistema conversor, *many-to-one*, na qual a voz de praticamente qualquer SS pode ser convertida de modo a se tornar perceptualmente parecida com a voz de um TS particular e, finalmente, *many-to-many*, na qual a voz de uma ampla gama de SSs pode ser convertida de modo a se tornar perceptualmente parecida com a voz de inúmeros TSs conhecidos do sistema conversor.

Registre-se, dentre as possíveis aplicações dos sistemas de VM, as seguintes:

- burlar sistemas de autenticação biométrica por voz [4], verificando as eficiências dos algoritmos autenticador e conversor. Nesse âmbito, é notável o uso dos sistemas conversores para gerar os ataques de *voice spoofing* [5];
- editar filmes, séries de televisão e rádio-novelas, nos quais determinado ator já faleceu, de modo que os possíveis *dublês* poderiam ter suas vozes alteradas;
- reproduzir fonéticamente um texto escrito valendo-se da síntese da voz de de-

terminado locutor indisponível ou incapacitado para tal, dentre outras possibilidades.

No passado recente, diversas estratégias foram experimentadas para construir sistemas de VM, inclusive valendo-se de novas famílias de *wavelets* [6]. Mais recentemente, entretanto, com o avanço dos sistemas de inteligência artificial, principalmente aqueles baseados em *deep learning* [2], outros procedimentos têm sido também considerados.

Diante do exposto, o conteúdo desta monografia está focado, no Capítulo 2, na revisão de técnicas para VM e de conceitos utilizados pelas mesmas, de modo a embasar a abordagem proposta, descrita no Capítulo 3. Prosseguindo, no Capítulo 4 são apresentados os testes e respectivos resultados e, por fim, no Capítulo 5 são apresentadas as conclusões, as quais precedem as referências.

CONCEITUAÇÃO TEÓRICA

2.1 *Voice Morphing*

Um aspecto essencial da VM é a possibilidade de converter ou transformar o som de um locutor, ou seja, o SS, para outro, isto é, o TS, evitando perdas no conteúdo original da mensagem linguística. Assim, o foco dos procedimentos de VM é a manipulação da identidade do locutor [2].

As estratégias utilizadas para viabilizar VM, conforme a Figura 2.1, podem ser divididas inicialmente entre paralelas e não-paralelas. Conversão de voz paralela considera uma situação ideal em que um par *source* e *target* de sinais são fornecidos para o treinamento do sistema, com o mesmo conteúdo linguístico. No caso não-paralelo, que está fora do escopo deste trabalho, o treinamento é realizado com sinais não pareados [7].

A metodologia típica para o desenvolvimento de um sistema de VM, usualmente envolvendo Inteligência Artificial (IA), consiste na execução de quatro etapas gerais:

- treinamento: construção do modelo de conversão por meio de treinamentos do sistema de IA utilizado;
- análise e extração de características: decomposição do sinal do SS em características que auxiliam no modelo de conversão;
- mapeamento: alteração das características obtidas do SS para a voz do TS;
- reconstrução: a reconstrução e sintetização para a voz do TS.

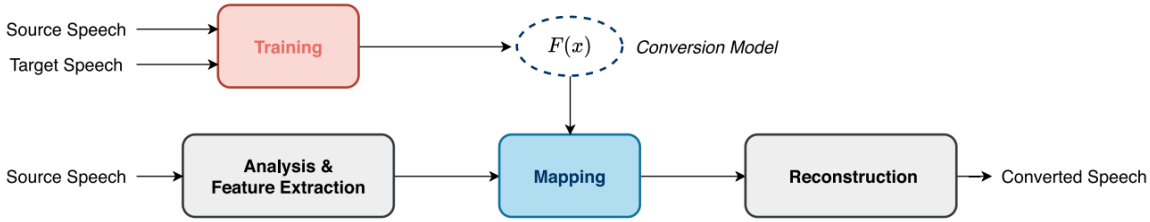


Figura 2.1: *Pipeline* típico de um sistema de VM [2].

Existem diferentes formas de criar um modelo de conversão, porém o uso da IA é a abordagem mais comum atualmente em trabalhos distintos que variam entre redes neurais e aprendizado profundo [8] [9] [10]. Essa tendência é observada devido à facilidade e eficiência na criação de modelos específicos. Por sua vez, a extração de características é importante para a identificação e transformação dos elementos que diferenciam vozes e componentes fonéticos como o *pitch*, entonação, emoção e sotaques [11] [12].

VM pode ser aplicado em conjunto com sistemas de *text-to-speak* (TTS) para uma melhor personalização na voz sintetizada com a possibilidade de aplicar métodos específicos de *one-to-many* ou mesmo o treinamento para personalização por meio de *one-to-one*. De qualquer forma, tradicionalmente VM difere da sintetização de voz, ou *voice synthesis*, na forma como opera, pois enquanto VM altera o sinal diretamente a sintetização de voz utiliza a representação fonética [7]. Porém, com o avanço do uso de redes neurais profundas, ou seja, *deep neural networks* ou DNNs, estudos têm surgido contendo relatos de melhorias na qualidade de conversões com o uso de técnicas linguísticas oriundas da sintetização [2] [7].

A conversão de texto para fala é amplamente utilizada na sintetização de voz. Essa técnica consiste na identificação dos componentes fonéticos e sua aplicação no conteúdo linguístico para a criação de uma voz sintética a partir de um texto escrito.

Com os atuais avanços na área de VM, especialmente com base em DNNs, abordagens que contemplam tanto técnicas para a sintetização quanto para conversão da voz estão sendo utilizadas [2]. Um exemplo disso é a utilização de técnicas de *encoding* e *decoding* para a conversão de voz. Portanto, VM pode ser aplicado em

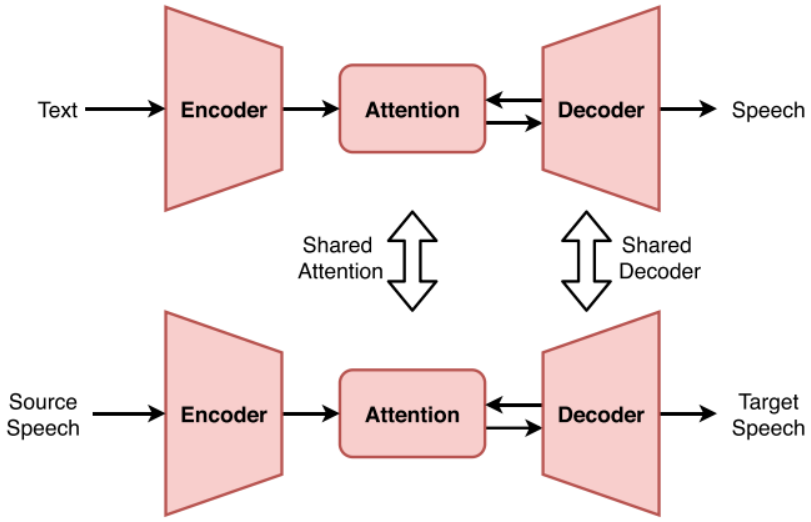


Figura 2.2: Similaridades de mecanismos de encode-decode com atenção sensível à localização para VM e TTS [2].

conjunto com sistemas de TTS para uma personalização na voz sintetizada [13].

A utilização de mecanismos de *encode-decode* em TTS e VM permite o reaproveitamento das características obtidas pelas estruturas inteligentes pois, para esse caso, ambos focam na linguística e fonética. Adicionalmente, a etapa de reconstrução da voz também pode ser reaproveitada, podendo estender um modelo existente de síntese de voz, como na Figura 2.2. Por outro lado, existe a possibilidade de refinar uma voz sintética com a intenção de melhorar a qualidade do sinal para compreensão humana, conforme utilizado em [14] e representado na Figura 2.3.

2.2 Trabalhos Correlatos

No artigo [9], os autores focam em redes neurais com funções *wavelets*, ou seja, *WaveNets*, para realizar VM numa abordagem *one-to-one*. As bases de dados CMU-ARCTIC e CSTR-VCTK foram utilizadas, totalizando 44 horas de gravação proferidas por 109 locutores. Os resultados foram promissores, considerando que apenas uma pequena parcela dos sinais disponíveis foi utilizada para o treinamento do sistema, implicando a capacidade de generalização do modelo testado.

Figura 2.3: Proposta de conversão do sinal de uma laringe eletrônica para melhoria na comunicação [14].

Prosseguindo, no artigo [10], os autores propõem o uso de *stochastic variational deep kernel learning* para a criação do modelo de conversão de voz, tratando-se de uma mistura de modelos estatísticos com redes de aprendizado profundo. Os resultados obtidos implicam a redução de distorções em comparação com outras abordagens, tais como DNN, com melhor qualidade perceptual no caso de aplicações paralelas. A base de dados utilizada foi a CMU ARCTIC, utilizando-se quatro locutores com 20 sentenças de 80 segundos de duração cada.

Interessantemente, no artigo [15], os autores propuseram uma forma de amenizar problemas causados por características espectrais supersuavizadas, durante a afinação da rede neural com núcleos *wavelet*, valendo-se de um mapeamento das características espectrais por meio de uma *cyclic recurrent neural network*. O algoritmo possibilitou uma melhoria na naturalidade do sinal resultante, segundo opinião dos entrevistados, e uma similaridade entre o sinal produzido artificialmente e o real calculado em 78.33%, considerando o total de 14 locutores utilizando das bases CMU Arctic e VCC 2018.

Os autores do artigo [12] apresentaram uma aplicação de conversão de voz em ambientes com ruídos adversos. Com a combinação de duas CNNs e um modelo mais genérico de DNN, foi possível combinar elementos visuais como forma de reduzir o impacto de ruídos durante a conversão de voz. Foram considerados três locutores

chineses com 100 locuções cada, selecionados da *Audio-Visual Whisper Database* (AVWD). O método proposto possibilitou melhores resultados perceptuais em comparação com outros métodos como *multimodal nonnegative matrix factorization*, o qual serviu de base inicial para o trabalho em apreço.

Baseando-se em aprendizado profundo, os autores do artigo [16] utilizaram características do periodograma fonético para melhorar a inteligibilidade da fala de pacientes disártricos, valendo-se de uma estratégia de VM. Considerando dez locutores disártricos e um regular, foram coletadas 288 expressões para treinamento e 32 para teste. Foi observado uma melhoria na compreensão de fala, em testes perceptuais, possibilitando um interessante uso dos algoritmos para VM: converter a fala de um locutor com a fonação deficiente para que se pareça com a fala de um outro locutor específico para o qual a locução é normal.

Para melhorar a naturalidade e qualidade das vozes produzidas, os autores do artigo [13] converteram parâmetros espectrais e características prosódicas com a combinação de modelos *wavelet* e DNN dos contornos dos *pitchs* dos SSs e TSs. Uma base de quatro locutores paralelos em quatro linguagens diferentes, com 50 expressões durando 3 minutos e 58 segundos usadas para treinamento e 10 expressões de 34 segundos para testes, implicou a avaliação perceptual dentro das expectativas, confirmando a viabilidade da proposta dos autores.

No artigo [17], os autores transformam vozes com o uso de *two-level dynamic warping* para o treinamento de uma rede neural, com o uso de informações espectrais dos SSs e TSs. A base CMU ARCTIC foi utilizada com dois locutores, valendo-se de transformações de vozes masculinas para femininas, de vetores de características baseados na análise de Fourier e de segmentação dos sinais em janelas Hamming de 32ms com sobreposição de 16ms. Os resultados implicaram em expressiva semelhança perceptual, validando a proposta dos autores.

Os autores do artigo [18] utilizaram uma DNN para converter, quadro-a-quadro, o *pitch* e demais características espectrais dos SSs para que elas passassem a pa-

recer com as dos TSs correspondentes. Com a utilização da base CMU ARCTIC de quatro locutores, foram utilizadas 100 vocalizações paralelas para treino e testes, implicando melhores resultados perceptuais em comparação com abordagens baseadas em modelos Gaussianos.

Finalmente, no artigo [7], é apresentado um reaproveitamento de treinamento de sintetização de voz para a conversão de voz, considerando dados não paralelos. Com o uso de técnicas de *encoding* e *decoding*, os autores reaproveitam o treinamento linguístico de um sistema TTS para converter vozes. A base de dados LibriTTS, contendo 904 locutores e mais de 191 horas de fala contínua, foi utilizada para o treinamento e para os testes. Por ser uma transferência de treinamento, no geral os resultados obtidos não foram condizentes com as expectativas, porém em algumas comparações perceptuais, considerou-se que ocorreu uma melhoria na qualidade.

2.3 Conceitos de Processamento de Voz Relacionados com VM

As conversões de voz paralelas têm como objetivo mapear o sinal do SS para o TS de modo que a transformação possa ser replicada durante a reconstrução. Porém, mesmo em pares paralelos, os sinais nem sempre possuem o mesmo *timing*, tornando importante a extração de características. Durante a fase de extração de características, padrões são retirados para formar um vetor de características com a intenção de facilitar o pareamento dos sinais.

Ao ser obtido um vetor de características, técnicas como *Dynamic Time Warping* (DTW) têm também sido utilizadas para o pareamento ou sincronização do vetor de características e com isso é possível realizar um treinamento eficiente, como consta na Figura 2.4.

Adicionalmente, a decomposição espectral [1] é um outro assunto de extrema importância em processamento de sinais digitais. Particularmente no caso do processamento de sinais de voz envolvendo VM, dois são os elementos de especial interesse

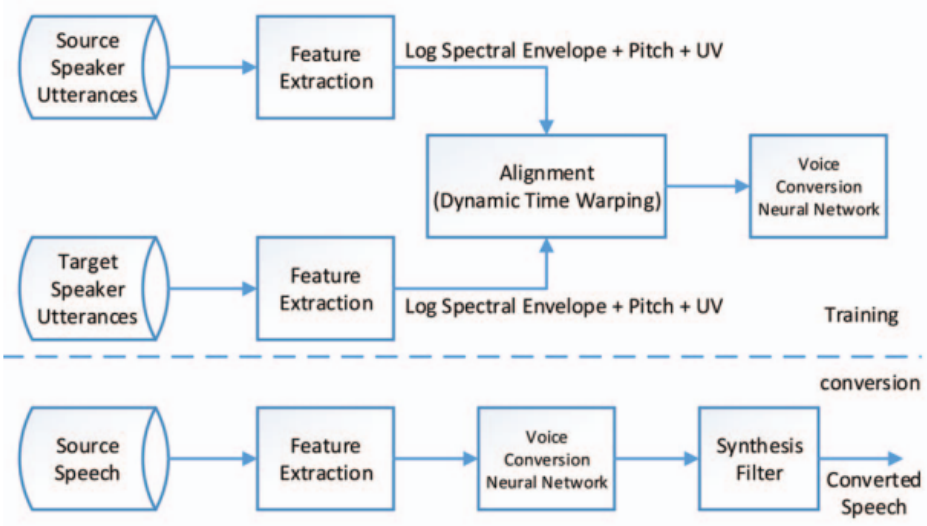


Figura 2.4: Conversão de voz utilizando extração de características, DTW e redes neurais [18].

do ponto de vista espectral: a frequência de *pitch*, comumente representada por F_0, e as frequências formantes, comumente representadas por $F_1, F_2, F_3, ...$, conforme definido adiante.

Notavelmente, o funcionamento básico do aparelho fonador humano é o que está representado na Figura 2.5. Como pode-se observar, a fala, coordenada pelo sistema cognitivo, inicia a sua formação por meio do movimento do diafragma que, pressionando os pulmões, produz um fluxo de ar que fornece a excitação necessária ao procedimento. Em seguida, a referida corrente de ar passa pelas pregas vocais que controlam a forma como ela será expelida: pulsos quase-periódicos, no caso dos chamados sinais vozeados, tal como as vogais ou as consoantes com sons de vogais, ou comportamento aleatório ou ruidoso, no caso dos sinais ditos não-vozeados, tal como as demais consoantes e os sons nasais.

Em seguida, o fluxo de ar com características quase-periódicas ou com comportamento ruidoso atravessa as estruturas vocais e/ou nasais, dependendo do som sendo manifestado, articulando-se com o auxílio da língua e do formato específico da cavidade vocal. Assim, produzem-se os sinais de fala que conhecemos. Nas situações em que as pregas vocais vibram de forma quase-periódica, os sons vozeados

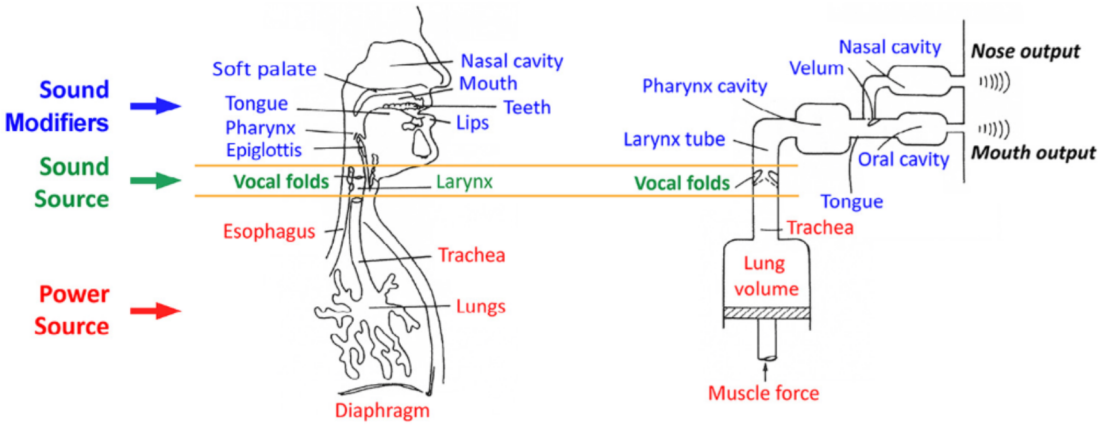


Figura 2.5: Estrutura Vocal Humana [19].

possuem um período fundamental que é justamente F_0. O sinal que atravessa as pregas vocais, entretanto, não é puro no sentido espectral. Contrariamente, ele possui uma forma de onda característica a qual implica a presença de um número considerável de harmônicas, ou seja, frequências múltiplas de F_0. Essas harmônicas são, em função do formato específico da cavidade oral num certo instante, amplificadas ou atenuadas. Desse modo, a primeira, segunda, terceira, e assim por diante, frequências para as quais ocorre maior amplificação são justamente $F_1, F_2, F_3, ...$, respectivamente. Portanto, a identificação dos formantes pode se dar pelo contorno do espectro do sinal de voz, chamado de envelope, conforme a 2.6.

De acordo com o que consta na literatura, de forma consolidada, o *cepstrum* [20] é a técnica mais propícia para extrair F_0 dos sinais de voz. Diferentemente, os formantes podem ser encontrados com base no código de predição linear (LPC) [20]. Além disso, visando rotular um determinado trecho de um sinal de voz como vozeado ou não vozeado, pode-se usar o seguinte procedimento [20]:

- alta taxa de cruzamentos por zero [21] e baixa energia [22]: trecho não-vozeado;
- baixa taxa de cruzamentos por zero e alta energia: trecho vozeado,

onde os termos “alta”e “baixa” dependem de adequada normalização.

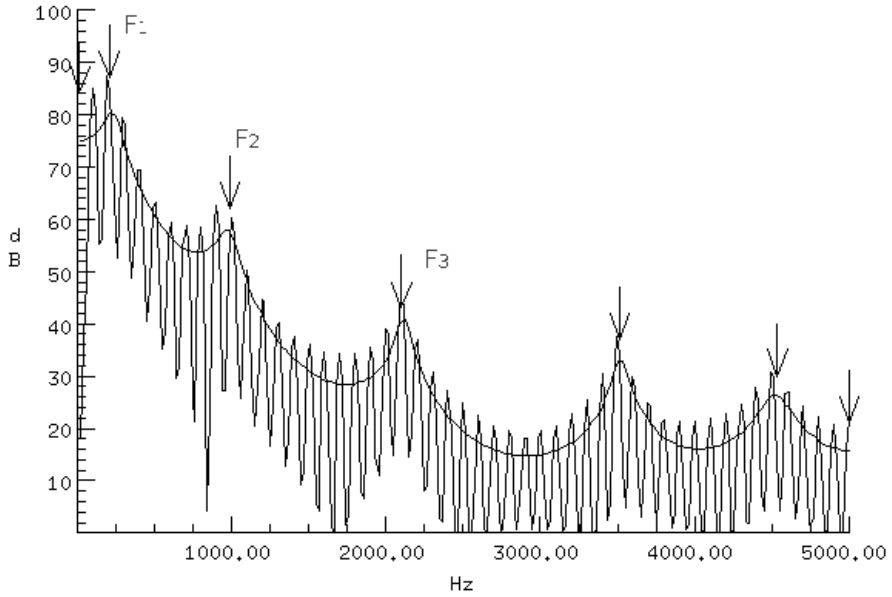


Figura 2.6: Exemplos de formantes, representados por picos, destacando-se F_1, F_2 e F_3 [20].

2.4 A Transformada *Wavelet* de Tempo Discreto (DTWT)

Tendo em vista que a Transformada *Wavelet* de Tempo Discreto (DTWT) é utilizada como peça-chave para decomposição tempo-espectral de sinais na abordagem proposta neste trabalho, uma breve revisão sobre ela é apresentada nesta seção. De acordo com os registros documentados no artigo [23], a DTWT, que é especificamente utilizada para processar sinais discretos, surgiu da adaptação das suas "irmãs" dedicadas ao tratamento de sinais contínuos: a Transformada *Wavelet* Contínua (CWT) e a Transformada *Wavelet* Discreta (DWT), observando que o acrônimo dessa tem também sido largamente utilizado para referir-se à DTWT [23][24].

Nota-se que, para a aplicação tratada nesta dissertação, existe a necessidade de decompor no espaço tempo-frequência o sinal de voz sob análise e, após modificá-lo, ressintetizá-lo. Assim, a DTWT inversa (IDTWT), e não somente a DTWT, é também necessária. Consequentemente, os comentários desta Seção contemplam a DTWT do ponto de vista de filtros digitais com particularidades de resposta em frequência e em fase, assim como do ponto de vista das funções *scaling* ($\phi(\cdot)$) e *wavelet* ($\psi(\cdot)$) associadas e que, implicitamente, fazem parte do processo de reconversão do domínio

tempo-frequência para o domínio do tempo.

Em termos da conversão direta, isto é, do domínio do tempo para o tempo-frequência, cabe destacar que o procedimento de conversão envolve um par de filtros, normalmente designados como $h[\cdot]$ e $g[\cdot]$, respectivamente com respostas em frequência de cunho passa-baixas e passa-altas. Notavelmente, diferentes famílias de filtros $\{h[\cdot]; g[\cdot]\}$ existem, sendo as mais comuns aquelas listadas na Tabela 2.1, todas com comportamento de resposta ao impulso finita (FIR) [23]. Para fins de conversão inversa, ou seja, do domínio tempo-frequência para o do tempo, convém observar o formato das funções $\phi(\cdot)$ e $\psi(\cdot)$, conforme as Figuras 2.7 e 2.8, pois a inversão da DTWT consiste, implicitamente, em sintetizar o sinal original na forma de combinação linear das mesmas.

Finalmente, cabe destacar que o processo prático de implementação da DTWT e da IDTWT é baseado no Algoritmo de Mallat [23], o qual pode ser caracterizado por sequências de multiplicações matriciais, a depender do nível (j) de transformação, como segue.

Tabela 2.1: Características das famílias de *wavelets* utilizadas nesta dissertação. Fonte: Elaborado pelo autor

Família	**Suporte(n)**	**Fase**	**Observação**
Haar	2	linear	é a mais simples das *wavelets*, criada por Alfred Haar
Daubechies	par, maior que 4	não linear	resposta ao impulso *maximally flat*, criada por Ingrid Daubechies
Symmlets	par, múltiplo de 8	não linear	resposta ao impulso mais simétrica
Coiflets	par, múltiplo de 6	quase linear	resposta ao impulso quase simétrica, criada por Ronald Coifman
Vaidyanathan	24	não linear	otimizada para voz, criada por P. P. Vaidyanathan
Beylkin	18	não linear	otimizada para áudio em geral

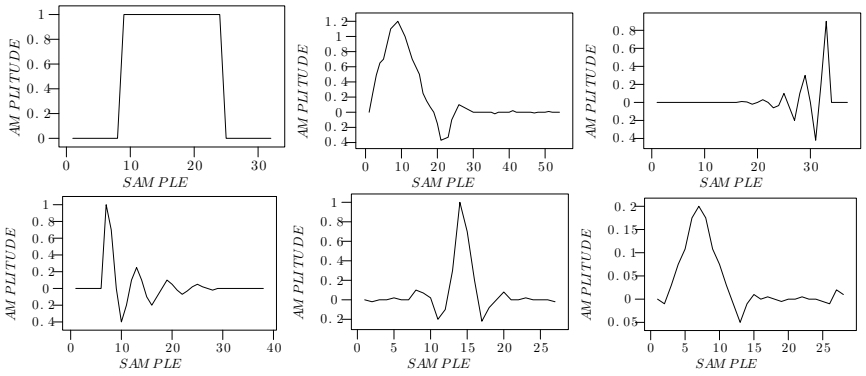


Figura 2.7: Formatos das funções *scaling* dos filtros *wavelet* de Haar, Daubechies, Vaidyanathan, Beylkin, Coiflet e Symmlet, respectivamente. [23]

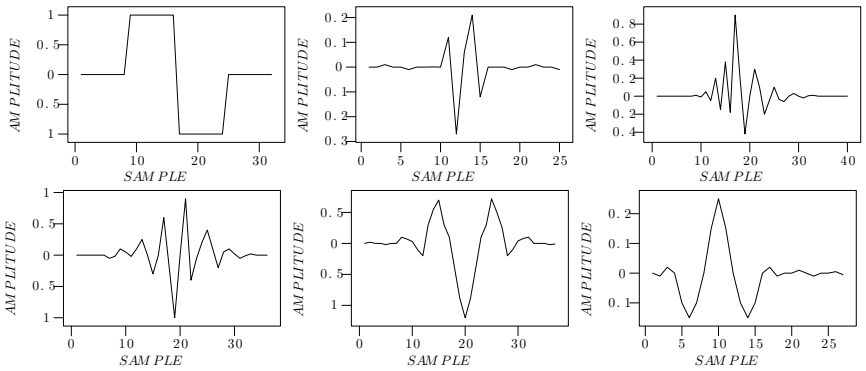


Figura 2.8: Formatos das funções *wavelet* dos filtros *wavelet* de Haar, Daubechies, Vaidyanathan, Beylkin, Coiflet e Symmlet, respectivamente. [23]

O algoritmo de Mallat, que embute os procedimentos de *downsampling* e *wrap-around* [25][26] que forçam a DTWT mantenha sempre o mesmo número de elementos do sinal original, consiste apenas a multiplicação de duas matrizes para cada nível de transformação pretendido. Considerando que $A[\cdot][\cdot]$ é a matriz de coeficientes dos filtros e $B[\cdot]$ é o vetor coluna correspondente ao sinal original, a nova matriz $C[\cdot] = A[\cdot][\cdot]B[\cdot]$ corresponde ao sinal transformado. A disposição dos coeficientes das matrizes é a seguinte:

$$A[\cdot][\cdot] = \begin{pmatrix} h_0 & h_1 & h_2 & \dots & \dots & \dots & \dots & h_{n-1} & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ g_0 & g_1 & g_2 & \dots & \dots & \dots & \dots & g_{n-1} & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & h_0 & h_1 & h_2 & \dots & \dots & \dots & .h_{n-1} & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \dots & \dots & \dots & g_{n-1} & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ . & & & . & & & . & & & . & & & . & & & . \\ . & & & . & & & . & & & . & & & . & & & . \\ . & & & . & & & . & & & . & & & . & & & . \\ h_{n-1} & 0 & 0 & 0 & \dots & \dots & \dots & 0 & 0 & h_0 & h_1 & \dots & \dots & \dots & h_{n-3} & h_{n-2} \\ g_{n-1} & 0 & 0 & 0 & \dots & \dots & \dots & 0 & 0 & g_0 & g_1 & \dots & \dots & \dots & g_{n-3} & g_{n-2} \end{pmatrix} , \quad (2.1)$$

$$B[\cdot] = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ \cdots \\ \cdots \\ \cdot \\ \cdot \\ \cdot \\ b_{n-2} \\ b_{n-1} \end{pmatrix} \quad e \quad C[\cdot] = \begin{pmatrix} c_0 \\ c_{\frac{n}{2}} \\ c_1 \\ c_{\frac{n}{2}+1} \\ \cdots \\ \cdots \\ \cdot \\ \cdot \\ \cdot \\ c_{n-1} \\ c_{\frac{n}{2}-1} \end{pmatrix} . \tag{2.2}$$

A inversão da DTWT implica tão somente na obtenção de $B[\cdot]$ a partir de $A[\cdot][\cdot]$ e $C[\cdot]$ [25][26]. Tendo em vista que $A[\cdot][\cdot]$ é normalmente construída de forma a ser ortogonal, então, $B[\cdot] = (A[\cdot][\cdot])^{-1} \cdot C[\cdot] == (A[\cdot][\cdot])^{T} \cdot C[\cdot]$ pois, em tal caso, a inversa de $A[\cdot][\cdot]$ corresponde à sua própria transposta.

Em cada nível da transformação realizado, dois novos sinais são obtidos e sub-amostrados por 2, pois eles contêm apenas metade da faixa de frequências do sinal original, de acordo com o Teorema de Nyquist [25] e como consta na Figura 2.9. Um sinal de M amostras tem a sua DTWT com o mesmo tamanho, sendo formada por uma sequência de coeficientes, iniciando por aqueles oriundos da aplicação do filtro passa-baixas no último nível, seguidos pelos coeficientes resultantes da aplicação dos filtros passa-altas nos níveis intermediários e finalizando com os coeficientes resultantes da aplicação do filtro passa-altas do primeiro nível de decomposição [25][26]. Melhores resoluções em frequência implicam decomposições até o último nível possível, requerendo que o sinal discreto tenha comprimento equivalente a uma potência de 2.

2.5 Comentários Finais

Diante dos trabalhos revisados, os quais foram extraídos dos registros da *Web of Science* e contemplam o estado-da-arte na área de VM, percebe-se que ainda existe considerável espaço para pesquisas na área. Nenhum dos métodos revisados possibilitou uma solução definitiva dos problemas comumente encontrados nos sistemas de VM. Além disso, em geral, as avaliações costumam ser realizadas com base em critérios puramente perceptuais, criando possíveis vieses. Desse modo, reunindo conceitos envolvendo decomposição tempo-frequência por meio de *wavelets*, assim como estruturas inteligentes, a ideia base deste trabalho não é a de converter as vo-

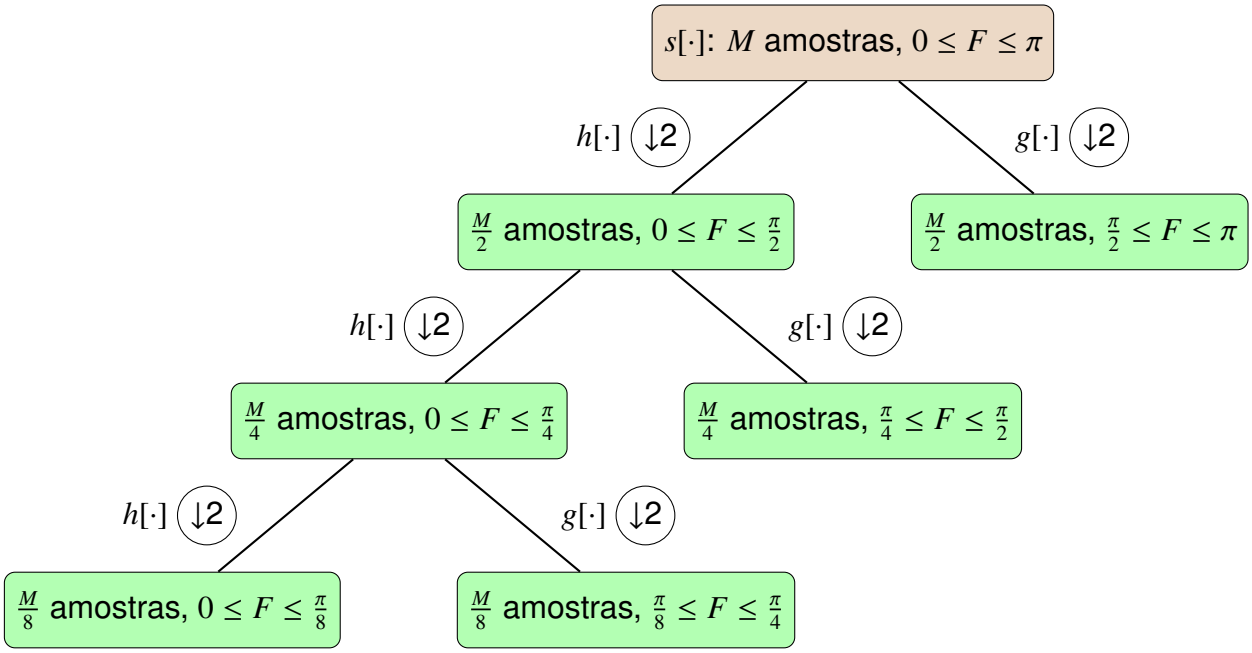


Figura 2.9: Exemplo da aplicação da DTWT: o sinal original $s[\cdot]$, no domínio do tempo e contendo M amostras, com máxima frequência π, decomposto até o terceiro nível. Elaborado pelo autor.

zes dos SSs para que se pareçam com as dos TSs mas, diferentemente, converter apenas os parâmetros elementares, isto é, *pitch* e frequências formantes, por meio de uma estrutura que funcionará como função de transferência. Os detalhes encontram-se descritos no próximo Capítulo.

A ABORDAGEM PROPOSTA

Os procedimentos metodológicos utilizados para o desenvolvimento da abordagem proposta nesta dissertação seguem a estratégia esboçada na Figura 3.1. Inicialmente, os sinais de vozes originais, pertencentes ao *source-speaker*, foram divididos em segmentos com 32ms, correspondente a 512 amostras de sinais amostrados com base na taxa de 16000 amostras por segundo e coerente com o que tem sido registrado na literatura [6]. Em seguida, cada trecho foi classificado como não-vozeado ou vozeado. Aqueles trechos tiveram "passagem livre" pelo algoritmo conversor, isto é, não foram alterados. Diferentemente, estes trechos foram convertidos por um processo de análise e ressíntese. O referido procedimento foi utilizado pois, de acordo com trabalhos anteriores [6], os trechos vozeados é que são os responsáveis pela percepção da voz de modo a caracterizar o locutor, sendo mínima ou quase insignificante a contribuição dos trechos não-vozeados, em função das suas naturezas ruidosas e parcialmente aleatórias.

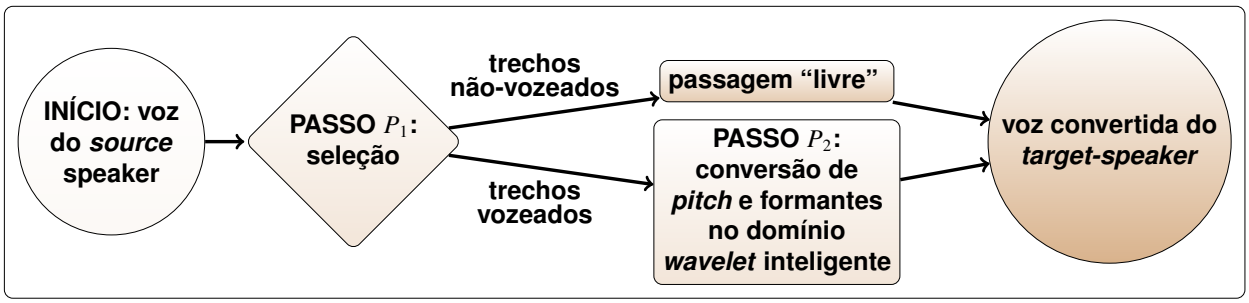


Figura 3.1: Estrutura da estratégia proposta. Destacam-se os passos P_1 e P_2, detalhados adiante. Fonte: Elaborado pelo autor

Particularmente, dois destaques existem na Figura 3.1: um referente ao passo P_1 e outro referente ao passo P_2. A especificação de cada um deles é a seguinte:

- **PASSO** P_1: distinguem-se os trechos vozeados dos não-vozeados de um sinal de voz $s[\cdot]$, visando aplicar o processo de conversão de locutor naqueles e man-

ter estes intactos, do seguinte modo [21]: cada segmento não-vozeado de $s[\cdot]$ implica uma região com alta e inconstante taxa de cruzamentos por zero e com alta energia. Diferentemente, partes vozeadas de $s[\cdot]$ exibem baixa e relativamente constante taxa de cruzamentos por zero e com alta energia. Além disso, espaços entre palavras, que correspondem aos segmentos de silêncio, possuem baixa energia em $s[\cdot]$. Definem-se as energias e as taxas de cruzamentos por zero em $s[\cdot]$ como sendo altas ou baixas por meio dos limiares H_A e H_B, respectivamente.

Além disso, definem-se os vetores $f_A[\cdot]$ e $f_B[\cdot]$, ambos de tamanho T, como sendo, respectivamente, aquele que mantém o registro das energias normalizadas de $s[\cdot]$ [22] e aquele que mantém o registro do número de cruzamentos por zero normalizados de $s[\cdot]$ [21]. Seguiu-se, então, o seguinte algoritmo [21]:

INÍCIO
Para a k-ésima amostra do vetor $f_A[\cdot]$ e $f_B[\cdot]$, $(0 \leqslant k \leqslant T-1)$:
{
Se ($f_{A_k} < H_A$)
a região correspondente em $s[\cdot]$ é de silêncio, ou seja, é uma região entre palavras, e, assim, não interessa ao processo;
senão se ($f_{B_k} < H_B$)
a região correspondente em $s[\cdot]$ é vozeada e é marcada para o processo de conversão;
senão
a região correspondente em $s[\cdot]$ é não-vozeada e, assim, não interessa ao processo.
}
FIM.

Notavelmente, as taxas de cruzamento por zero de sinais vozeados e não vozeados são, respectivamente, próximas de 1400 and 4900 cruzamentos por segundo [21]. Dessa forma, um valor razoável para H_B corresponde à média, isto é, $\frac{1400+4900}{2} = 3150$ cruzamentos por segundo. Para janelas de 32ms, $H_B =$

$3150 \cdot 0.032 = 100.8$ cruzamentos por 512 amostras. Normalizando, tem-se $H_B = \frac{100.8}{512-1} = \frac{100.8}{511} \approx 0.197$, que é o valor adotado.

Em termos de energia, foi observado [21] que o limiar da audição para os seres humanos é de 0 dB na intensidade de 20μPa, ou seja, 20 micro-Pascal, na frequência de 1000 Hz e na temperatura de $25°C$. Além disso, também conforme observado em [21], para comparar um sinal falado com o limiar da audição, o nível de reprodução deveria ser conhecido, o que não é fácil na prática. Entretanto, é comum definir tal nível como a menor quantidade que pode ser representada pelo processo de discretização e de quantização adotado para digitalizar o sinal sob análise, ou seja, é o menor nível de energia representado pelo bit menos significante. Tendo em vista que os sinais sob análise, conforme detalhado no próximo Capítulo, foram quantizados com 16 bits, sendo um deles reservado para a diferenciação entre valores positivos e negativos, $\frac{1}{2^{16-1}} = \frac{1}{2^{15}}$ é o menor valor possível. Considerando janelas de tamanho 512 amostras, $512 \cdot \frac{1}{2^{15}} = \frac{2^9}{2^{15}} = 2^{-6} = 0.015625 \approx 0.016$ foi o valor adotado para o limiar H_A.

- **PASSO** P_2: Para a conversão do *pitch* e dos formantes selecionados no passo anterior, a técnica aqui usada é baseada na decomposição *wavelet* com a DTWT atuando em conjunto com uma função de transferência inteligente implementada com base em uma rede neural perceptron multicamadas, de acordo com as Figuras 3.2 e 3.3. Particularmente, o objetivo foi escolher um nível de decomposição da DTWT que implicasse uma árvore capaz de, em geral, separar o *pitch* dos formantes, visando um esforço menor para treinar a função inteligente de conversão, que atua conjuntamente com a DTWT. Assim, uma vez que a árvore DTWT de um certo sinal decomposto em um determinado nível tenha sido obtida, cada sub-banda é modificada pela rede neural para, posteriormente, ser submetida à DTWT inversa.

Em termos da DTWT, observou-se que para sinais amostrados à 16000 amostras por segundo, a máxima frequência presente, de acordo com o Teorema da Amostragem, é de $\frac{16000}{2} = 8000$ Hz. Assim, para uma decomposição de nível $j = 5$, a resolução da sub-banda mais profunda é de $\frac{8000}{2^4} = 500$ Hz, que é su-

ficiente para contemplar o *pitch* praticamente da totalidade dos sinais de vozes reais. Nas camadas menos profundas, a partir da mais profunda, as resoluções seriam $\frac{8000}{2^3} = 1000$ Hz, $\frac{8000}{2^2} = 2000$ Hz e $\frac{8000}{2^1} = 4000$ Hz, as quais, possivelmente, contemplariam, cada qual, um dos formantes, isto é, F_1, F_2 e F_3, de acordo com a Figura 3.4.

Com relação à estrutura das cinco redes neurais adotadas, de acordo com as Figuras 3.2, 3.3 e 3.4, o número de neurônios nas diversas camadas que as compõem são aqueles registrados na Tabela 3.1. Nas camadas iniciais de todas as cinco redes, os neurônios são sempre passivos. Nas demais camadas, a função de ativação de cada neurônio é a tangente hiperbólica, ou seja,

$$\text{saída} = \frac{2}{1 + e^{-\text{entrada}}} - 1 \qquad .$$

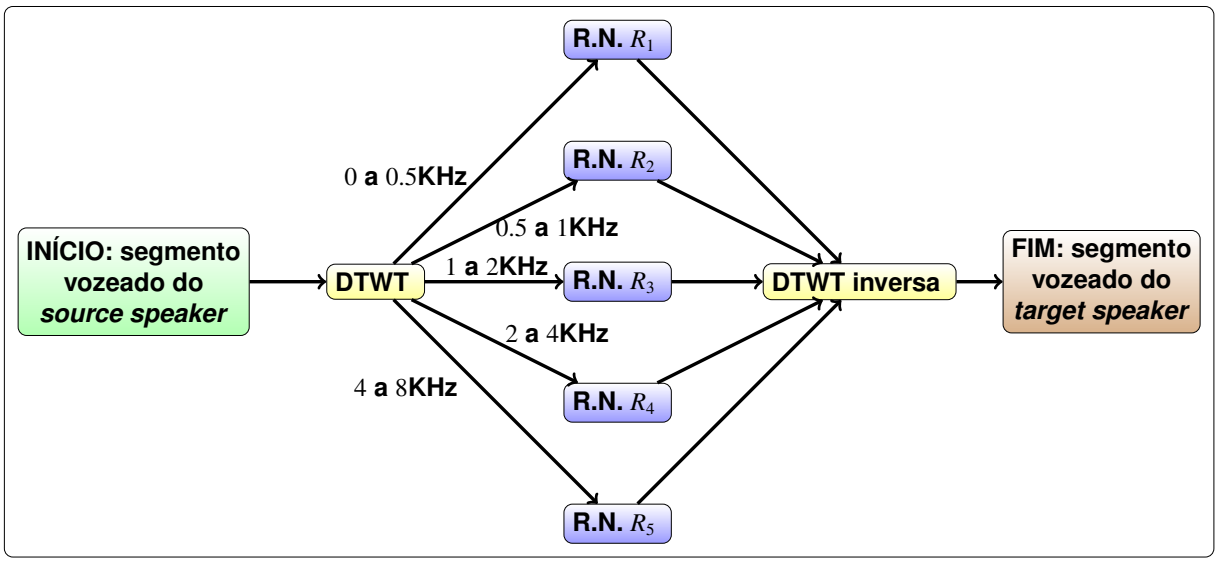


Figura 3.2: Estratégia de conversão dos segmentos vozeados dos sinais de voz sob análise com base nas 5 sub-bandas *wavelet* indicadas também na Figura 3.4 e nas cinco redes neurais (R.N.), ou seja, $R_1, R_2, R_3, R4$ e R_5. Fonte: Elaborado pelo autor

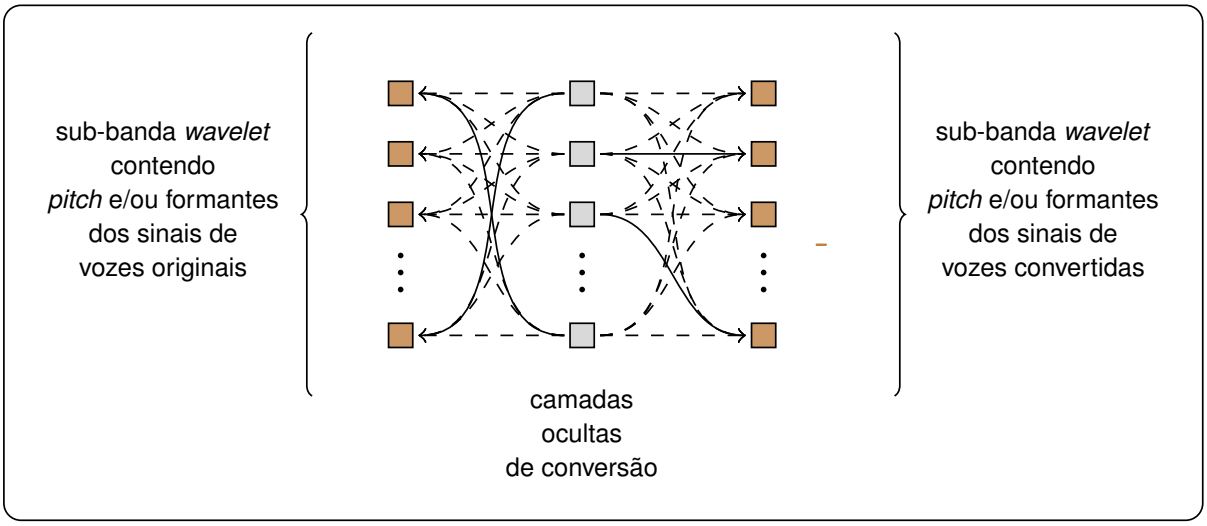


Figura 3.3: A estratégia de conversão baseada em estruturas profundas de aprendizagem: uma para cada uma das cinco sub-bandas *wavelet* da Figura 3.4. Fonte: Elaborado pelo autor

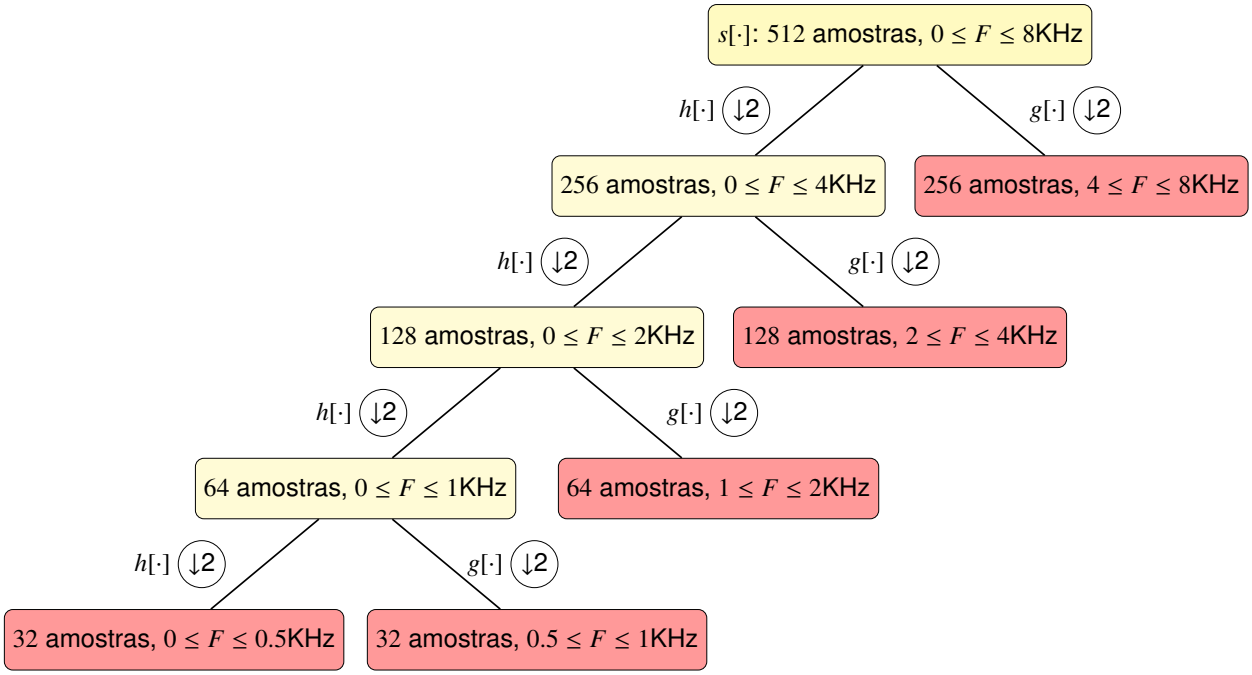


Figura 3.4: As cinco sub-bandas da DTWT para as quais os segmentos vozeados dos sinais de voz são convertidos com a estrutura inteligente baseada nas cinco respectivas redes neurais aparecem na cor vermelha. As faixas de frequências F em cada sub-bandas estão indicadas. Elaborado pelo autor.

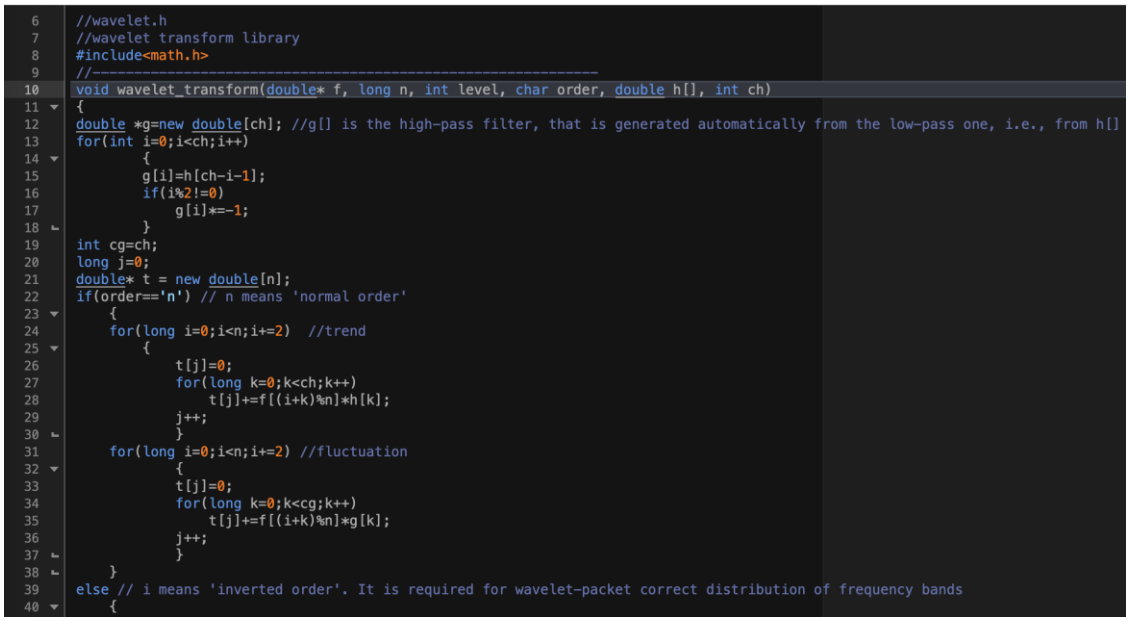
Tabela 3.1: Configuração das cinco redes neurais usadas no experimento. Na terceira coluna, que expressa o número de neurônios em cada camada intermediária (NNCI), foi usada a regra NNCI = $\sqrt{\text{DCE} \cdot \text{DCS}}$, sendo DCE e DCS respectivamente dimensão da camada de entrada e dimensão da camada de saída [27]. Experimentalmente, foi constatada a suficiência de três camadas intermediárias em cada rede. Fonte: Elaborado pelo autor

rede neural	**neurônios nas camadas de entrada e de saída**	**número de camadas ocultas e número de neurônios em cada uma**
R_1	32	3 camadas com $\sqrt{32 \cdot 32} = 32$ neurônios.
R_2	32	3 camadas com $\sqrt{32 \cdot 32} = 32$ neurônios.
R_3	64	3 camadas com $\sqrt{64 \cdot 64} = 64$ neurônios.
R_4	128	3 camadas com $\sqrt{128 \cdot 128} = 128$ neurônios.
R_5	256	3 camadas com $\sqrt{256 \cdot 256} = 256$ neurônios.

Notavelmente, para fins de treinamento das redes neurais, baseado no clássico algoritmo do gradiente descendente e retropropagação do erro [27], cada trecho vozeado do *source speaker* no domínio *wavelet* apresentado na entrada das redes foi convertido de forma a minimizar o erro em relação ao respectivo trecho vozeado do *target speaker* no domínio *wavelet*.

Questões referentes ao alinhamento dos pares de sinais {*source speaker ; target speaker*} foram balizadas manualmente, não tendo sido implementado um sistema automático para tal. Isso se deu em função do expressivo volume de trabalho que as demais etapas do processos requereram e, além disso, pelo fato de que as etapas implementadas para funcionamento automático foram suficientes para viabilizar os resultados referentes à essência do processo de VM.

No próximo Capítulo, os resultados são apresentados, discutindo-se a influência do filtro *wavelet* utilizado e do número de camadas ocultas nas redes neurais. As implementações foram realizadas em linguagem C/C++ para fins da extração dos trechos vozeados dos sinais, assim como das DTWTs direta e inversa e também das redes neurais, conforme os exemplos constantes nas Figuras 3.5 e 3.6.



```
//wavelet.h
//wavelet transform library
#include<math.h>
//---------------------------------------------------------------
void wavelet_transform(double* f, long n, int level, char order, double h[], int ch)
{
double *g=new double[ch]; //g[] is the high-pass filter, that is generated automatically from the low-pass one, i.e., from h[]
for(int i=0;i<ch;i++)
        {
        g[i]=h[ch-i-1];
        if(i%2!=0)
            g[i]*=-1;
        }
int cg=ch;
long j=0;
double* t = new double[n];
if(order=='n') // n means 'normal order'
    {
    for(long i=0;i<n;i+=2)  //trend
        {
            t[j]=0;
            for(long k=0;k<ch;k++)
                t[j]+=f[(i+k)%n]*h[k];
            j++;
            }
    for(long i=0;i<n;i+=2) //fluctuation
            {
            t[j]=0;
            for(long k=0;k<cg;k++)
                t[j]+=f[(i+k)%n]*g[k];
            j++;
            }
    }
else // i means 'inverted order'. It is required for wavelet-packet correct distribution of frequency bands
    {
```

Figura 3.5: Trecho de código em linguagem C/C++ para implementação da DTWT e sua inversa. Fonte: Elaborado pelo autor

```
        hidden_weights[i][j]=time(0)*rand();
        while(hidden_weights[i][j]>1)
            hidden_weights[i][j]/=10.0;
        hidden_weights[i][j]-=0.5;
        }
double output_weights[dimension_of_output_layer][dimension_of_hidden_layer];
for(int i=0;i<dimension_of_output_layer;i++)
    for(int j=0;j<dimension_of_hidden_layer;j++)
        {
        output_weights[i][j]=time(0)*rand();
        while(output_weights[i][j]>1)
            output_weights[i][j]/=10.0;
        output_weights[i][j]-=0.5;
        }
double gotten_hiddens[dimension_of_hidden_layer];
double gotten_outputs[dimension_of_output_layer];
double e;
double delta_output[dimension_of_output_layer];
double delta_hidden[dimension_of_hidden_layer];
double mu=0.000001;
do
    {
    e=0;
    for(int k=0;k<number_of_elements_for_training;k++)
        {
        for(int i=0;i<dimension_of_hidden_layer;i++)
            {
            gotten_hiddens[i]=0;
            for(int j=0;j<dimension_of_input_layer;j++)
                    gotten_hiddens[i]+=hidden_weights[i][j]*s[k+j];
            gotten_hiddens[i]=tanh(gotten_hiddens[i]);
            }
        for(int i=0;i<dimension_of_output_layer;i++)
            {
            gotten_outputs[i]=0;
            for(int j=0;j<dimension_of_hidden_layer;j++)
                gotten_outputs[i]+=output_weights[i][j]*gotten_hiddens[j];
            gotten_outputs[i]=tanh(gotten_outputs[i]);
            }
        ////////////////////////////////////////////
```

Figura 3.6: Trecho de código em linguagem C/C++ para implementação das redes neurais. Fonte: Elaborado pelo autor

TESTES E RESULTADOS

Neste Capítulo, descrevem-se os testes realizados com o algoritmo de VM proposto no capítulo anterior, seguindo o padrão adotado em [6]. Tais testes foram realizados com o objetivo de avaliar a qualidade das vozes convertidas de dois modos: um quantitativo, com base em distâncias Euclidianas, e outro qualitativo-subjetivo, com base em avaliações perceptuais, de acordo com os procedimentos registrados em [6]. As locuções usadas são de conteúdo fonético rico, contendo trechos diversos vozeados e não-vozeados. Nos testes realizados, todas as vozes, que são oriundas da base TIMIT [28] do *Linguistic Data Consortium*, foram digitalizadas com 16000 amostras por segundo, 16-bits, sem compressão, implicando uma frequência máxima de 8 KHz.

4.1 Testes de Convergência das Redes Neurais

Os testes de convergência das redes $\{R_1, ..., R_5\}$, variando-se as *wavelets*, estão listados nas Tabelas 4.1 e 4.2, 4.3 e 4.4, 4.5 e 4.6, 4.7 e 4.8, assim como 4.9 e 4.10, respectivamente para as redes $R_1, ..., R_5$, na qual SP indica o tamanho de suporte utilizado e QM quantidade máxima de iterações para se obter um erro de $< 10^{-3}$. Notavelmente, na parte superior das Tabelas 4.1, 4.3, 4.5, 4.7 e 4.9, a convergência foi analisada para a sentença *sa1.wav* do diretório */timit/test/ dr1/mdab0* da base TIMIT utilizada como *source speech*, para ser convertida na sentença *sa1.wav* do diretório */timit/test/dr1/mjsw0*, da mesma base e de mesmo conteúdo textual, sendo que ambos os locutores são do sexo masculino. Seguindo o mesmo procedimento, a parte inferior das referidas tabelas contêm as descrições dos resultados obtidos quando a sentença *sa1.wav* do diretório */timit/test/dr1/faks0* é convertida para a *sa1.wav* do diretório */timit/test/dr1/fdac1*, sendo ambas de mesmo conteúdo falado, mas com ambos os locutores do sexo feminino.

Prosseguindo, a parte superior das Tabela 4.2, 4.4, 4.6, 4.8 e 4.10 contêm os resultados obtidos quando a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* é convertida para a *sa1.wav* do diretório */timit/test/dr1/fjem0*, sendo ambas de mesmo conteúdo textual, considerando o *source speaker* do sexo masculino e o *target speaker* do sexo feminino. Finalmente, a parte inferior das referidas tabelas contêm os testes contrários, isto é, quando a sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* é convertida para a *sa1.wav* do diretório */timit/test/dr1/mwbt0*, caracterizando conversão de voz de um locutor de sexo feminino para o sexo masculino.

É importante observar que um maior número de iterações máximas está associado a um desempenho considerado inferior no algoritmo avaliado. Além disso, em alguns casos, diferentes tamanhos de suporte foram empregados no mesmo algoritmo para fins de comparação de desempenho entre eles.

4.2 Testes de comparação dos resultados das conversões com distância Euclidiana

Os resultados dos testes comparativos das vozes convertidas em relação às suas versões originais, levando-se em conta a distância Euclidiana entre os sinais original e convertido, estão registrados nas Tabelas 4.11 e 4.12, lembrando que a distância Euclidiana (DE) entre dois vetores $x[\cdot]$ e $y[\cdot]$, ambos de comprimento M, é

$$\text{DE} = \sqrt{\sum_{i=0}^{M-1} (x_i - y_i)^2} \quad .$$

Adicionalmente, na parte superior da Tabela 4.11, a convergência foi analisada para a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* da TIMIT, utilizada como *source speech*, visando a conversão para a sentença *sa1.wav* do diretório */timit/test/dr1/mjsw0*, ambas com o mesmo conteúdo textual, sendo que ambos os locutores são do sexo masculino. Do mesmo modo, a parte inferior da referida tabela contém os resultados obtidos quando a sentença *sa1.wav* do diretório */timit/test/dr1/faks0* é convertida para a *sa1.wav* do diretório */timit/test/dr1/fadac1*, de mesmo conteúdo

textual mas com ambos os locutores femininos.

Em complemento, a parte superior da Tabela 4.12 contém os resultados obtidos durante a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* para a sentença *sa1.wav* do diretório */timit/test/dr1/fjem0*, de mesmo conteúdo textual, com o *source speaker* masculino e o *target speaker* feminino. Finalmente, a parte inferior da mesma tabela contempla o teste contrário, isto é, quando a sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* é convertida para a *sa1.wav* do diretório */timit/test/dr1/mwbt0*, caracterizando uma conversão de um locutor do sexo feminino para um do sexo masculino.

4.3 Testes de comparação dos resultados das conversões via critério perceptual

Os testes visando comparar os sinais de voz convertidos com suas versões pronunciadas pelos locutores originais, que estão disponíveis na base TIMIT, levando-se em conta a avaliação perceptual entre os sinais original e convertido, estão registrados nas Tabelas 4.1, 4.2, 4.3 e 4.4. Para viabilizar tal comparação, 8 ouvintes voluntários, sendo 4 do sexo masculino e 4 do sexo feminino, de idades entre 19 e 70 anos, foram entrevistados e avaliaram cada uma das conversões numa escala de 0 até 5, sendo que 0 significa que a voz original não se parece em nada com a voz convertida e 5 significa que ambas soam perceptualmente indistinguíveis. Notavelmente, registre-se que, para todas as sentenças, originais e convertidas, foi possível compreender normalmente o que está sendo pronunciado, implicando que as conversões foram inteligíveis.

4.4 Comentários Finais

De modo geral, os resultados foram consistentes com as expectativas e semelhantes aos obtidos no estudo documentado em [6], e abrange transformações de sexo iguais e distintas. O filtro Daubechies demonstrou um desempenho superior à medida que o suporte aumentou, até certo limite, enquanto outros filtros mantiveram

um bom desempenho em diferentes testes, como Vaidyanathan e Belkyn.

Os testes perceptuais, que foram subjetivos, complementam-se com os testes baseados em distâncias Euclidianas, visando quantificá-los com mais padronização e menos subjetividade. No próximo Capítulo, as conclusões são apresentadas com base nos resultados descritos neste Capítulo.

[MASCULINO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2151	Daubechies	4	522	Daubechies	6	515	Daubechies	8	508	Daubechies	10	507	Daubechies	12	507
Daubechies	14	496	Daubechies	16	489	Daubechies	18	488	Daubechies	20	481	Daubechies	22	481	Daubechies	24	480
Daubechies	26	489	Daubechies	28	482	Daubechies	30	481	Daubechies	32	483	Daubechies	34	482	Daubechies	36	483
Daubechies	38	486	Daubechies	40	473	Daubechies	42	474	Daubechies	44	477	Daubechies	46	474	Daubechies	48	473
Daubechies	50	484	Daubechies	52	470	Daubechies	54	476	Daubechies	56	477	Daubechies	58	456	Daubechies	60	455
Daubechies	62	455	Daubechies	64	450	Daubechies	66	443	Daubechies	68	443	Daubechies	70	441	Daubechies	72	440
Daubechies	74	476	Daubechies	76	445	Coiflet	6	1687	Coiflet	12	1688	Coiflet	18	1709	Coiflet	24	1788
Coiflet	30	1800	Symmlet	8	2113	Symmlet	16	2039	Beylkin	18	1965	Vaidyanathan	24	477			

[FEMININO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	428	Daubechies	4	613	Daubechies	6	607	Daubechies	8	597	Daubechies	10	593	Daubechies	12	599
Daubechies	14	598	Daubechies	16	587	Daubechies	18	587	Daubechies	20	583	Daubechies	22	581	Daubechies	24	572
Daubechies	26	538	Daubechies	28	522	Daubechies	30	533	Daubechies	32	532	Daubechies	34	521	Daubechies	36	520
Daubechies	38	424	Daubechies	40	423	Daubechies	42	422	Daubechies	44	420	Daubechies	46	413	Daubechies	48	413
Daubechies	50	409	Daubechies	52	396	Daubechies	54	395	Daubechies	56	392	Daubechies	58	390	Daubechies	60	381
Daubechies	62	368	Daubechies	64	365	Daubechies	66	360	Daubechies	68	360	Daubechies	70	362	Daubechies	72	356
Daubechies	74	355	Daubechies	76	356	Coiflet	6	458	Coiflet	12	454	Coiflet	18	447	Coiflet	24	446
Coiflet	30	448	Symmlet	8	1817	Symmlet	16	1789	Beylkin	18	413	Vaidyanathan	24	165			

Tabela 4.1: Com relação à rede neural R_1, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* na *sa1.wav* do diretório */timit/test/dr1/mjsw0*, onde ambos os locutores, isto é, *source* e *target* são do sexo masculino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/faks0* na *sa1.wav* do diretório */timit/test/dr1/fadac1*, onde ambos os locutores são do sexo feminino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO ⟶ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2034	Daubechies	4	2063	Daubechies	6	2056	Daubechies	8	2066	Daubechies	10	2057	Daubechies	12	2051
Daubechies	14	2038	Daubechies	16	2041	Daubechies	18	2042	Daubechies	20	2038	Daubechies	22	2033	Daubechies	24	2027
Daubechies	26	2026	Daubechies	28	2021	Daubechies	30	1991	Daubechies	32	1993	Daubechies	34	1993	Daubechies	36	1991
Daubechies	38	1985	Daubechies	40	1985	Daubechies	42	1984	Daubechies	44	1985	Daubechies	46	1982	Daubechies	48	1976
Daubechies	50	1974	Daubechies	52	1963	Daubechies	54	1967	Daubechies	56	1956	Daubechies	58	1951	Daubechies	60	1943
Daubechies	62	1944	Daubechies	64	1942	Daubechies	66	1938	Daubechies	68	1937	Daubechies	70	1937	Daubechies	72	1936
Daubechies	74	1935	Daubechies	76	1923	Coiflet	6	618	Coiflet	12	599	Coiflet	18	594	Coiflet	24	583
Coiflet	30	536	Symmlet	8	561	Symmlet	16	532	Beylkin	18	435	Vaidyanathan	24	175			

[FEMININO ⟶ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2		Daubechies	4	2081	Daubechies	6	2130	Daubechies	8	2129	Daubechies	10	2103	Daubechies	12	2100
Daubechies	14	2096	Daubechies	16	2094	Daubechies	18	2091	Daubechies	20	2088	Daubechies	22	2084	Daubechies	24	2074
Daubechies	26	2070	Daubechies	28	2065	Daubechies	30	2065	Daubechies	32	2062	Daubechies	34	2043	Daubechies	36	2041
Daubechies	38	2031	Daubechies	40	1929	Daubechies	42	1912	Daubechies	44	1902	Daubechies	46	1899	Daubechies	48	1895
Daubechies	50	1893	Daubechies	52	1891	Daubechies	54	1891	Daubechies	56	1860	Daubechies	58	1859	Daubechies	60	1861
Daubechies	62	1845	Daubechies	64	1830	Daubechies	66	1840	Daubechies	68	1842	Daubechies	70	1841	Daubechies	72	1832
Daubechies	74	1930	Daubechies	76	1921	Coiflet	6	756	Coiflet	12	718	Coiflet	18	695	Coiflet	24	612
Coiflet	30	602	Symmlet	8	597	Symmlet	16	588	Beylkin	18	487	Vaidyanathan	24	434			

Tabela 4.2: Com relação à rede neural R_1, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* na *sa1.wav* do diretório */timit/test/dr1/fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* na *sa1.wav* do diretório */timit/test/dr1/mwbt0*, onde *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2151	Daubechies	4	601	Daubechies	6	615	Daubechies	8	560	Daubechies	10	541	Daubechies	12	513
Daubechies	14	498	Daubechies	16	485	Daubechies	18	484	Daubechies	20	479	Daubechies	22	472	Daubechies	24	472
Daubechies	26	483	Daubechies	28	480	Daubechies	30	472	Daubechies	32	475	Daubechies	34	466	Daubechies	36	469
Daubechies	38	465	Daubechies	40	459	Daubechies	42	460	Daubechies	44	446	Daubechies	46	451	Daubechies	48	430
Daubechies	50	470	Daubechies	52	465	Daubechies	54	451	Daubechies	56	440	Daubechies	58	445	Daubechies	60	434
Daubechies	62	433	Daubechies	64	423	Daubechies	66	427	Daubechies	68	418	Daubechies	70	417	Daubechies	72	408
Daubechies	74	406	Daubechies	76	404	Coiflet	6	1890	Coiflet	12	1810	Coiflet	18	1765	Coiflet	24	1776
Coiflet	30	1654	Symmlet	8	2200	Symmlet	16	2138	Beylkin	18	1916	Vaidyanathan	24	416			

[FEMININO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	436	Daubechies	4	620	Daubechies	6	565	Daubechies	8	554	Daubechies	10	539	Daubechies	12	538
Daubechies	14	548	Daubechies	16	535	Daubechies	18	555	Daubechies	20	524	Daubechies	22	520	Daubechies	24	515
Daubechies	26	512	Daubechies	28	506	Daubechies	30	507	Daubechies	32	503	Daubechies	34	504	Daubechies	36	497
Daubechies	38	465	Daubechies	40	432	Daubechies	42	412	Daubechies	44	410	Daubechies	46	400	Daubechies	48	394
Daubechies	50	399	Daubechies	52	354	Daubechies	54	365	Daubechies	56	343	Daubechies	58	344	Daubechies	60	347
Daubechies	62	332	Daubechies	64	333	Daubechies	66	334	Daubechies	68	321	Daubechies	70	312	Daubechies	72	310
Daubechies	74	304	Daubechies	76	298	Coiflet	6	293	Coiflet	12	292	Coiflet	18	289	Coiflet	24	279
Coiflet	30	281	Symmlet	8	1660	Symmlet	16	1678	Beylkin	18	410	Vaidyanathan	24	211			

Tabela 4.3: Com relação à rede neural R_2, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* na *sa1.wav* do diretório */timit/test/dr1/mjsw0*, onde ambos os locutores, isto é, *source* e *targe* são do sexo masculino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/faks0* na *sa1.wav* do diretório */timit/test/dr1/fadac1*, onde ambos os locutores são do sexo feminino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO ⟶ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	3010	Daubechies	4	3150	Daubechies	6	3058	Daubechies	8	3068	Daubechies	10	3057	Daubechies	12	3038
Daubechies	14	3009	Daubechies	16	2987	Daubechies	18	2835	Daubechies	20	2933	Daubechies	22	2870	Daubechies	24	2778
Daubechies	26	2876	Daubechies	28	2889	Daubechies	30	2843	Daubechies	32	2897	Daubechies	34	2876	Daubechies	36	2860
Daubechies	38	2875	Daubechies	40	2875	Daubechies	42	2876	Daubechies	44	2875	Daubechies	46	2873	Daubechies	48	2872
Daubechies	50	2871	Daubechies	52	2863	Daubechies	54	2861	Daubechies	56	2860	Daubechies	58	2855	Daubechies	60	2850
Daubechies	62	2859	Daubechies	64	2846	Daubechies	66	2848	Daubechies	68	2855	Daubechies	70	2845	Daubechies	72	2840
Daubechies	74	2835	Daubechies	76	2839	Coiflet	6	660	Coiflet	12	630	Coiflet	18	599	Coiflet	24	578
Coiflet	30	556	Symmlet	8	530	Symmlet	16	542	Beylkin	18	425	Vaidyanathan	24	176			

[FEMININO ⟶ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2		Daubechies	4	2154	Daubechies	6	2140	Daubechies	8	2049	Daubechies	10	2038	Daubechies	12	2027
Daubechies	14	2018	Daubechies	16	2017	Daubechies	18	2015	Daubechies	20	2005	Daubechies	22	2003	Daubechies	24	2006
Daubechies	26	2002	Daubechies	28	2003	Daubechies	30	2002	Daubechies	32	2001	Daubechies	34	2000	Daubechies	36	2001
Daubechies	38	2001	Daubechies	40	1998	Daubechies	42	1998	Daubechies	44	1978	Daubechies	46	1975	Daubechies	48	1977
Daubechies	50	1968	Daubechies	52	1964	Daubechies	54	1963	Daubechies	56	1962	Daubechies	58	1948	Daubechies	60	1951
Daubechies	62	1951	Daubechies	64	1947	Daubechies	66	1946	Daubechies	68	1942	Daubechies	70	1923	Daubechies	72	1924
Daubechies	74	1921	Daubechies	76	1919	Coiflet	6	843	Coiflet	12	850	Coiflet	18	832	Coiflet	24	821
Coiflet	30	801	Symmlet	8	689	Symmlet	16	661	Beylkin	18	475	Vaidyanathan	24	440			

Tabela 4.4: Com relação à rede neural R_2, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* na *sa1.wav* do diretório */timit/test/dr1/fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* na *sa1.wav* do diretório */timit/test/dr1/mwbt0*, onde *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2342	Daubechies	4	2123	Daubechies	6	2054	Daubechies	8	1984	Daubechies	10	1913	Daubechies	12	1854
Daubechies	14	1821	Daubechies	16	1754	Daubechies	18	1745	Daubechies	20	1712	Daubechies	22	1698	Daubechies	24	1643
Daubechies	26	1608	Daubechies	28	1601	Daubechies	30	1578	Daubechies	32	1566	Daubechies	34	1543	Daubechies	36	1498
Daubechies	38	1482	Daubechies	40	1469	Daubechies	42	1470	Daubechies	44	1476	Daubechies	46	1461	Daubechies	48	1440
Daubechies	50	1420	Daubechies	52	1465	Daubechies	54	1461	Daubechies	56	1423	Daubechies	58	1422	Daubechies	60	1410
Daubechies	62	1401	Daubechies	64	1400	Daubechies	66	1396	Daubechies	68	1387	Daubechies	70	1356	Daubechies	72	1343
Daubechies	74	1332	Daubechies	76	1321	Coiflet	6	1725	Coiflet	12	1720	Coiflet	18	1765	Coiflet	24	1745
Coiflet	30	1748	Symmlet	8	2030	Symmlet	16	2011	Beylkin	18	1945	Vaidyanathan	24	561			

[FEMININO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	1426	Daubechies	4	1600	Daubechies	6	1549	Daubechies	8	1533	Daubechies	10	1539	Daubechies	12	1543
Daubechies	14	1583	Daubechies	16	1545	Daubechies	18	1545	Daubechies	20	1534	Daubechies	22	1520	Daubechies	24	1515
Daubechies	26	1503	Daubechies	28	1509	Daubechies	30	1507	Daubechies	32	1507	Daubechies	34	1508	Daubechies	36	1501
Daubechies	38	1480	Daubechies	40	1414	Daubechies	42	1414	Daubechies	44	1413	Daubechies	46	1409	Daubechies	48	1411
Daubechies	50	1401	Daubechies	52	1389	Daubechies	54	1385	Daubechies	56	1384	Daubechies	58	1357	Daubechies	60	1373
Daubechies	62	1361	Daubechies	64	1363	Daubechies	66	1351	Daubechies	68	1344	Daubechies	70	1345	Daubechies	72	1343
Daubechies	74	1344	Daubechies	76	1341	Coiflet	6	1250	Coiflet	12	1245	Coiflet	18	1209	Coiflet	24	1219
Coiflet	30	1223	Symmlet	8	1801	Symmlet	16	1809	Beylkin	18	1240	Vaidyanathan	24	1181			

Tabela 4.5: Com relação à rede neural R_3, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* na *sa1.wav* do diretório */timit/test/dr1/mjsw0*, onde ambos os locutores, isto é, *source* e *targe* são do sexo masculino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/faks0* na *sa1.wav* do diretório */timit/test/dr1/fadac1*, onde ambos os locutores são do sexo feminino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2099	Daubechies	4	2151	Daubechies	6	2147	Daubechies	8	2148	Daubechies	10	2145	Daubechies	12	2133
Daubechies	14	2133	Daubechies	16	2121	Daubechies	18	2122	Daubechies	20	2123	Daubechies	22	2111	Daubechies	24	2109
Daubechies	26	2100	Daubechies	28	2108	Daubechies	30	2098	Daubechies	32	2087	Daubechies	34	2078	Daubechies	36	2065
Daubechies	38	2054	Daubechies	40	2043	Daubechies	42	2023	Daubechies	44	2022	Daubechies	46	2012	Daubechies	48	1999
Daubechies	50	1983	Daubechies	52	1970	Daubechies	54	1971	Daubechies	56	1970	Daubechies	58	1965	Daubechies	60	1959
Daubechies	62	1955	Daubechies	64	1958	Daubechies	66	1952	Daubechies	68	1949	Daubechies	70	1946	Daubechies	72	1941
Daubechies	74	1933	Daubechies	76	1931	Coiflet	6	780	Coiflet	12	710	Coiflet	18	689	Coiflet	24	640
Coiflet	30	623	Symmlet	8	611	Symmlet	16	602	Beylkin	18	575	Vaidyanathan	24	265			

[FEMININO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2		Daubechies	4	2121	Daubechies	6	2112	Daubechies	8	2031	Daubechies	10	2028	Daubechies	12	2029
Daubechies	14	2011	Daubechies	16	2006	Daubechies	18	2001	Daubechies	20	1993	Daubechies	22	1991	Daubechies	24	1893
Daubechies	26	1875	Daubechies	28	1873	Daubechies	30	1869	Daubechies	32	1863	Daubechies	34	1821	Daubechies	36	1830
Daubechies	38	1821	Daubechies	40	1806	Daubechies	42	1873	Daubechies	44	1875	Daubechies	46	1872	Daubechies	48	1872
Daubechies	50	1865	Daubechies	52	1861	Daubechies	54	1854	Daubechies	56	1840	Daubechies	58	1839	Daubechies	60	1832
Daubechies	62	1824	Daubechies	64	1811	Daubechies	66	1812	Daubechies	68	1809	Daubechies	70	1808	Daubechies	72	1803
Daubechies	74	1801	Daubechies	76	1788	Coiflet	6	807	Coiflet	12	794	Coiflet	18	791	Coiflet	24	759
Coiflet	30	732	Symmlet	8	711	Symmlet	16	711	Beylkin	18	614	Vaidyanathan	24	518			

Tabela 4.6: Com relação à rede neural R_3, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* na *sa1.wav* do diretório */timit/test/dr1/fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* na *sa1.wav* do diretório */timit/test/dr1/mwbt0*, onde *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2151	Daubechies	4	511	Daubechies	6	509	Daubechies	8	508	Daubechies	10	507	Daubechies	12	506
Daubechies	14	498	Daubechies	16	492	Daubechies	18	493	Daubechies	20	489	Daubechies	22	490	Daubechies	24	487
Daubechies	26	490	Daubechies	28	484	Daubechies	30	480	Daubechies	32	482	Daubechies	34	482	Daubechies	36	478
Daubechies	38	476	Daubechies	40	474	Daubechies	42	473	Daubechies	44	473	Daubechies	46	472	Daubechies	48	473
Daubechies	50	474	Daubechies	52	469	Daubechies	54	469	Daubechies	56	472	Daubechies	58	470	Daubechies	60	467
Daubechies	62	463	Daubechies	64	463	Daubechies	66	461	Daubechies	68	458	Daubechies	70	451	Daubechies	72	449
Daubechies	74	442	Daubechies	76	441	Coiflet	6	1848	Coiflet	12	1804	Coiflet	18	1760	Coiflet	24	1713
Coiflet	30	1701	Symmlet	8	2260	Symmlet	16	2138	Beylkin	18	2146	Vaidyanathan	24	493			

[FEMININO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	432	Daubechies	4	613	Daubechies	6	612	Daubechies	8	599	Daubechies	10	587	Daubechies	12	584
Daubechies	14	581	Daubechies	16	583	Daubechies	18	582	Daubechies	20	582	Daubechies	22	579	Daubechies	24	571
Daubechies	26	562	Daubechies	28	554	Daubechies	30	539	Daubechies	32	541	Daubechies	34	538	Daubechies	36	537
Daubechies	38	518	Daubechies	40	511	Daubechies	42	501	Daubechies	44	487	Daubechies	46	489	Daubechies	48	484
Daubechies	50	474	Daubechies	52	465	Daubechies	54	453	Daubechies	56	456	Daubechies	58	450	Daubechies	60	439
Daubechies	62	438	Daubechies	64	433	Daubechies	66	431	Daubechies	68	430	Daubechies	70	431	Daubechies	72	428
Daubechies	74	427	Daubechies	76	424	Coiflet	6	422	Coiflet	12	418	Coiflet	18	413	Coiflet	24	403
Coiflet	30	401	Symmlet	8	2113	Symmlet	16	2014	Beylkin	18	1765	Vaidyanathan	24	980			

Tabela 4.7: Com relação à rede neural R_4, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* na *sa1.wav* do diretório */timit/test/dr1/mjsw0*, onde ambos os locutores, isto é, *source* e *targe* são do sexo masculino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/faks0* na *sa1.wav* do diretório */timit/test/dr1/fadac1*, onde ambos os locutores são do sexo feminino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO ⟶ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2111	Daubechies	4	2054	Daubechies	6	2053	Daubechies	8	2048	Daubechies	10	2048	Daubechies	12	2041
Daubechies	14	2039	Daubechies	16	2032	Daubechies	18	2033	Daubechies	20	2036	Daubechies	22	2027	Daubechies	24	2021
Daubechies	26	2022	Daubechies	28	2018	Daubechies	30	2012	Daubechies	32	2011	Daubechies	34	2004	Daubechies	36	1999
Daubechies	38	1995	Daubechies	40	1995	Daubechies	42	1987	Daubechies	44	1984	Daubechies	46	1981	Daubechies	48	1976
Daubechies	50	1975	Daubechies	52	1969	Daubechies	54	1968	Daubechies	56	1962	Daubechies	58	1965	Daubechies	60	1959
Daubechies	62	1953	Daubechies	64	1952	Daubechies	66	1951	Daubechies	68	1949	Daubechies	70	1947	Daubechies	72	1943
Daubechies	74	1945	Daubechies	76	1940	Coiflet	6	789	Coiflet	12	730	Coiflet	18	678	Coiflet	24	612
Coiflet	30	568	Symmlet	8	750	Symmlet	16	725	Beylkin	18	678	Vaidyanathan	24	561			

[FEMININO ⟶ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2341	Daubechies	4	2091	Daubechies	6	2093	Daubechies	8	2089	Daubechies	10	2070	Daubechies	12	2067
Daubechies	14	2060	Daubechies	16	2059	Daubechies	18	2055	Daubechies	20	2049	Daubechies	22	2047	Daubechies	24	2043
Daubechies	26	2045	Daubechies	28	2039	Daubechies	30	2028	Daubechies	32	2026	Daubechies	34	2024	Daubechies	36	2022
Daubechies	38	2018	Daubechies	40	2011	Daubechies	42	2001	Daubechies	44	1998	Daubechies	46	1987	Daubechies	48	1983
Daubechies	50	1976	Daubechies	52	1972	Daubechies	54	1973	Daubechies	56	1969	Daubechies	58	1962	Daubechies	60	1959
Daubechies	62	1957	Daubechies	64	1951	Daubechies	66	1946	Daubechies	68	1945	Daubechies	70	1939	Daubechies	72	1935
Daubechies	74	1931	Daubechies	76	1926	Coiflet	6	810	Coiflet	12	789	Coiflet	18	741	Coiflet	24	699
Coiflet	30	667	Symmlet	8	710	Symmlet	16	648	Beylkin	18	515	Vaidyanathan	24	491			

Tabela 4.8: Com relação à rede neural R_4, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* na *sa1.wav* do diretório */timit/test/dr1/fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* na *sa1.wav* do diretório */timit/test/dr1/mwbt0*, onde *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	1951	Daubechies	4	1501	Daubechies	6	1505	Daubechies	8	1500	Daubechies	10	1470	Daubechies	12	1431
Daubechies	14	1410	Daubechies	16	1390	Daubechies	18	1379	Daubechies	20	1379	Daubechies	22	1340	Daubechies	24	1349
Daubechies	26	1323	Daubechies	28	1319	Daubechies	30	1317	Daubechies	32	1299	Daubechies	34	1288	Daubechies	36	1279
Daubechies	38	1267	Daubechies	40	1259	Daubechies	42	1240	Daubechies	44	1236	Daubechies	46	1231	Daubechies	48	1220
Daubechies	50	1210	Daubechies	52	1205	Daubechies	54	1191	Daubechies	56	1180	Daubechies	58	1172	Daubechies	60	1160
Daubechies	62	1151	Daubechies	64	1140	Daubechies	66	1130	Daubechies	68	1120	Daubechies	70	1121	Daubechies	72	1109
Daubechies	74	1090	Daubechies	76	1080	Coiflet	6	2145	Coiflet	12	2130	Coiflet	18	2090	Coiflet	24	1945
Coiflet	30	1898	Symmlet	8	2142	Symmlet	16	2048	Beylkin	18	1926	Vaidyanathan	24	1464			

[FEMININO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	1726	Daubechies	4	1700	Daubechies	6	1639	Daubechies	8	1590	Daubechies	10	1589	Daubechies	12	1568
Daubechies	14	1543	Daubechies	16	1535	Daubechies	18	1541	Daubechies	20	1534	Daubechies	22	1520	Daubechies	24	1515
Daubechies	26	1511	Daubechies	28	1519	Daubechies	30	1501	Daubechies	32	1477	Daubechies	34	1468	Daubechies	36	1451
Daubechies	38	1427	Daubechies	40	1419	Daubechies	42	1414	Daubechies	44	1418	Daubechies	46	1411	Daubechies	48	1410
Daubechies	50	1401	Daubechies	52	1389	Daubechies	54	1382	Daubechies	56	1378	Daubechies	58	1347	Daubechies	60	1350
Daubechies	62	1341	Daubechies	64	1330	Daubechies	66	1320	Daubechies	68	1319	Daubechies	70	1311	Daubechies	72	1308
Daubechies	74	1301	Daubechies	76	1298	Coiflet	6	2050	Coiflet	12	1980	Coiflet	18	1739	Coiflet	24	1539
Coiflet	30	1421	Symmlet	8	2410	Symmlet	16	1839	Beylkin	18	1441	Vaidyanathan	24	982			

Tabela 4.9: Com relação à rede neural R_5, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mdab0* na *sa1.wav* do diretório */timit/test/dr1/mjsw0*, onde ambos os locutores, isto é, *source* e *targe* são do sexo masculino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/faks0* na *sa1.wav* do diretório */timit/test/dr1/fadac1*, onde ambos os locutores são do sexo feminino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO $\longrightarrow$ FEMININO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2113	Daubechies	4	2100	Daubechies	6	2079	Daubechies	8	2068	Daubechies	10	2067	Daubechies	12	2058
Daubechies	14	2057	Daubechies	16	2045	Daubechies	18	2038	Daubechies	20	2035	Daubechies	22	2027	Daubechies	24	2021
Daubechies	26	2019	Daubechies	28	2018	Daubechies	30	2011	Daubechies	32	2001	Daubechies	34	1998	Daubechies	36	1991
Daubechies	38	1985	Daubechies	40	1986	Daubechies	42	1976	Daubechies	44	1973	Daubechies	46	1963	Daubechies	48	1960
Daubechies	50	1951	Daubechies	52	1953	Daubechies	54	1951	Daubechies	56	1943	Daubechies	58	1941	Daubechies	60	1938
Daubechies	62	1933	Daubechies	64	1921	Daubechies	66	1919	Daubechies	68	1905	Daubechies	70	1904	Daubechies	72	1897
Daubechies	74	1889	Daubechies	76	1879	Coiflet	6	1700	Coiflet	12	1520	Coiflet	18	1389	Coiflet	24	1280
Coiflet	30	1126	Symmlet	8	1658	Symmlet	16	1421	Beylkin	18	1225	Vaidyanathan	24	966			

[FEMININO $\longrightarrow$ MASCULINO]:

DTWT	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*	*DTWT*	*SP*	*QM*
Haar	2	2321	Daubechies	4	2211	Daubechies	6	2203	Daubechies	8	2892	Daubechies	10	2890	Daubechies	12	2879
Daubechies	14	2878	Daubechies	16	2857	Daubechies	18	2845	Daubechies	20	2835	Daubechies	22	2834	Daubechies	24	2820
Daubechies	26	2815	Daubechies	28	2800	Daubechies	30	2789	Daubechies	32	2770	Daubechies	34	2860	Daubechies	36	2850
Daubechies	38	2841	Daubechies	40	2836	Daubechies	42	2831	Daubechies	44	2825	Daubechies	46	2805	Daubechies	48	2790
Daubechies	50	2765	Daubechies	52	2761	Daubechies	54	2754	Daubechies	56	2740	Daubechies	58	2839	Daubechies	60	2821
Daubechies	62	2804	Daubechies	64	2790	Daubechies	66	2780	Daubechies	68	2740	Daubechies	70	2736	Daubechies	72	2731
Daubechies	74	2711	Daubechies	76	2708	Coiflet	6	2370	Coiflet	12	2190	Coiflet	18	1891	Coiflet	24	1703
Coiflet	30	1604	Symmlet	8	1810	Symmlet	16	1606	Beylkin	18	1465	Vaidyanathan	24	1110			

Tabela 4.10: Com relação à rede neural R_5, na parte superior da tabela, observa-se a convergência do procedimento de VM para converter a sentença *sa1.wav* do diretório */timit/test/dr1/mwbt0* na *sa1.wav* do diretório */timit/test/dr1/fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo se demonstra para a conversão da sentença *sa1.wav* do diretório */timit/test/dr1/fjem0* na *sa1.wav* do diretório */timit/test/dr1/mwbt0*, onde *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Legenda: QM: quantidade máxima de iterações para obter um erro de $< 10^{-3}$ para cada caso; SP: suporte da DTWT utilizada. Fonte: Elaborado pelo autor

[MASCULINO ⟶ MASCULINO]:

DTWT	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*
Haar	2	0.812	Daubechies	4	0.810	Daubechies	6	0.810	Daubechies	8	0.816	Daubechies	10	0.806	Daubechies	12	0.800
Daubechies	14	0.800	Daubechies	16	0.821	Daubechies	18	0.800	Daubechies	20	0.802	Daubechies	22	0.820	Daubechies	24	0.800
Daubechies	26	0.798	Daubechies	28	0.799	Daubechies	30	0.810	Daubechies	32	0.798	Daubechies	34	0.797	Daubechies	36	0.797
Daubechies	38	0.795	Daubechies	40	0.784	Daubechies	42	0.796	Daubechies	44	0.791	Daubechies	46	0.792	Daubechies	48	0.791
Daubechies	50	0.792	Daubechies	52	0.784	Daubechies	54	0.790	Daubechies	56	0.793	Daubechies	58	0.810	Daubechies	60	0.789
Daubechies	62	0.793	Daubechies	64	0.784	Daubechies	66	0.790	Daubechies	68	0.794	Daubechies	70	0.774	Daubechies	72	0.789
Daubechies	74	0.794	Daubechies	76	0.790	Coiflet	6	0.818	Coiflet	12	0.803	Coiflet	18	0.800	Coiflet	24	0.810
Coiflet	30	0.796	Symmlet	8	0.800	Symmlet	16	0.788	Beylkin	18	0.795	Vaidyanathan	24	0.714			

[FEMININO ⟶ FEMININO]:

DTWT	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*
Haar	2	0.817	Daubechies	4	0.794	Daubechies	6	0.796	Daubechies	8	0.787	Daubechies	10	0.786	Daubechies	12	0.795
Daubechies	14	0.811	Daubechies	16	0.787	Daubechies	18	0.794	Daubechies	20	0.788	Daubechies	22	0.798	Daubechies	24	0.785
Daubechies	26	0.810	Daubechies	28	0.788	Daubechies	30	0.796	Daubechies	32	0.780	Daubechies	34	0.801	Daubechies	36	0.786
Daubechies	38	0.789	Daubechies	40	0.783	Daubechies	42	0.789	Daubechies	44	0.780	Daubechies	46	0.798	Daubechies	48	0.795
Daubechies	50	0.811	Daubechies	52	0.794	Daubechies	54	0.810	Daubechies	56	0.780	Daubechies	58	0.798	Daubechies	60	0.791
Daubechies	62	0.798	Daubechies	64	0.789	Daubechies	66	0.799	Daubechies	68	0.798	Daubechies	70	0.799	Daubechies	72	0.787
Daubechies	74	0.798	Daubechies	76	0.789	Coiflet	6	0.784	Coiflet	12	0.794	Coiflet	18	0.785	Coiflet	24	0.799
Coiflet	30	0.772	Symmlet	8	0.794	Symmlet	16	0.787	Beylkin	18	0.796	Vaidyanathan	24	0.782			

Tabela 4.11: Registro das distâncias Euclidianas normalizadas, em relação ao tamanho dos sinais, entre a sentença *sa1.wav* do diretório */timit/ test/ dr1/ mdab0* e a *sa1.wav* do diretório */timit/ test/ dr1/ mjsw0*, sendo ambos os locutores do sexo masculino. Na parte inferior, o mesmo se registra para as sentenças *sa1.wav* do diretório */timit/ test/ dr1/ faks0* e *sa1.wav* do diretório */timit/ test/ dr1/ fadac1*, respectivamente, onde o *source speaker* e o *target speaker* são do sexo feminino. Legenda: DTWT: base *wavelet*; SP: suporte correspondente dos filtros, DE: distância Euclidiana entre os sinais original e convertido. Fonte: Elaborado pelo autor

[MASCULINO ⟶ FEMININO]:

DTWT	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*
Haar	2	0.906	Daubechies	4	0.902	Daubechies	6	0.900	Daubechies	8	0.893	Daubechies	10	0.893	Daubechies	12	0.893
Daubechies	14	0.878	Daubechies	16	0.884	Daubechies	18	0.801	Daubechies	20	0.895	Daubechies	22	0.894	Daubechies	24	0.891
Daubechies	26	0.874	Daubechies	28	0.886	Daubechies	30	0.888	Daubechies	32	0.887	Daubechies	34	0.889	Daubechies	36	0.894
Daubechies	38	0.882	Daubechies	40	0.884	Daubechies	42	0.886	Daubechies	44	0.888	Daubechies	46	0.886	Daubechies	48	0.888
Daubechies	50	0.888	Daubechies	52	0.888	Daubechies	54	0.889	Daubechies	56	0.886	Daubechies	58	0.886	Daubechies	60	0.886
Daubechies	62	0.887	Daubechies	64	0.888	Daubechies	66	0.886	Daubechies	68	0.884	Daubechies	70	0.888	Daubechies	72	0.886
Daubechies	74	0.889	Daubechies	76	0.886	Coiflet	6	0.836	Coiflet	12	0.834	Coiflet	18	0.836	Coiflet	24	0.834
Coiflet	30	0.831	Symmlet	8	0.786	Symmlet	16	0.789	Beylkin	18	0.782	Vaidyanathan	24	0.701			

[FEMININO ⟶ MASCULINO]:

DTWT	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*	*DTWT*	*SP*	*DE*
Haar	2	0.800	Daubechies	4	0.801	Daubechies	6	0.813	Daubechies	8	0.812	Daubechies	10	0.810	Daubechies	12	0.810
Daubechies	14	0.810	Daubechies	16	0.813	Daubechies	18	0.805	Daubechies	20	0.803	Daubechies	22	0.805	Daubechies	24	0.806
Daubechies	26	0.812	Daubechies	28	0.812	Daubechies	30	0.807	Daubechies	32	0.809	Daubechies	34	0.809	Daubechies	36	0.809
Daubechies	38	0.801	Daubechies	40	0.813	Daubechies	42	0.818	Daubechies	44	0.802	Daubechies	46	0.801	Daubechies	48	0.802
Daubechies	50	0.813	Daubechies	52	0.811	Daubechies	54	0.801	Daubechies	56	0.801	Daubechies	58	0.801	Daubechies	60	0.809
Daubechies	62	0.815	Daubechies	64	0.889	Daubechies	66	0.803	Daubechies	68	0.801	Daubechies	70	0.802	Daubechies	72	0.808
Daubechies	74	0.815	Daubechies	76	0.890	Coiflet	6	0.883	Coiflet	12	0.842	Coiflet	18	0.805	Coiflet	24	07890
Coiflet	30	0.778	Symmlet	8	0.840	Symmlet	16	0.830	Beylkin	18	0.779	Vaidyanathan	24	0.775			

Tabela 4.12: Registro das distâncias Euclidianas normalizadas, em relação ao tamanho dos sinais, entre a sentença *sa1.wav* do diretório */timit/ test/ dr1/ mwbt0* e a *sa1.wav* do diretório */timit/ test/ dr1/ fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Na parte inferior, o mesmo registra-se para as sentenças *sa1.wav* do diretório */timit/ test/ dr1/ fjem0* e *sa1.wav* do diretório */timit/ test /dr1/ mwbt0*, respectivamente, ou seja, o *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino; Legenda: SP: suporte correspondente dos filtros, DE: distância Euclidiana entre os sinais original e convertido. Fonte: Elaborado pelo autor

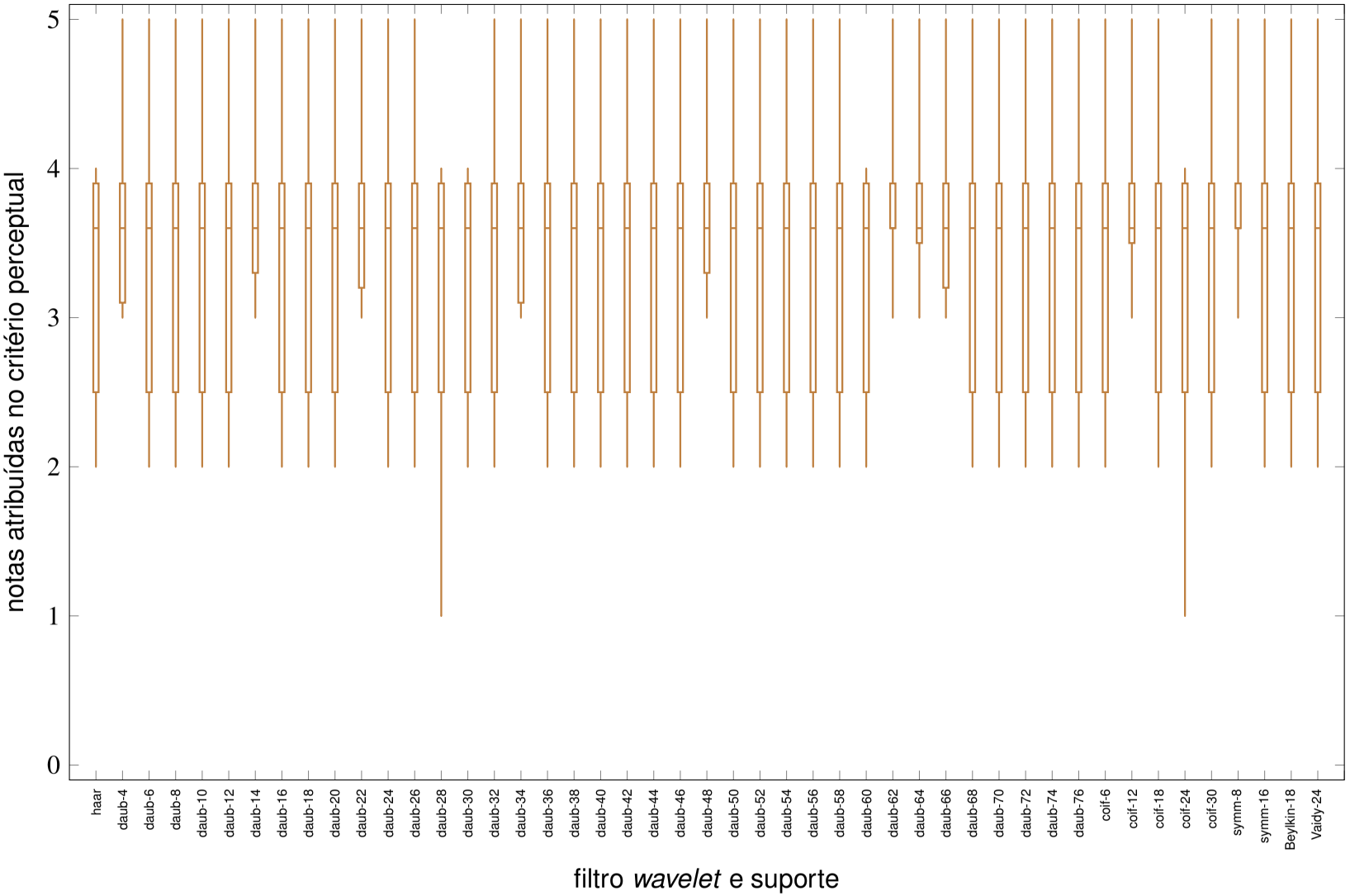


Figura 4.1: *Box-plots* das notas atribuídas pelos ouvintes para a qualidade da conversão das vozes quando da conversão entre a sentença *sa1.wav* do diretório */timit/ test/ dr1/ mwbt0* e a *sa1.wav* do diretório */timit/ test/ dr1/ fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* também é do sexo masculino. Fonte: Elaborado pelo autor

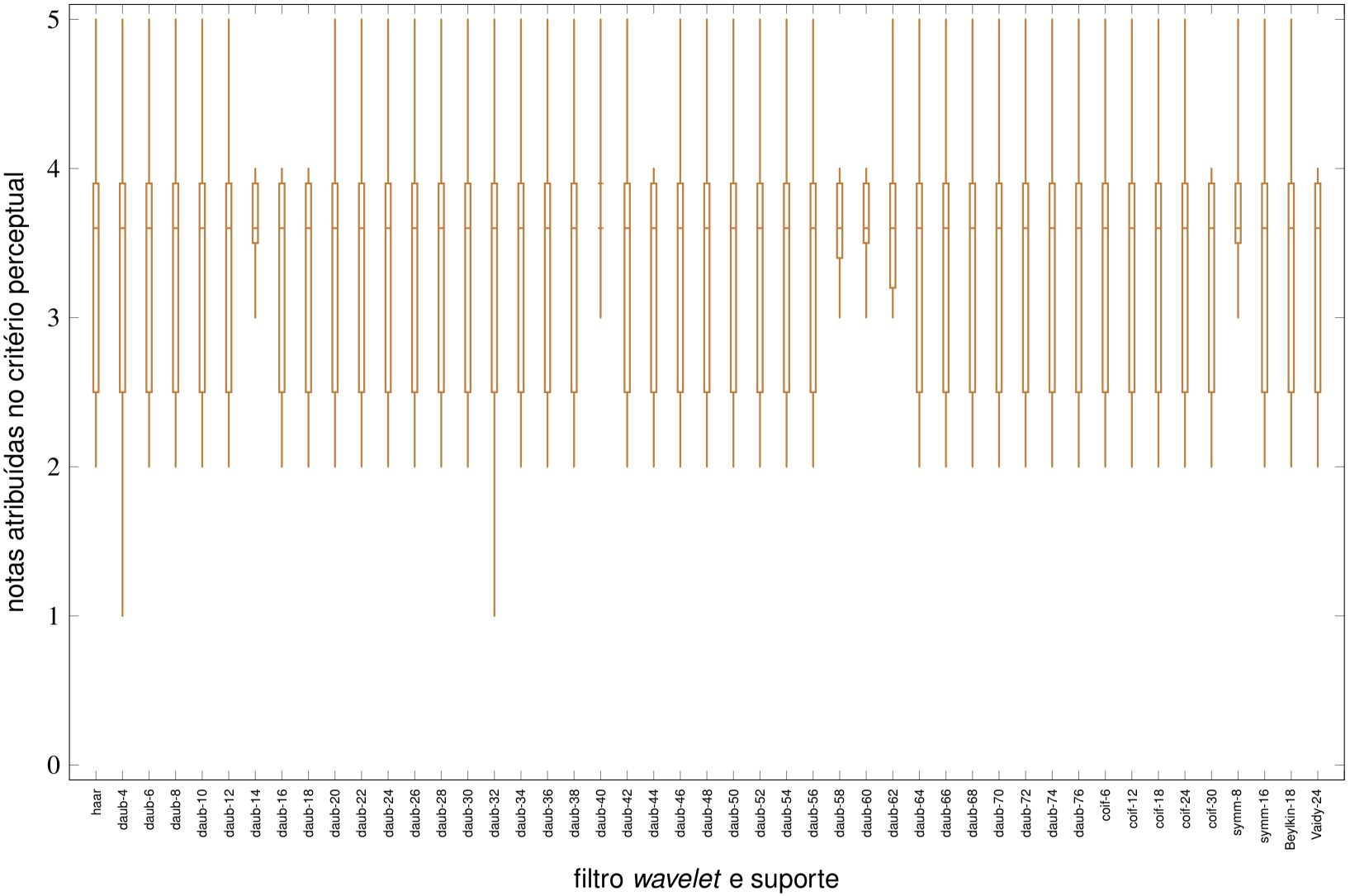


Figura 4.2: *Box-plots* das notas atribuídas pelos ouvintes para a qualidade da conversão das vozes quando da conversão entre a sentença *sa1.wav* do diretório */timit/ test/ dr1/ fjem0* e *sa1.wav* do diretório */timit/ test/ dr1/ mwbt0*, em que o *source speaker* é do sexo feminino e o *target speaker* também é do sexo feminino. Fonte: Elaborado pelo autor

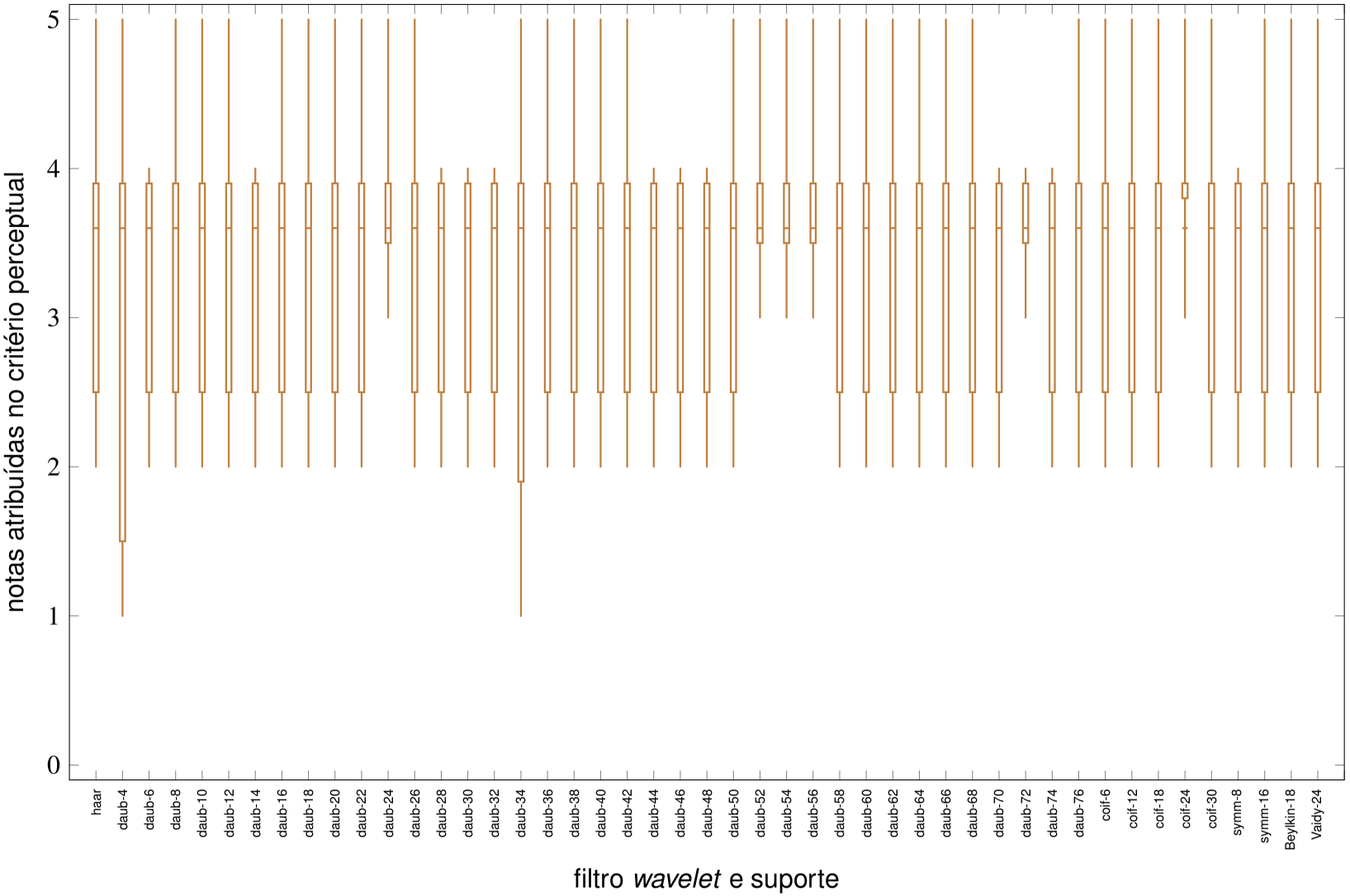


Figura 4.3: *Box-plots* das notas atribuídas pelos ouvintes para a qualidade da conversão entre a sentença *sa1.wav* do diretório */timit/ test/ dr1/ mwbt0* e a *sa1.wav* do diretório */timit/ test/ dr1/ fjem0*, onde o *source speaker* é do sexo masculino e o *target speaker* é do sexo feminino. Fonte: Elaborado pelo autor

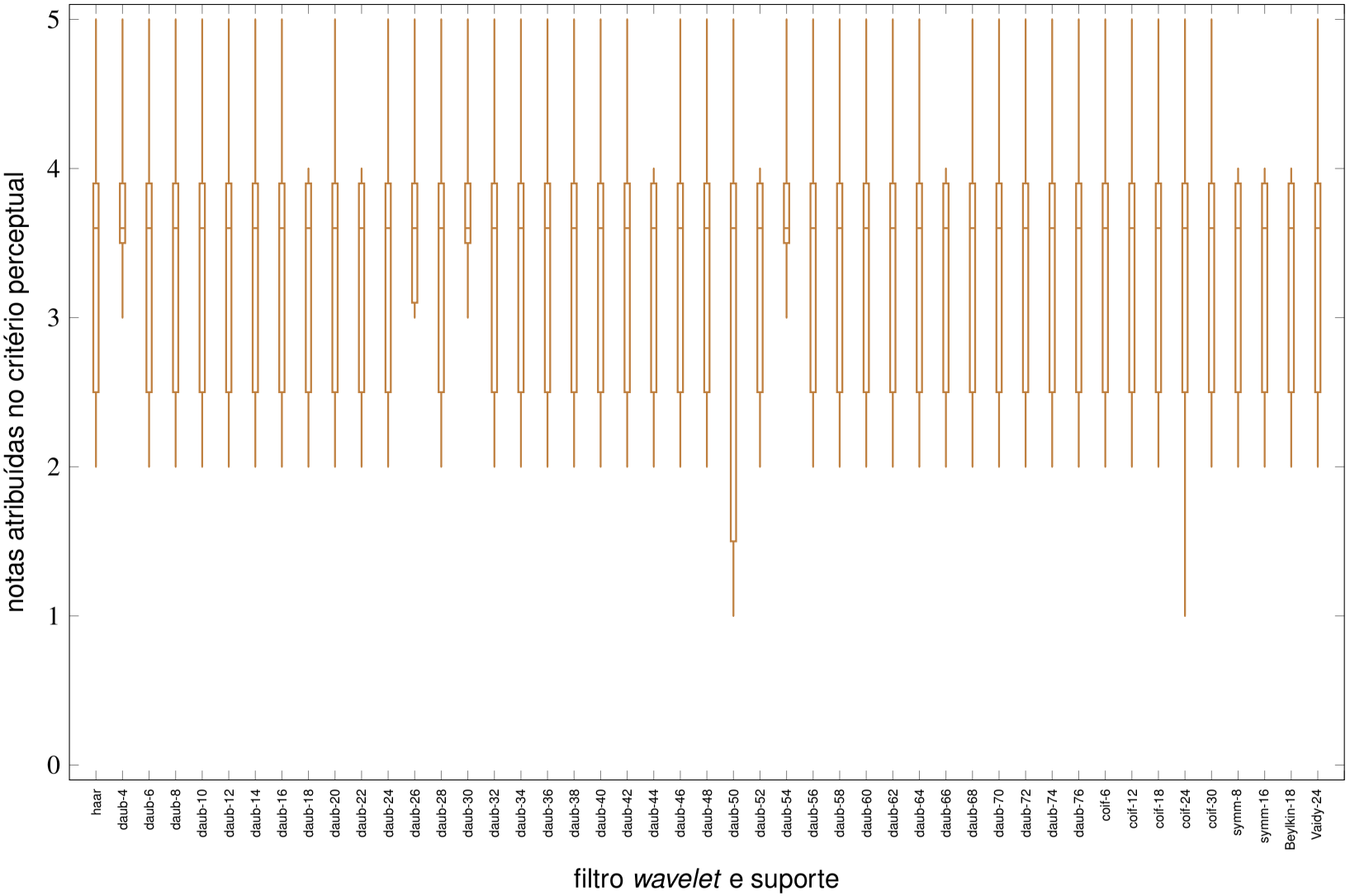


Figura 4.4: *Box-plots* das notas atribuídas pelos ouvintes para a qualidade da conversão entre as sentenças *sa1.wav* do diretório */timit/ test/ dr1/ fjem0* e *sa1.wav* do diretório */timit/ test/ dr1/ mwbt0*, em que o *source speaker* é do sexo feminino e o *target speaker* é do sexo masculino. Fonte: Elaborado pelo autor

CONCLUSÕES

Voice Morphing é um tema amplamente debatido e estudado na atualidade. Neste contexto, esta pesquisa se concentra em explorar uma abordagem alternativa: a conversão de voz com base na DTWT, combinada com redes neurais artificiais rasas, que oferecem a vantagem de produzir modelos mais interpretáveis.

Outro aspecto diferenciador reside na utilização de validações tanto qualitativas quanto quantitativas. A mensuração da distância Euclidiana entre os sinais transformados e dos sinais originais não só enriquece a análise subjetiva, mas também mitiga eventuais vieses por parte dos avaliadores.

Com base nos resultados apresentados, é evidente que a transformação usando os filtros *wavelet* de Vaidyanathan implicaram um desempenho superior no que diz respeito ao número de iterações necessárias para alcançar um erro inferior a 10^{-3}, o que também foi observado com o filtro Belkyn de suporte 18. Por outro lado, a transformação usando os filtros Daubechies de variados suportes acompanha esse desempenho quando se lida com suportes maiores da DTWT. No entanto, mesmo que o número de iterações seja maior, a qualidade do sinal transformado permanece comparável em uma nota média global em torno de 4.1. Além disso, é importante notar que o aumento na quantidade de neurônios nas camadas intermediárias não necessariamente resulta em uma melhoria na quantidade de iterações necessárias para atingir o erro mínimo desejado.

Com base nessas observações, destaca-se a importância de selecionar a abordagem adequada, levando em consideração as características específicas do problema e os objetivos desejados. A comparação entre as transformações, juntamente

com as implicações da complexidade da rede neural, proporciona *insights* valiosos para a escolha e otimização das técnicas de transformação de sinais. De modo geral, acredita-se, também, que os resultados aqui obtidos são compatíveis com os demais existentes na literatura e que foram revisados no Capítulo 2.

Em trabalhos subsequentes, ter-se-á oportunidade de utilizar os resultados deste estudo como uma base para ampliar a compreensão e a aplicação da conversão de voz. É viável o refinamento e o aprimoramento das técnicas propostas, explorando diversas combinações de redes neurais e otimizando a seleção dos filtros *wavelet*, o que pode resultar em modelos de conversão de voz ainda mais eficazes e interpretáveis. Finalmente, existe a perspectiva de ampliar o alcance deste trabalho para diferentes contextos, como a tradução de idiomas estrangeiros e aprimoramentos em sinais com ruídos ou danos.

REFERÊNCIAS

[1] ALLEN B. DOWNEY. *Think DSP: Digital Signal Processing in Python*. O'Reilly Media, Inc., 1st edition, 2016.

[2] BERRAK SISMAN, JUNICHI YAMAGISHI, SIMON KING, and HAIZHOU LI. An overview of voice conversion and its challenges: From statistical modeling to deep learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:132–157, 2021.

[3] HUI YE and S. YOUNG. High quality voice morphing. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–9, 2004.

[4] RAOUDHA RAHMENI, ANIS BEN AICHA, and YASSINE BEN AYED. On the contribution of the voice texture for speech spoofing detection. In *2019 19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 501–505, 2019.

[5] YONG WANG and ZHUOYI SU. Detection of voice transformation spoofing based on dense convolutional network. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2587–2591, 2019.

[6] LUCIMAR SASSO VIEIRA. *Morphlet: uma nova família de transformadas wavelet aplicadas ao processo de conversão de voz*. PhD thesis, Universidade de São Paulo, Instituto de Física de São Carlos, São Carlos, 2012.

[7] MINGYANG ZHANG, YI ZHOU, LI ZHAO, and HAIZHOU LI. Transfer learning from speech synthesis to voice conversion with non-parallel training data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1290–1302, 2021.

[8] CHUNHUI DENG, YING CHEN, and HUIFANG DENG. One-shot voice conversion algorithm based on representations separation. *IEEE Access*, 8:196578–196586, 2020.

[9] HONGQIANG DU, XIAOHAI TIAN, LEI XIE, and HAIZHOU LI. Factorized wavenet for voice conversion with limited data. *Speech Communication*, 130:45–54, 2021.

[10] MOHAMADREZA JAFARYANI, HAMID SHEIKHZADEH, and VAHID POURAHMADI. Parallel voice conversion with limited training data using stochastic variational deep kernel learning. *Engineering Applications of Artificial Intelligence*, 115:105279, 2022.

[11] YOUNG-SUN YUN, JINMAN JUNG, SEONGBAE EUN, SHIN CHA, and SUN-SUP SO. Voice conversion of synthesized speeches using deep neural networks. In *2019 International Conference on Green and Human Information Technology (ICGHIT)*, pages 93–96, 2019.

[12] JIAN ZHOU, YUTING HU, HAILUN LIAN, HUABIN WANG, LIANG TAO, and HON KEUNG KWAN. Multimodal voice conversion under adverse environment using a deep convolutional neural network. *IEEE Access*, 7:170878–170887, 2019.

[13] MICHAEL GIAN V. GONZALES, CRISRON RUDOLF G. LUCAS, MICHAEL GRINGO ANGELO R. BAYONA, and FRANZ A. DE LEON. Voice conversion of philippine spoken languages using deep neural networks. In *2020 IEEE 8th Conference on Systems, Process and Control (ICSPC)*, pages 118–121, 2020.

[14] ETSURO URABE, RIN HIRAKAWA, HIDEAKI KAWANO, KENICHI NAKASHI, and YOSHIHISA NAKATOH. Electrolarynx system using voice conversion based on wavernn. In *2020 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–2, 2020.

[15] PATRICK LUMBAN TOBING, YI-CHIAO WU, TOMOKI HAYASHI, KAZUHIRO KOBAYASHI, and TOMOKI TODA. Voice conversion with cyclernn-based spectral mapping and finely tuned wavenet vocoder. *IEEE Access*, 7:171114–171125, 2019.

[16] WEI-ZHONG ZHENG, JI-YAN HAN, CHEN-KAI LEE, YU-YI LIN, SHU-HAN CHANG, and YING-HUI LAI. Phonetic posteriorgram-based voice conversion system to improve speech intelligibility of dysarthric patients. *Computer Methods and Programs in Biomedicine*, 215:106602, 2022.

[17] AL-WALED AL-DULAIMI, TODD K. MOON, and JACOB H. GUNTHER. Voice transformation using two-level dynamic warping. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 143–147, 2019.

[18] FENG-LONG XIE, YAO QIAN, FRANK K. SOONG, and HAIFENG LI. Pitch transformation in neural network based voice conversion. In *The 9th International Symposium on Chinese Spoken Language Processing*, pages 197–200, 2014.

[19] CHRISTIAN T. HERBST. A review of singing voice subsystem interactions—toward an extended physiological model of “support”. *Journal of Voice*, 31(2):249.e13–249.e19, 2017.

[20] JONATHAN HARRINGTON and STEVE CASSIDY. *Techniques in Speech Acoustics*. Kluwer, The Netherlands, 1 edition, 1999.

[21] RODRIGO CAPOBIANCO GUIDO. Zcr-aided neurocomputing: A study with applications. *Knowledge-Based Systems*, 105:248–269, 2016.

[22] RODRIGO CAPOBIANCO GUIDO. A tutorial on signal energy and its applications. *Neurocomputing*, 179:264–282, 2016.

[23] RODRIGO CAPOBIANCO GUIDO. Wavelets behind the scenes: Practical aspects, insights, and perspectives. *Physics Reports*, 985:1–23, 2022. Wavelets behind the scenes: Practical aspects, insights, and perspectives.

[24] RODRIGO CAPOBIANCO GUIDO, FERNANDO PEDROSO, ANDRÉ FURLAN, RODRIGO COLNAGO CONTRERAS, LUIZ GUSTAVO CAOBIANCO, and JOGI SUDA NETO. Cwt × dwt × dtwt × sdtwt: Clarifying terminologies and roles of different types of wavelet transforms. *International Journal of Wavelets, Multiresolution and Information Processing*, 18(06):2030001, 2020.

[25] RODRIGO CAPOBIANCO GUIDO. Effectively interpreting discrete wavelet transformed signals [lecture notes]. *IEEE Signal Processing Magazine*, 34(3):89–100, 2017.

[26] RODRIGO CAPOBIANCO GUIDO. Practical and useful tips on discrete wavelet transforms [sp tips tricks]. *IEEE Signal Processing Magazine*, 32(3):162–166, 2015.

[27] SIMON HAYKIN. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999.

[28] J. S. GAROFOLO, L. F. LAMEL, W. M. FISHER, J. G. FISCUS, D. S. PALLETT, and N. L. DAHLGREN. Darpa timit acoustic phonetic continuous speech corpus cdrom, 1993.