

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELETRÔNICA E DE
COMPUTAÇÃO

Algoritmos para redução de ruído em sinais de áudio

Autor:

Vinicius R. V. G. da Silva

Orientador:

Prof. Mariane Rembold Petraglia, Ph.D.

Examinador:

Prof. José Gabriel Rodríguez Carneiro Gomes, Ph.D.

Examinador:

Prof. Julio César Boscher Torres, D.Sc.

DEL
Dezembro de 2007

AGRADECIMENTO

Gostaria de agradecer à minha família e aos meus amigos pelo apoio e incentivo, e a todos que contribuíram na conclusão da minha formação como engenheiro, e sem os quais a realização deste projeto não teria sido possível.

À minha Orientadora de Projeto de Fim de Curso, Mariane Petraglia, pela sua paciência, motivação e constante apoio.

Aos meus pais e minhas irmãs pelo constante apoio e confiança que sempre depositaram em mim.

RESUMO

Este trabalho tem por objetivo viabilizar o processamento e restauração de sinais de áudio que sofreram distorções devido à introdução de ruído, utilizando para tanto, algoritmos implementados no software Matlab.

Diversos algoritmos foram propostos ao longo dos anos para eliminação de ruídos em sinais de áudio. Porém, os resultados desses introduziam distorções na voz, e não eliminavam completamente o ruído, criando o “musical noise” (ruído de fundo).

Esse projeto consiste na implementação de quatro algoritmos para minimizar os efeitos do ruído, bem como diminuir as distorções no sinal de voz provocadas pela restauração deste sinal após a subtração do ruído. Esses algoritmos são: Subtração Espectral, Filtro de Wiener, Subspace Tracking e Subspace Tracking com transformada Wavelet. A Subtração Espectral consiste em suprimir o ruído a partir de uma estimativa das estatísticas do ruído em períodos de silêncio. O Filtro de Wiener minimiza o erro quadrático no processo de filtragem inversa para suavização do ruído. O “Subspace Tracking” consiste em reduzir o ruído através da análise dos autovetores e autovalores da matriz de covariância do sinal de voz.

A Transformada Wavelet consiste na decomposição do sinal de voz no domínio da frequência, com aplicação sucessiva de filtros passa-baixas e passa-altas. A transformada será combinada com o algoritmo “Subspace Tracking”, sendo este aplicado nas diversas componentes de frequência do sinal.

Os resultados obtidos com cada algoritmo serão visualizados graficamente, e serão comparados aos resultados obtidos com os demais, para diferentes tipos de ruídos.

PALAVRAS-CHAVE

Redução do ruído

Processamento de áudio

Transformada Wavelet

Inteligibilidade da voz

Qualidade da voz

SUMÁRIO

1 INTRODUÇÃO.....	1
1.1 MOTIVAÇÃO.....	1
1.2 DESCRIÇÃO DO PROJETO.....	1
2 FUNDAMENTOS TEÓRICOS.....	2
2.1 PROCESSAMENTO DE ÁUDIO.....	2
2.2 ALGORITMOS.....	2
2.2.1 Subtração Espectral.....	2
2.2.2 Filtro de Wiener.....	5
2.2.3 Subspace Tracking.....	6
2.2.4 Subspace Tracking com a Transformada Wavelet.....	8
2.2.4.1 Transformada Wavelet.....	8
3 RESULTADOS.....	12
Nesta capítulo, iremos discorrer acerca da performance de cada algoritmo em relação ao sinal de voz contaminado por diferentes tipos de ruído.....	
12	12
3.1 Ruído de Avião.....	12
3.1.1 Subtração Espectral.....	12
3.1.2 Filtro de Wiener.....	13
3.1.3 Subspace Tracking.....	15
3.1.4 Subspace Tracking com Transformada Wavelet.....	17
3.1.5 Comparação entre os Algoritmos.....	19
Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído de avião. Destacamos a presença do “musical noise” nos sinais resultantes dos métodos Filtro de Wiener e Subtração Espectral, e de componentes de alta frequência, antes não existente, nos sinais obtidos com o Subspace Tracking e, em menor intensidade, com o Subspace Tracking com a transformada wavelet. Do ponto de vista subjetivo, a maior parte dos ouvintes indicou o Filtro de Wiener como o algoritmo que melhor eliminou o ruído.....	
20	20
3.2 Ruído de Carro.....	20
3.2.1 Subtração Espectral.....	21
3.2.2 Filtro de Wiener.....	22
3.2.3 Subspace Tracking.....	24
3.2.4 Subspace Tracking com Transformada Wavelet.....	26
3.2.5 Comparação entre os Algoritmos.....	28
Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído de carro. Destacamos a presença do “musical noise” nos sinais resultantes do Filtro de Wiener e da Subtração Espectral, e o surgimento de componentes em alta frequência no Subspace Tracking e no mesmo com a transformada wavelet. Do ponto de vista subjetivo, os ouvintes não distinguiram entre o Filtro de Wiener e a Subtração espectral. No caso do Subspace Tracking ficou claro ao ouvir o sinal de voz que a utilização da transformada wavelet melhorou consideravelmente a qualidade do sinal resultante em relação ao do algoritmo original...29	
29	29
3.3 Ruído de Rua.....	29
3.3.1 Subtração Espectral.....	30
3.3.2 Filtro de Wiener.....	31
3.3.3 Subspace Tracking.....	33
3.3.4 Subspace Tracking com Transformada Wavelet.....	35
3.3.4 Comparação entre os Algoritmos.....	37

Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído, pois se observa a existência de vozes paralelas ao sinal de voz de interesse, que não são eliminadas pelo algoritmo. Ainda ocorre a introdução de componentes em alta frequência, antes não existentes, no sinal resultante do algoritmo subspace tracking e no mesmo com a transformada wavelet.....38

3.4 Ruído Branco.....38

3.4.1 Subtração Espectral.....39

3.4.2 Filtro de Wiener.....40

3.4.3 Subspace Tracking.....42

3.4.4 Subspace Tracking com Transformada Wavelet.....44

3.4.5 Comparação entre os Algoritmos.....46

Destacamos a presença do “musical noise” no Filtro de Wiener e Subtração Espectral, e a perda de componentes em alta frequência no Sbspace Tracking e no mesmo com a transformada wavelet. Do ponto de vista do aspecto subjetivo, o Filtro de Wiener foi o que obteve o melhor resultado na redução do ruído branco. Ficou claro, mais uma vez, que a utilização da transformada wavelet melhorou consideravelmente o resultado do algoritmo Subspace Tracking.....47

ÍNDICE DE FIGURAS

2.1	Diagrama de Blocos.....	5
2.2	Transformada Wavelet.....	9
2.3	Wavelets de Daubechies.....	10
2.4	Tranformada de Wavelets de Daubechie D4 e D16.....	10

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

Aplicações de processamento e reconhecimento do sinal de voz estão muito difundidas nos dias de hoje. Entre elas podemos citar os telefones celulares e sistemas de navegação para automóveis. Uma das dificuldades que essas aplicações encontram é a presença do ruído no ambiente, como em ruas, carros e escritórios. A qualidade da voz é seriamente distorcida e degradada nesses lugares, porque o sinal de voz é corrompido pelo ruído do ambiente, o que resulta numa baixa taxa de inteligibilidade.

Esse projeto visa a implementação de algoritmos supressores de ruído, com o objetivo de reduzir ou eliminar a influência daquele no sinal de voz.

1.2 DESCRIÇÃO DO PROJETO

O objetivo deste projeto é poder restaurar os sinais de voz que sofreram algum tipo de degradação ou distorção. Para realizar esta restauração foram implementados quatro algoritmos para eliminar o fenômeno de degradação e distorção da voz. Estes métodos iterativos são implementados e testados no software Matlab.

Esses algoritmos são: Subtração Espectral, Filtro de Wiener, Subspace Tracking e Subspace Tracking com transformada Wavelet. A Subtração Espectral consiste em subtrair uma boa parte do ruído a partir da estimativa da densidade espectral de potência do ruído, obtida em períodos de silêncio. O Filtro de Wiener suaviza o ruído através da minimização do erro quadrático no processo de filtragem inversa. O algoritmo “Subspace Tracking” reduz o ruído através da decomposição do sinal ruidoso em subespaços associados ao sinal de voz e ao ruído, baseada na análise dos autovetores e autovalores da matriz de covariância do sinal. O algoritmo “Subspace Tracking” pode ser melhorado fazendo-se a decomposição do sinal ruidoso em diferentes faixas de frequência, através da utilização da Transformada Wavelet, e aplicando-se este algoritmo nas diversas componentes de frequência. Com a aplicação da transformada Wavelet podemos extrair informações desejadas com boa localização tanto no domínio da frequência como no tempo.

O sinal de voz utilizado neste trabalho é amostrado a uma taxa de 16 KHz e contaminado com ruído de quatro tipos de ruído : carro, rua, avião e ruído branco. Os resultados serão dispostos graficamente.

2 FUNDAMENTOS TEÓRICOS

2.1 PROCESSAMENTO DE ÁUDIO

Uma das aplicações do processamento de áudio é o “Speech Enhancement” [8], ou seja: a melhoria da inteligibilidade e qualidade da voz, quando essa é degradada ou distorcida em ambientes ruidosos.

O “Speech Enhancement” é um resultado difícil de obter por duas razões. A primeira, por causa da natureza e características do ruído, que podem mudar drasticamente no tempo, e dependem da aplicação. É recomendável que se use algoritmos versáteis e que estes funcionem em ambientes práticos do dia-a-dia. Segundo, o parâmetro de desempenho pode ser definido diferentemente para cada aplicação. Os critérios mais usados são a inteligibilidade e a qualidade. Enquanto a qualidade é subjetiva (reflete a opinião individual do ouvinte), a inteligibilidade é objetiva (oferece a percentagem de palavras que foram identificadas corretamente pelo ouvinte).

Não é fácil satisfazer ambos os critérios ao mesmo tempo. Quando a degradação da voz ocorre por causa de um ruído aditivo não-correlacionado, é possível a melhora da qualidade em detrimento da inteligibilidade. Não obstante, é preciso ter cuidado com a inteligibilidade, uma vez que, quando o sinal de voz é muito ruidoso, a tendência do ouvinte é perder a concentração ao longo do tempo, diminuindo o desempenho da inteligibilidade. Por esta razão, a inteligibilidade pode ser verificada de forma favorável, mas apenas após um longo período de escuta, dependendo da concentração do ouvinte.

Sistemas de comunicação nos quais o algoritmo de redução de ruído é requerido são inúmeros: VOIP, sistemas de telecomunicações de longa distância, teleconferência, telefones móveis e outros.

2.2 ALGORITMOS

2.2.1 Subtração Espectral

Se o sinal de voz de voz está contaminado com ruído cuja distribuição espectral tem banda estreita, a componente de ruído pode ser suprimida com aplicação de uma filtragem digital. Contudo, se a componente de ruído contiver uma distribuição espectral larga, uma simples fil-

tragem de supressão de banda pode não ser apropriada. O método da Subtração Espectral é uma abordagem eficaz para reduzir o ruído de fundo estacionário.

Este método é baseado na premissa de que o espectro na frequência do sinal de voz é expresso como a soma do espectro de voz e do espectro do ruído. O processamento é feito inteiramente no domínio da frequência.

Considere um sinal de fala com ruído, $x(n)$. Este sinal pode ser assumido com sendo composto por um sinal de voz sem ruído $s(n)$ e por um sinal de voz de ruído $v(n)$, ou seja:

$$x(n) = s(n) + v(n)$$

Calculando-se a transformada de Fourier de ambos os lados da equação acima, obtemos:

$$X(j\omega) = S(j\omega) + V(j\omega)$$

Assim:

$$\begin{aligned} |X(j\omega)|^2 &= |S(j\omega)|^2 + |V(j\omega)|^2 + 2|S(j\omega)||V(j\omega)|\cos\Theta \\ &= S^2(\omega) + V^2(\omega) + 2S(\omega)V(\omega)\cos\Theta \end{aligned}$$

onde $S(\omega)$ e $V(\omega)$ são as magnitudes de $S(j\omega)$ e $V(j\omega)$, respectivamente, e Θ é a diferença de fase entre o sinal de voz e o sinal de ruído. Assumindo que $s(n)$ e $v(n)$ são processos aleatórios estacionários e não-correlacionados, podemos fazer a seguinte aproximação:

$$X^2(\omega) = S^2(\omega) + V^2(\omega)$$

A potência instantânea do espectro ou o quadrado da magnitude de espectro do sinal de voz, $S^2(\omega)$, pode ser recuperada subtraindo-se uma estimativa de $V^2(\omega)$ de $X^2(\omega)$, ou seja:

$$\hat{S}^2(\omega) = X^2(\omega) - \hat{V}^2(\omega) = S^2(\omega) + [V^2(\omega) - \hat{V}^2(\omega)]$$

Logo, a magnitude do espectro da voz é computada como:

$$\hat{S}(\omega) = \sqrt{\hat{S}^2(\omega)} = \sqrt{X^2(\omega) - \hat{V}^2(\omega)}$$

A magnitude do espectro da voz pode ser diretamente estimada pela equação:

$$\hat{S}(\omega) = X(\omega) - \hat{V}(\omega) = S(\omega) + [V(\omega) - \hat{V}(\omega)]$$

No caso mais geral, pode-se expressar a B-ésima potência do espectro de potência do sinal de voz como:

$$\hat{S}^B(\omega) = X^B(\omega) - n\hat{V}^B(\omega)$$

onde B é um inteiro (normalmente igual a 1 ou a 2) e n é um parâmetro para controlar a quantidade de ruído a ser subtraída. Conseqüentemente, a estimativa do espectro da voz pode ser obtida pela equação:

$$\hat{S}(j\omega) = [X^B(\omega) - n\hat{V}^B(\omega)]^{1/B} e^{j\Psi(\omega)}$$

sendo $\Psi(\omega)$ a fase de $X(j\omega)$.

A subtração instantânea da potência de espectro resulta de $B=2$ e $n=1$, e a subtração da magnitude de espectro origina-se de $B=1$ e $n=1$.

A estimação do espectro do ruído é feita segmentando-se o sinal de voz de ruído $x(n)$ em blocos de N amostras. Cada bloco é transformado, via DFT, em um bloco de N amostras espectrais. Blocos sucessivos de amostras espectrais formam uma matriz bidimensional (com informação tempo-freqüência), denotada por $X_T(j\omega)$, na qual T é o índice do número do bloco (*frame*) e designa a dimensão do tempo.

A estimação do ruído é expressa como:

- $\hat{V}_T^B(\omega) = \alpha_A \hat{V}_{T-1}^B(\omega) + (1 - \alpha_A) X_T^B(\omega)$, se $X_T^B \geq V_{T-1}^B(\omega)$
- $\hat{V}_T^B(\omega) = \alpha_I \hat{V}_{T-1}^B(\omega) + (1 - \alpha_I) X_T^B(\omega)$, se $X_T^B < V_{T-1}^B(\omega)$

Nas respectivas equações, α_I e α_A são parâmetros que controlam as constantes de tempo das recursões, atualizando mais rapidamente nos casos em que o ruído é predominante no sinal, e mais lentamente nos casos em que ele não é tão relevante.

Na figura 2.1 é mostrado o diagrama de blocos do algoritmo de Subtração Espectral.

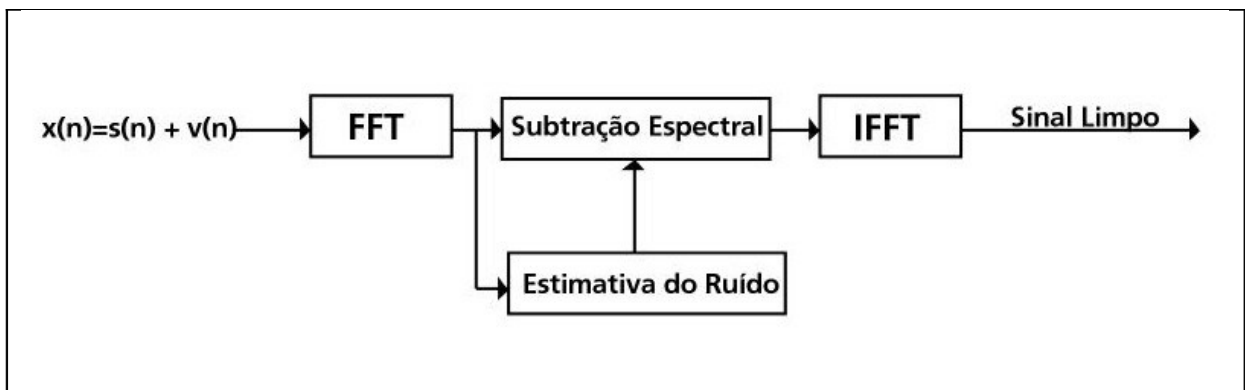


Figura 2.1: Diagrama de Blocos do Algoritmo de Estimação Espectral

2.2.2 Filtro de Wiener

O objetivo do filtro de Wiener é a redução do ruído a partir das estatísticas de segunda-ordem desse. O Filtro de Wiener assume que o sinal de voz e o ruído aditivo são processos estocásticos estacionários, com características espectrais conhecidas.

Considere um sinal de voz com ruído $x(n)$. Como no método anterior, o sinal de voz com ruído pode ser assumido com sendo um sinal de voz sem ruído $s(n)$ acrescido de um sinal de ruído $v(n)$, ou seja:

$$x(n) = s(n) + v(n)$$

sendo $s(n)$ e $v(n)$ não correlacionados.

No domínio da transformada de Fourier temos:

$$X(j\omega) = S(j\omega) + V(j\omega)$$

O Filtro de Wiener corresponde a uma estimativa do filtro ótimo MMSE (mínimo erro médio quadrático), o qual pode ser descrito como:

$$\hat{H}_z(\omega) = \frac{E[S^2(\omega)]}{E[S^2(\omega)] + E[V^2(\omega)]}$$

Então, a estimativa do sinal de voz através do filtro é:

$$\hat{S}(j\omega) = \hat{H}_z(\omega)X(j\omega)$$

Podemos observar que a subtração espectral da potência instantânea (descrita na seção anterior) pode ser escrita como:

$$\hat{S}(j\omega) = \sqrt{X^2(\omega) - \hat{V}^2(\omega)} e^{j\psi(\omega)}$$

Denotamos:

$$\hat{H}_{IP}(\omega) = \left[\frac{X^2(\omega) - \hat{V}^2(\omega)}{X^2(\omega)} \right]^{1/2},$$

Logo:

$$\hat{S}(j\omega) = \hat{H}_{IP}(\omega)X(j\omega)$$

Isso significa que recuperar um sinal de voz através da subtração espectral da potência instantânea é equivalente a submeter o sinal de voz ruidoso a um filtro cuja função de transferência é $\hat{H}_{IP}(\omega)$. Portanto, há uma relação muito próxima entre o método da subtração espectral e o filtro de Wiener.

Similarmente, na sua forma mais geral, a subtração espectral pode ser interpretada como uma forma paramétrica do filtro de Wiener, ou seja:

$$\hat{S}(j\omega) = \hat{H}_{p\sigma}(\omega)X(j\omega)$$

e

$$\hat{H}_{p\sigma}(\omega) = \{[X^B(\omega) - n\hat{V}^B(\omega)]/X^B(\omega)\}^{1/\Lambda}$$

onde n é o parâmetro de controle da quantidade de ruído a ser suprimida. O desempenho de redução do ruído para a forma paramétrica de Wiener pode ser expressa como:

$$\Gamma(\omega) = 1 / \hat{H}_{p\sigma}^2(\omega).$$

No filtro de Wiener, o ganho do filtro é importante para manter uma boa relação sinal de voz/ruído. A estimativa da SNR oscila devido à variância da DFT do espectro. Logo, uma aproximação pode ser feita para reduzir essa oscilação, denominada como média no tempo da DFT do espectro, realizada antes de se calcular a SNR. Se chamarmos $X_T(\omega)$ a magnitude do espectro do sinal de voz ruidoso em cada bloco com índice T , a aproximação supracitada pode ser implementada como:

- $\hat{X}_T(\omega) = \beta_A \hat{X}_{T-1}(\omega) + (1 - \beta_A)X_T(\omega)$, se $X_T(\omega) \geq \bar{X}_{T-1}(\omega)$
- $\hat{X}_T(\omega) = \beta_Y \hat{X}_{T-1}(\omega) + (1 - \beta_Y)X_T(\omega)$, se $X_T < \bar{X}_{T-1}(\omega)$

onde β_A e β_Y são parâmetros que controlam as constantes de tempo das recursões, como explicado na seção anterior.

2.2.3 Subspace Tracking

O algoritmo de Subspace Tracking[1],[6] é um método adaptativo, que consiste projetar o sinal ruidoso em dois subespaços: o subespaço do sinal e o subespaço do ruído. Uma estimativa do sinal limpo pode ser feita ao remover as componentes do sinal no subespaço do ruído e manter apenas as componentes do sinal no subespaço do sinal. A decomposição em dois subespaços pode ser feita usando a decomposição em autovalores.

O modelo usado assume que o ruído é aditivo e não correlacionado com o sinal de voz de voz. Utilizando uma notação matricial, podemos escrever:

$$\mathbf{y} = \mathbf{x} + \mathbf{n}$$

onde \mathbf{y} , \mathbf{x} e \mathbf{n} são vetores com dimensão K que descrevem o sinal de voz com ruído, o sinal de voz limpo e o ruído branco, respectivamente. Seja $\hat{\mathbf{x}} = \mathbf{H}\mathbf{y}$ uma estimativa do sinal limpo, aonde \mathbf{H} é uma matriz $K \times K$. O erro do sinal de voz é dado por:

$$\boldsymbol{\varepsilon} = \hat{\mathbf{x}} - \mathbf{x} = (\mathbf{H} - \mathbf{I})\mathbf{x} + \mathbf{H}\mathbf{n} = \boldsymbol{\varepsilon}_X + \boldsymbol{\varepsilon}_N$$

Na equação acima $\boldsymbol{\varepsilon}_X$ representa a distorção do sinal de voz e $\boldsymbol{\varepsilon}_N$ a distorção residual. A energia da distorção do sinal de voz é $\boldsymbol{\varepsilon}_X = \text{tr}(E\{\boldsymbol{\varepsilon}_X \boldsymbol{\varepsilon}_X^T\})$ e a matriz de autovetores da matriz de covariância de \mathbf{x} é representada por $\mathbf{U} = [\mathbf{u}_1 \ \dots \ \mathbf{u}_K]$. O avaliador de SDC (domínio espectral limitado) designado por \mathbf{H} é obtido como:

$$\mathbf{H} = \arg \min \boldsymbol{\varepsilon}_X^2, \quad k=1, \dots, M$$

com limite

$$E\{|\mathbf{u}_N^T \boldsymbol{\varepsilon}_N|^2\} \leq \alpha_k \vartheta^2$$

A variância do ruído e a dimensão do sub-espaço são, respectivamente, ϑ^2 e M . Seja o k -ésimo autovalor $\lambda(k)$ associado a um auto-vetor \mathbf{u}_k . Considerando que os autovalores são ordenados de forma decrescente conforme os seus valores, para $k=1, \dots, K$, e que o avaliador \mathbf{H} é [1]:

$$\mathbf{H} = \mathbf{U}\mathbf{Q}\mathbf{U}^T$$

onde \mathbf{Q} é a uma matriz diagonal com elemento não-nulo da k -ésima coluna igual a q_k , faz-se:

$$q_k = \alpha^{1/2}, \quad k=1, \dots, M$$

$$q_k = 0, \quad k=M+1, \dots, K$$

sendo $\alpha = \exp\{-\vartheta^2 / \lambda(k)\}$. Para $q_k = 0$, assumimos que há somente ruído.

O avaliador requer uma estimativa dos autovetores e autovalores da matriz de covariância do sinal de voz limpo. Como o ruído é tido como branco e não correlacionado com o sinal de voz limpo, fica claro que os autovetores do sinal de voz com ruído são os mesmos para o sinal de voz limpo. O $\lambda(k)$ pode se obtido subtraindo os autovalores do ruído dos autovalores do sinal ruidoso.

O algoritmo recursivo a seguir é utilizado para achar os autovetores da matriz de covariância \mathbf{U} [1]

$$\mathbf{U}(0) = [\mathbf{u}_1(0) \ \dots \ \mathbf{u}_K(0)] = \mathbf{I}_K$$

$$d_i(0) = 0$$

Para $n=1, 2, \dots$ fazer

$$\mathbf{y}_i(n) = \mathbf{y}(n)$$

Para $i=1, 2, \dots, k$ fazer

$$v_i(n) = \mathbf{u}_i^T(n-1) \mathbf{y}_i(n)$$

$$d_i(n) = d_i(n-1) + |v_i(n)|^2$$

$$\mathbf{e}_i = \mathbf{y}_i(n) - \mathbf{u}_i(n-1) * v_i(n)$$

$$\mathbf{u}_i(n) = \mathbf{u}_i(n-1) + \mathbf{e}_i(n) \frac{v_i(n)}{d_i(n)}$$

$$\mathbf{y}_{i+1}(n) = \mathbf{y}_i(n) - \mathbf{u}_i(n)v_i(n)$$

End

End

$$\mathbf{U}(n) = [\mathbf{u}(0)_1 \quad \dots \quad \mathbf{u}_K(n)]$$

2.2.4 Subspace Tracking com a Transformada Wavelet

Esta técnica baseia-se em dividir o sinal de voz de entrada em sub-bandas. Para cada sub-banda aplica-se o algoritmo de subspace tracking. Apresentamos a seguir uma descrição resumida da transformada wavelet.

2.2.4.1 Transformada Wavelet

A Transformada Wavelet[7] pode ser vista como técnica por janelas, com dimensão variável de janela, o que permite avaliação do sinal sob análise, no espaço tempo versus frequência.

Wavelets são "ondas pequenas", funções obtidas a partir de uma função protótipo, denominada wavelet mãe $\psi(t)$, pertencente a $L^2(\mathbb{R})$ (funções reais com energia finita), por meio de dilatações (contrações ou escalamento) e translações (deslocamentos). O objetivo primordial em se realizar estas operações é o de se obter uma família de funções base para descrever outras funções pertencentes a $L^2(\mathbb{R})$.

A utilidade das wavelets está na sua possibilidade de atuarem como funções base na decomposição de outras funções $\in L^2(\mathbb{R})$. As bases wavelets podem ser redundantes ou ortonormais, cobrindo e descrevendo completamente o espaço $L^2(\mathbb{R})$, de uma forma mais sofisticada que as bases senoidais dos métodos de Fourier. Análise com a transformada Wavelet permite o uso de intervalos de tempo longos onde se queira informação de baixa-freqüência mais precisa, e regiões mais curtas onde se deseje informação de alta-freqüência.

Além desses aspectos, outras propriedades são desejadas e exigidas das famílias wavelets para que sejam úteis em aplicações práticas em processamento de sinais e na análise de espaços funcionais. Neste contexto, wavelets devem satisfazer requisitos como:

- possuírem energia finita, de forma que a análise preserve a energia, e que a síntese leve a uma reconstrução perfeita.
- possuírem certo grau de regularidade (suavidade)
- serem nulas no infinito
- possuírem certo número de momentos nulos
- que sejam funções de classe C^k ($0 < k < \infty$)
- que tenham suporte compacto, no tempo e na frequência.

A transformada wavelet integral (contínua), em se tratando de aplicações com sinais ópticos (analógicos), pode ser implementada através de filtros ópticos, à velocidade da luz, com todas as vantagens (e desvantagens) inerentes a esse tipo de processamento analógico. A implementação da transformada contínua em sistemas eletrônicos ficará, entretanto, sujeita aos fatores inerentes ao processamento elétrico, como limitações na resolução e precisão, margem de ruído estocástico, efeitos não lineares, etc.

Em princípio, a implementação da transformada wavelet contínua em sistemas digitais implica num processamento no qual as funções envolvidas, seus respectivos cálculos e transformações, são realizados de forma a simular discretamente o processamento analógico. Simulações digitais de sistemas contínuos normalmente são acompanhadas de uma alta carga computacional a fim de se reproduzir o efeito analógico tão próximo quanto possível, e não se esquivam das limitações impostas pelo interfaceamento análogo-digital quanto à resolução da amostragem, erros de quantização, precisão da conversão, etc. A alternativa natural é procurar abordar a transformada wavelet em sua forma matemática discreta, cuja implementação computacional é direta, dispensando os estágios sensíveis a erros e de difícil implementação em tempo real.

2.2.4.2 Wavelet Daubechies

Neste projeto utilizamos a wavelet Daubechies [7]. As origens das wavelets de Daubechies estão ligadas a famílias de filtros com propriedades especiais. Dois canais de filtros existem nas implementações por banco de filtros, um estando associado às wavelets e outro associado às funções escaladoras. Os filtros de Daubechies são ortogonais, e exibem máxima planura (*maximum flatness*) em $\omega=0$ e $\omega=\pi$, isto é, maximizam a suavidade nas funções escaladoras associadas maximizando a taxa de decaimento de suas transformadas de Fourier.

A transformada wavelet pode ser tratada sob a forma discreta sem prejuízo de suas qualidades e propriedades, e de maneira numericamente estável. A Transformada Wavelet consiste na decomposição do sinal de voz no domínio da frequência, com aplicação de filtros passa-baixas e passa-altas, conforme ilustrado na Figura 2.2.

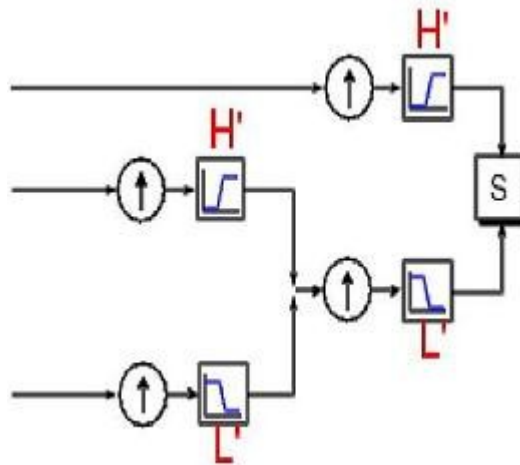


Figura 2.2: Transformada Wavelet

Fonte:ToolBox do Matlab 07

Se desejarmos que as wavelets sejam úteis para análise de sinais regulares e suaves, é necessário impor condições sobre os filtros associados a elas além das exigências da capacidade de reconstrução perfeita e da preservação de energia. Condições impostas sobre o grau de regularidade da wavelet, sua taxa de decaimento no infinito e seu número de momentos nulos irão habilitá-las a melhor aproximarem e analisarem um maior número de classes de sinais e funções, bem como produzirem melhores resultados e desempenho. O grau de regularidade da wavelet e a sua taxa de decaimento são governados pelo número de momentos nulos que apresenta. Esta propriedade é importante para deduzir as propriedades de aproximação exibidas pela wavelet nos espaços de multiresolução.

Momentos nulos também impõem uma condição necessária para que as wavelets sejam N vezes diferenciáveis, isto é, de classe C^N . As wavelets de Daubechies são numeradas em função do número de momentos nulos que possuem. O índice em si corresponde ao número de coeficientes que os filtros associados possuem, que também é o número de derivadas nulas do filtro associado à função escaladora em $\omega=\pi$ (o filtro tem um zero em $\omega=\pi$ de ordem $N-1$). Assim, a wavelet D4 possui 2 momentos nulos, e os filtros associados possuem 4 zeros em $\omega=\pi$, e 4 taps (4 coeficientes). As wavelets D16 possuem 8 momentos nulos, e seus filtros associados têm 16 coeficientes. A taxa de decaimento e grau de suavidade para esta wavelet são

maiores do que para a D4, e elas se mostram mais eficientes para representar sinais de áudio.

Nas Figuras 2.3 e 2.4 temos, respectivamente, as Wavelets de Daubechies D4 e D16 e suas Transformadas de Fourier.

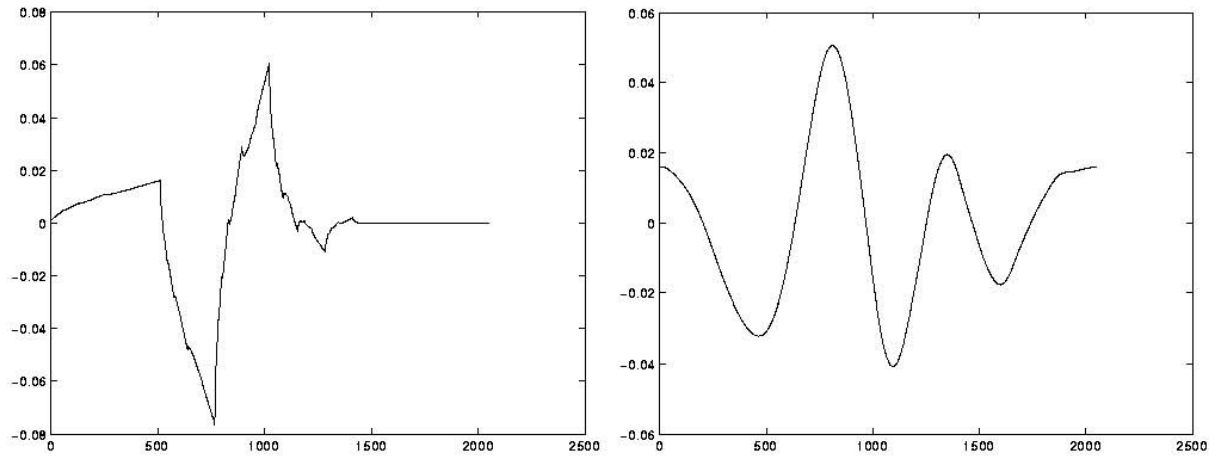


Figura 2.3: Wavelets de Daubechies D4 e D16

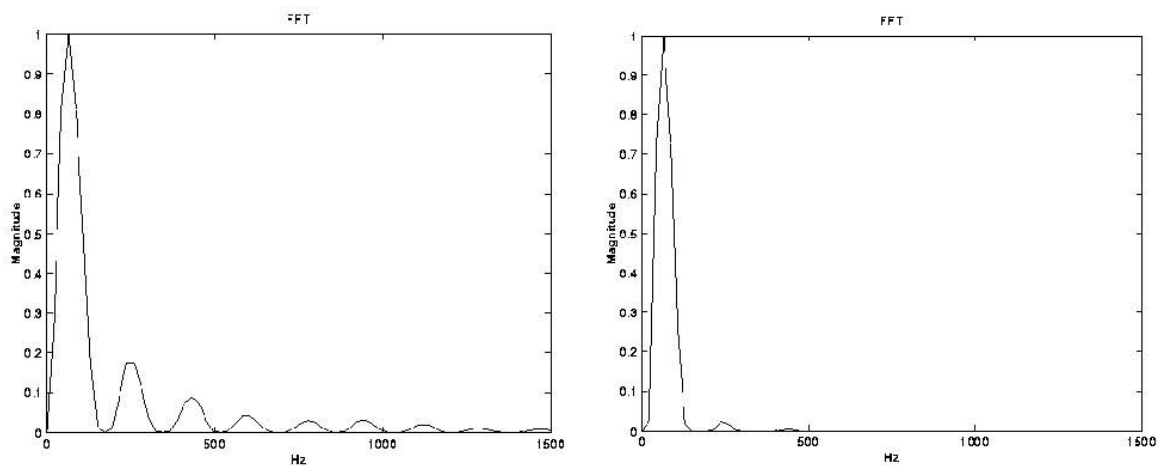


Figura 2.4: Transformadas de Fourier das wavelets D4 e D16

3 RESULTADOS

Nesta capítulo, iremos discorrer acerca da performance de cada algoritmo em relação ao sinal de voz contaminado por diferentes tipos de ruído.

3.1 Ruído de Avião

A seguir, mostramos os resultados obtidos com os diferentes métodos para o sinal de voz contaminado por um ruído de avião.

3.1.1 Subtração Espectral

A Figura 3.1 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz obtido após a aplicação do método da subtração espectral (sinal limpo).

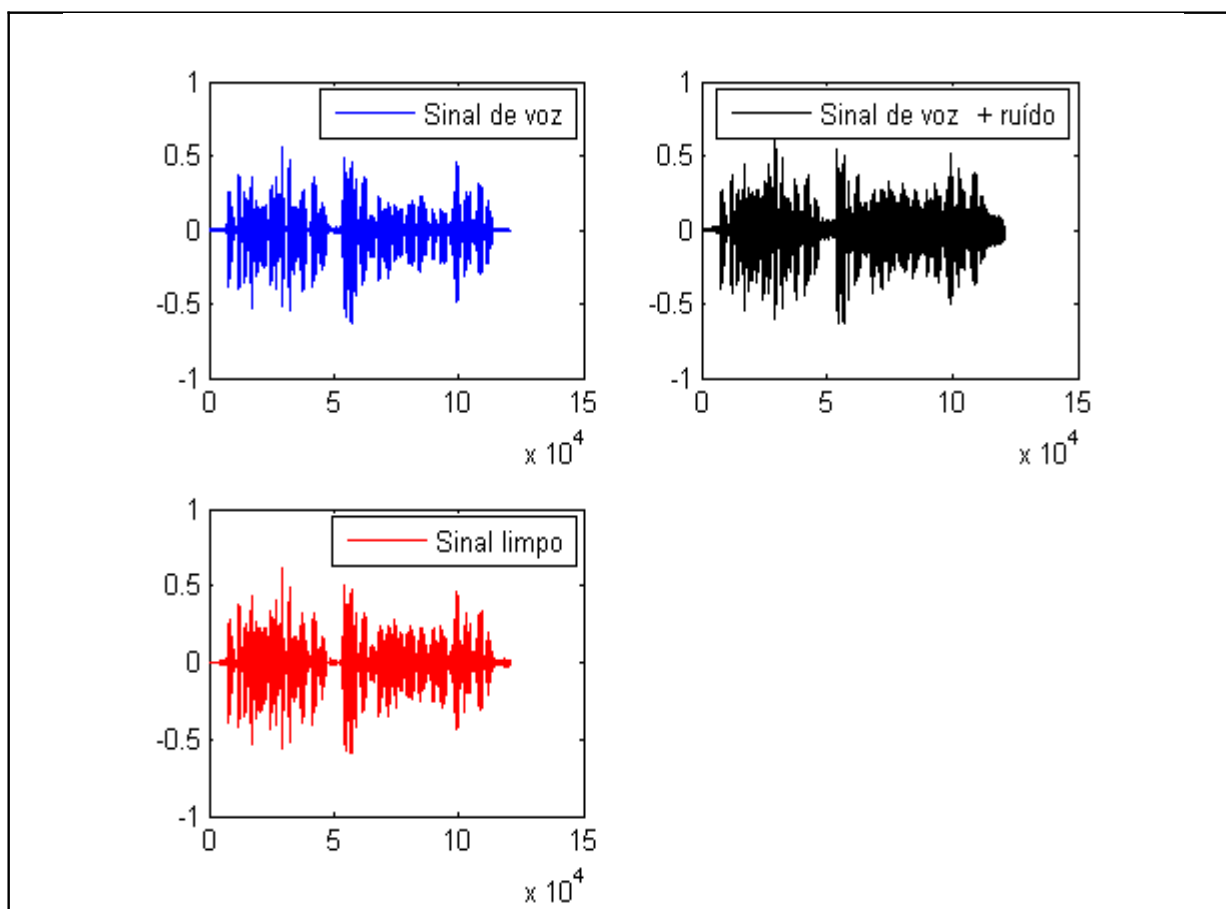


Figura 3.1.: Subtração Espectral aplicada ao sinal de voz com ruído de avião

A Figura 3.2 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.1.

A partir das Figuras 3.1 e 3.2 verificamos que o algoritmo da subtração espectral conseguiu reduzir consideravelmente o ruído. Entretanto, através de uma avaliação auditiva, percebe-se a introdução de um ruído de fundo (conhecido como “*musical noise*”) no momento em que o ruído de avião se torna mais forte no sinal de voz.

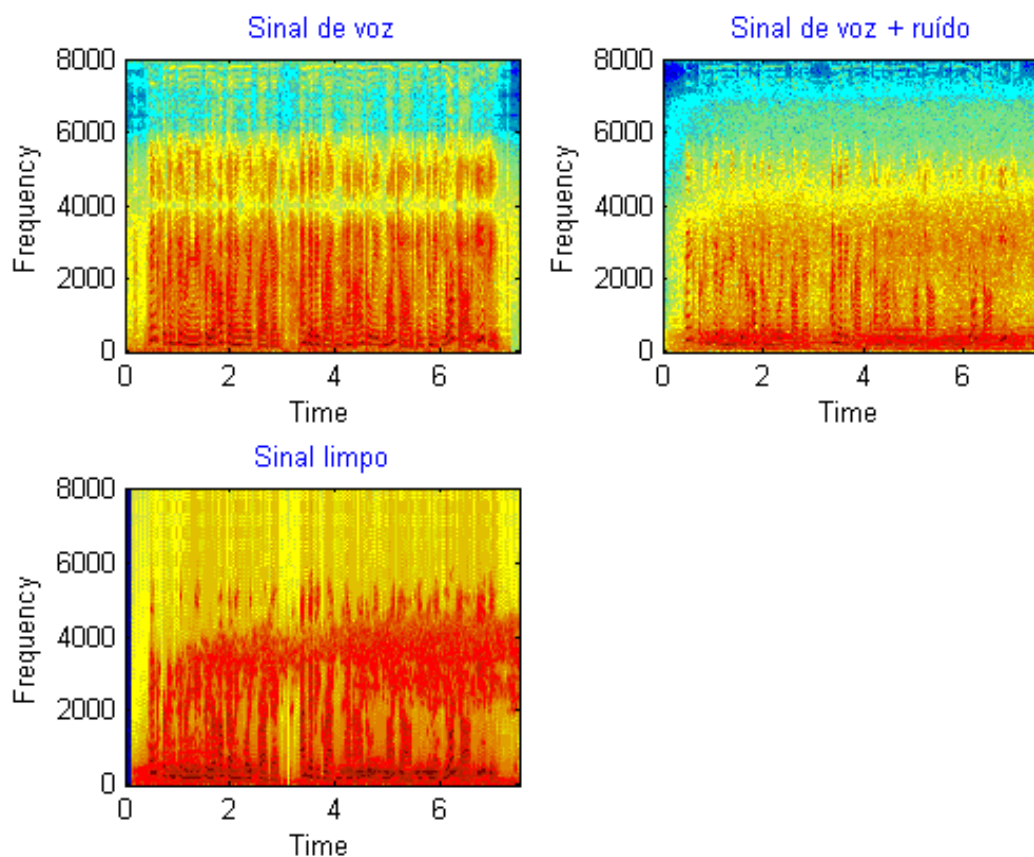


Figura 3.2: Espectrograma do sinal de voz com ruído de avião aplicando-se o algoritmo de a Subtração Espectral

3.1.2 Filtro de Wiener

A Figura 3.3 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e o sinal voz depois de ter sido aplicado o Filtro de Wiener (sinal limpo).

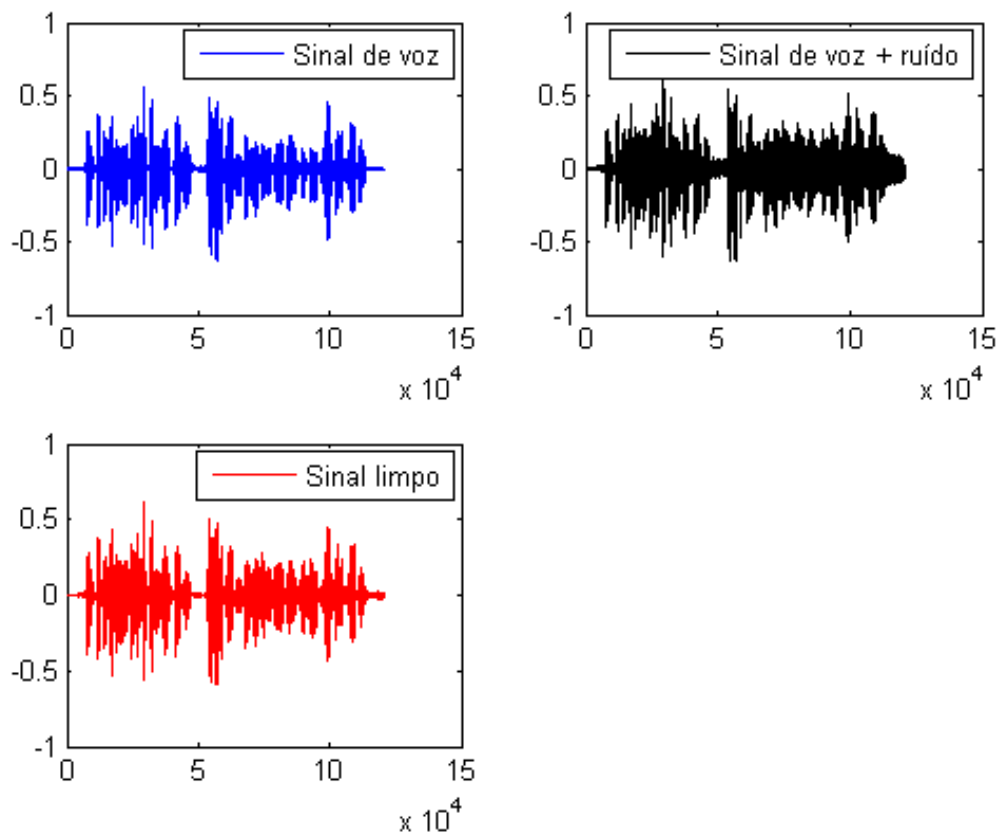


Figura 3.3: Filtro de Wiener aplicado ao sinal de voz com ruído de avião

A Figura 3.4 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.3.

Semelhante ao algoritmo da Subtração Espectral, o Filtro de Wiener não conseguiu evitar o surgimento do ruído de fundo.

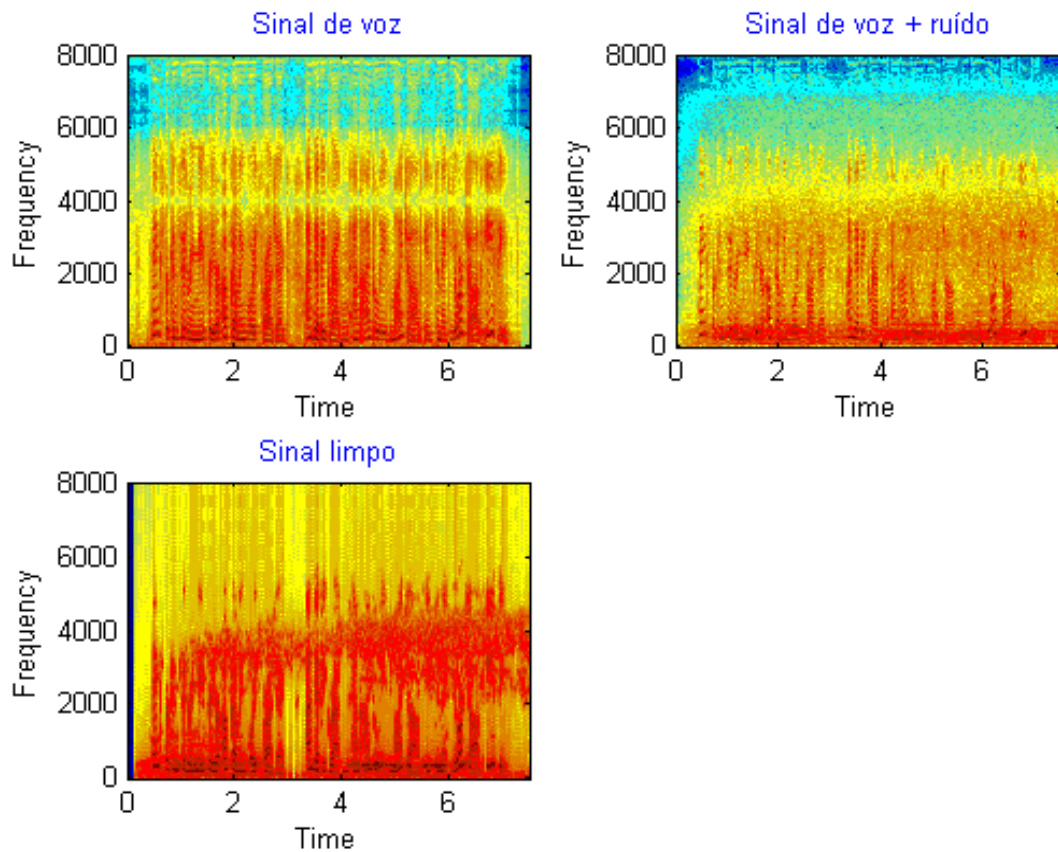


Figura 3.4: Espectrogramas do sinal de voz com ruído de avião aplicando-se o algoritmo de o Filtro de Wiener

3.1.3 Subspace Tracking

A Figura 3.5 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo Subspace Tracking.

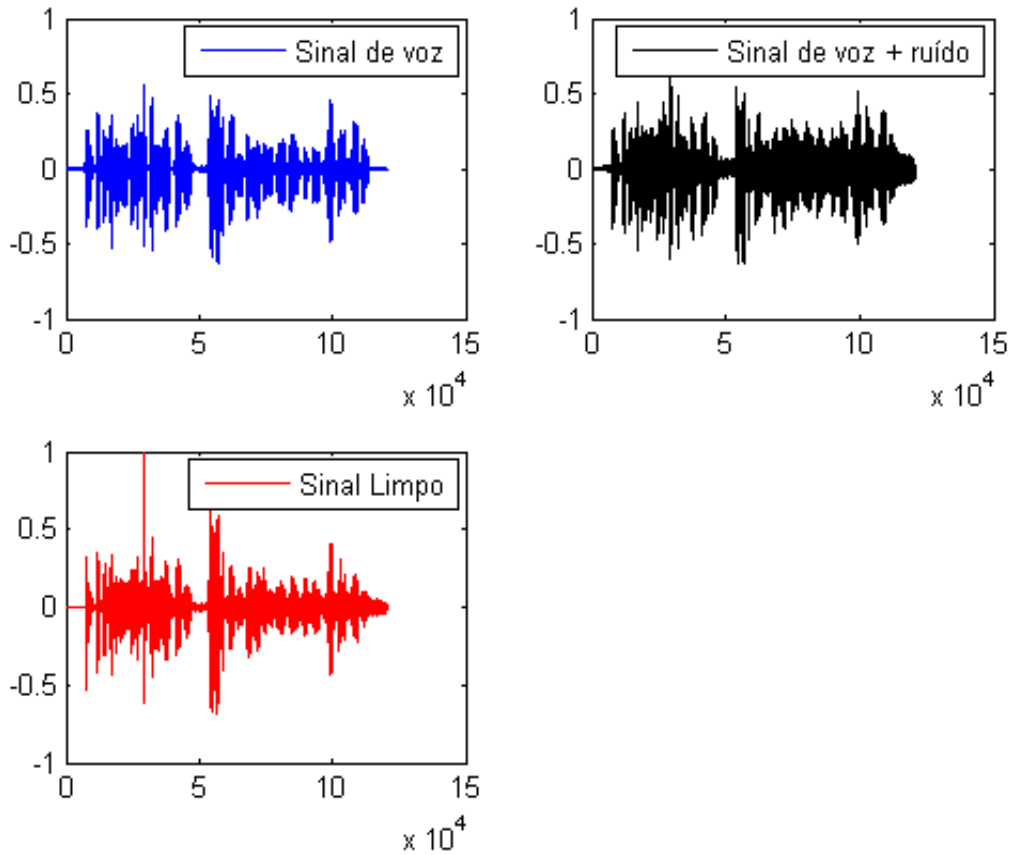


Figura 3.5: Subspace Tracking aplicado ao sinal de voz com ruído de avião

A Figura 3.6 apresenta os gráficos dos espectrogramas correspondentes dos sinais da Figura 3.5.

No Subspace tracking, o sinal de voz não apresenta o “musical noise” como aconteceu nos algoritmos supracitados. Contudo, o ruído de avião é nitidamente perceptível no sinal de voz. Além da baixa supressão do ruído, notamos a introdução de componentes em alta frequência no sinal de voz, o que acarretou distorções do mesmo.

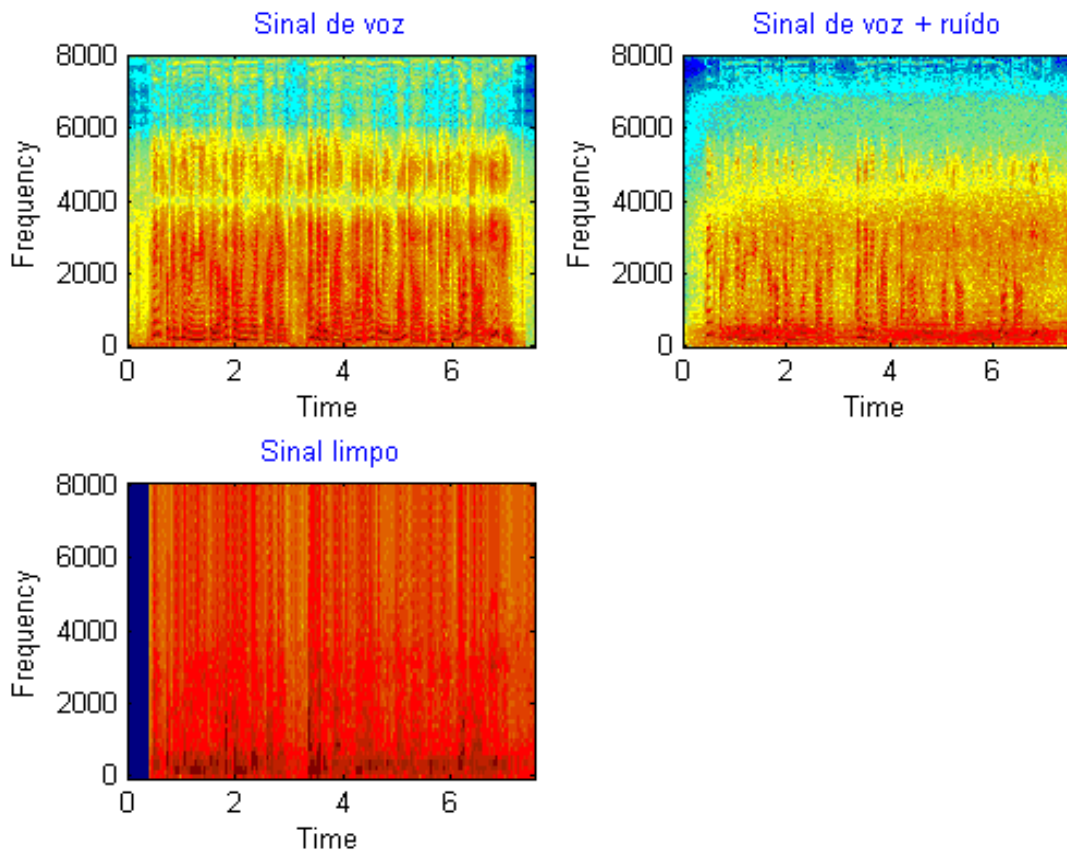


Figura 3.6: Espectrograma do sinal de voz com ruído de avião aplicando-se o algoritmo de a Subspace Tracking

3.1.4 Subspace Tracking com Transformada Wavelet

A Figura 3.7 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal voz depois de ter sido aplicado o algoritmo subspace tracking com transformada wavelet.

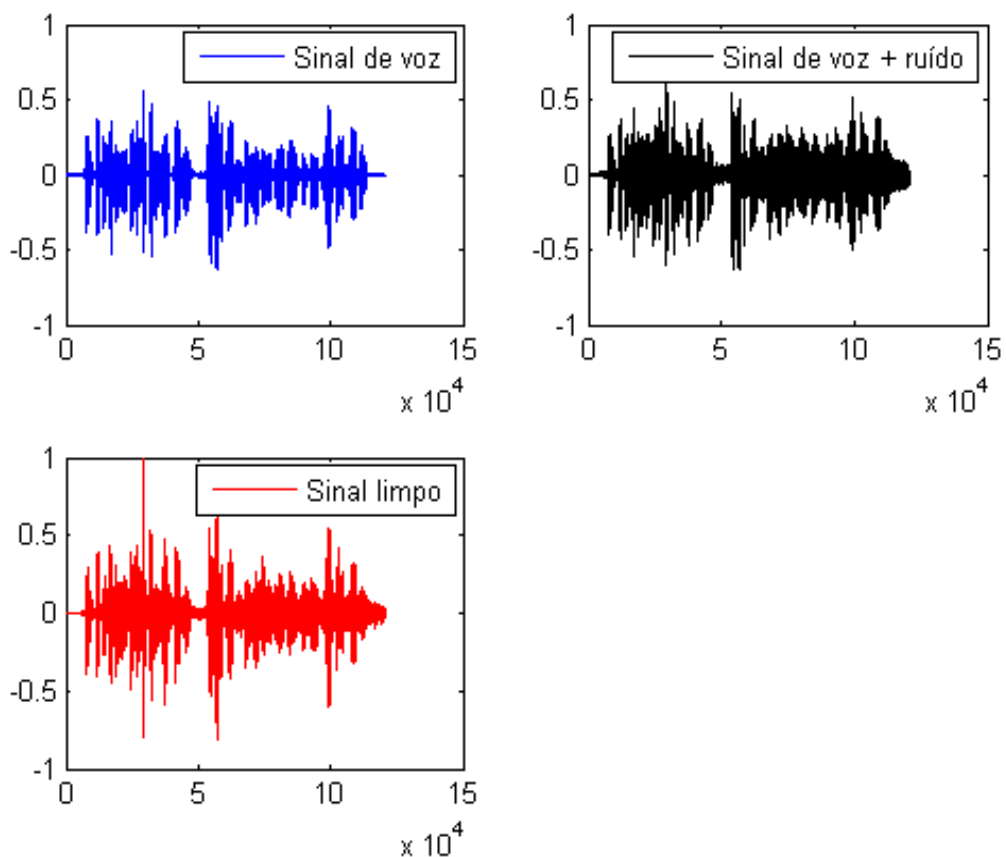


Figura 3.7: Subspace Tracking utilizando-se a transformada Wavelet aplicado ao sinal de voz com ruído de avião

A Figura 3.8 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.7.

No subspace tracking usando a transformada wavelet mantém-se a inexistência do “musical noise”. O ruído de avião ainda é forte no sinal de voz, mas menor do que sem a utilização da transformada wavelet. A introdução de componentes em alta frequência ainda existe, mas de forma mais branda. Logo, é possível concluir que a decomposição do sinal de voz usando a wavelet diminui as distorções no sinal de voz provocadas pelo surgimento de componentes em alta frequência no método do subspace tracking.

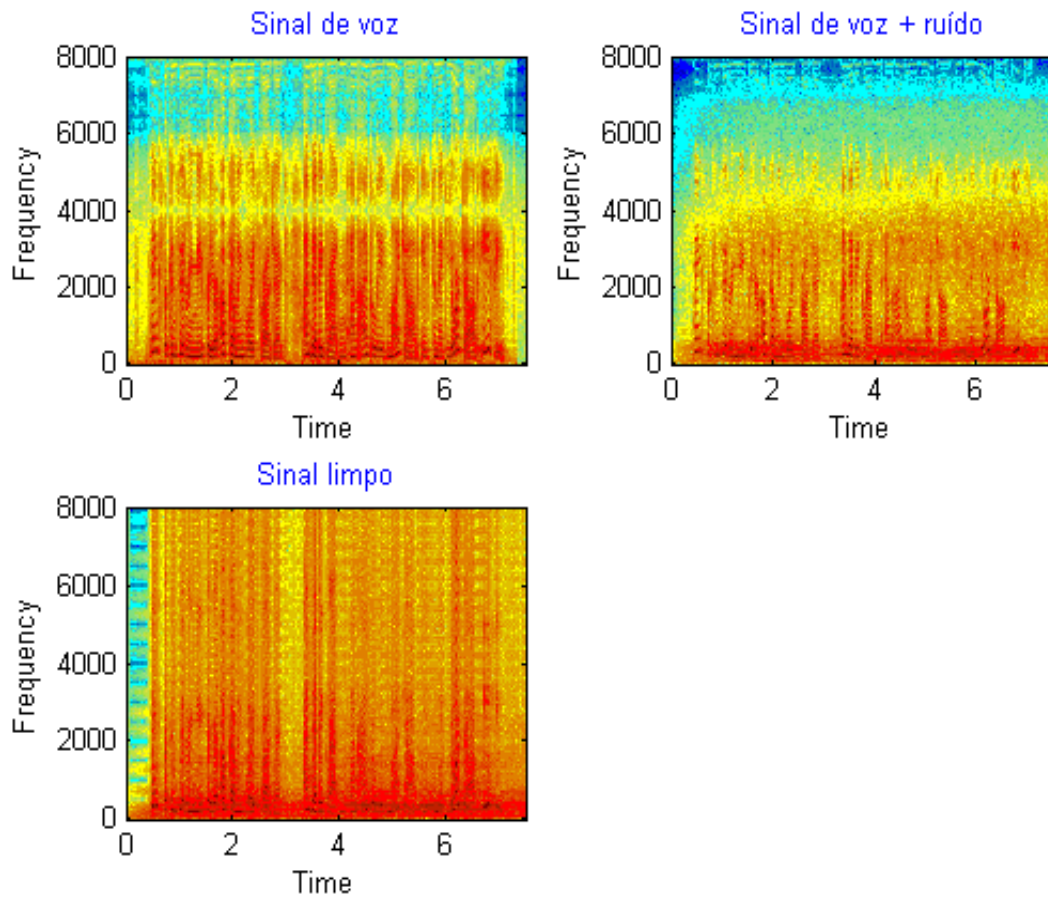


Figura 3.8: Espectrograma do sinal de voz com ruído de avião aplicando-se o algoritmo Subspace Tracking com a transformada Wavelet.

3.1.5 Comparação entre os Algoritmos

A Figura 3.9 apresenta os resultados obtidos com cada algoritmo para a redução de ruído de avião no sinal de voz.

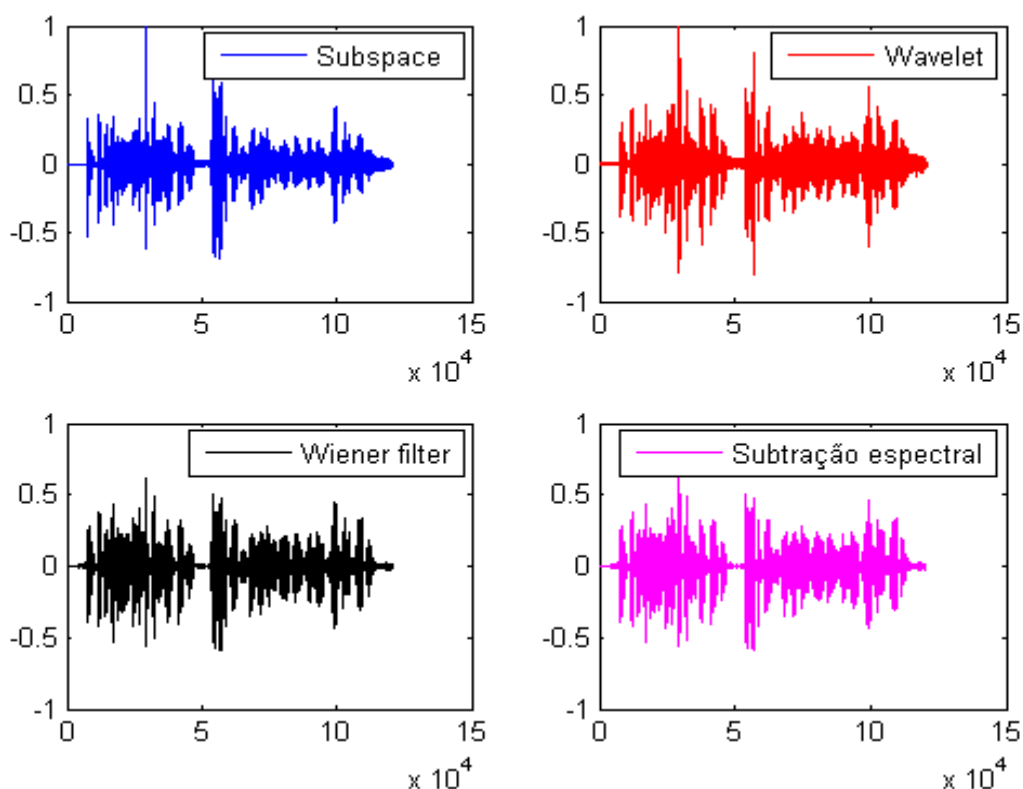


Figura 3.9: Comparação entre os algoritmos aplicados ao sinal de voz com ruído de avião

Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído de avião. Destacamos a presença do “musical noise” nos sinais resultantes dos métodos Filtro de Wiener e Subtração Espectral, e de componentes de alta frequência, antes não existente, nos sinais obtidos com o Subspace Tracking e, em menor intensidade, com o Subspace Tracking com a transformada wavelet. Do ponto de vista subjetivo, a maior parte dos ouvintes indicou o Filtro de Wiener como o algoritmo que melhor eliminou o ruído.

3.2 Ruído de Carro

A seguir, iremos discutir acerca do desempenho de cada algoritmo em relação ao sinal de voz contaminado pelo ruído de carro.

3.2.1 Subtração Espectral

A Figura 3.10 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo da subtração espectral.

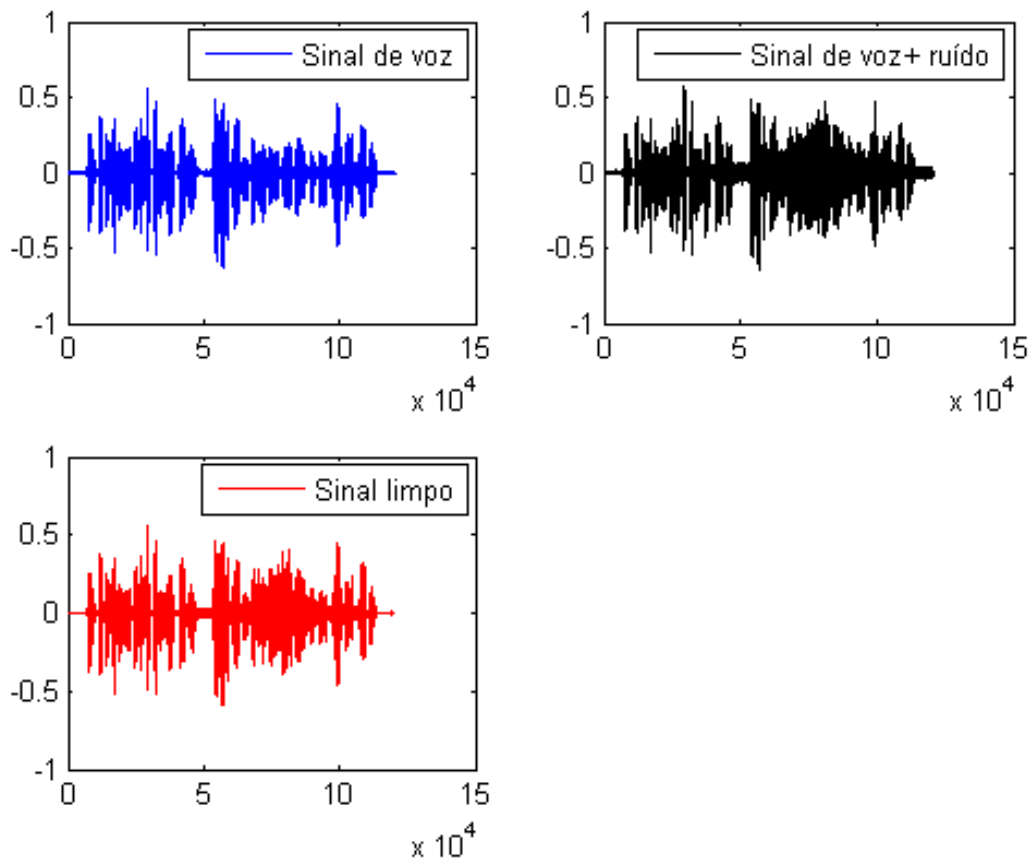


Figura 3.10: Subtração Espectral aplicado ao sinal de voz com ruído de carro

A Figura 3.11 apresenta os gráficos dos espectrogramas dos sinais da Figura. 3.10.

O algoritmo da subtração espectral conseguiu reduzir o ruído. Entretanto, temos presente o ruído de fundo (“musical noise”) de modo intenso em parte do sinal de voz.

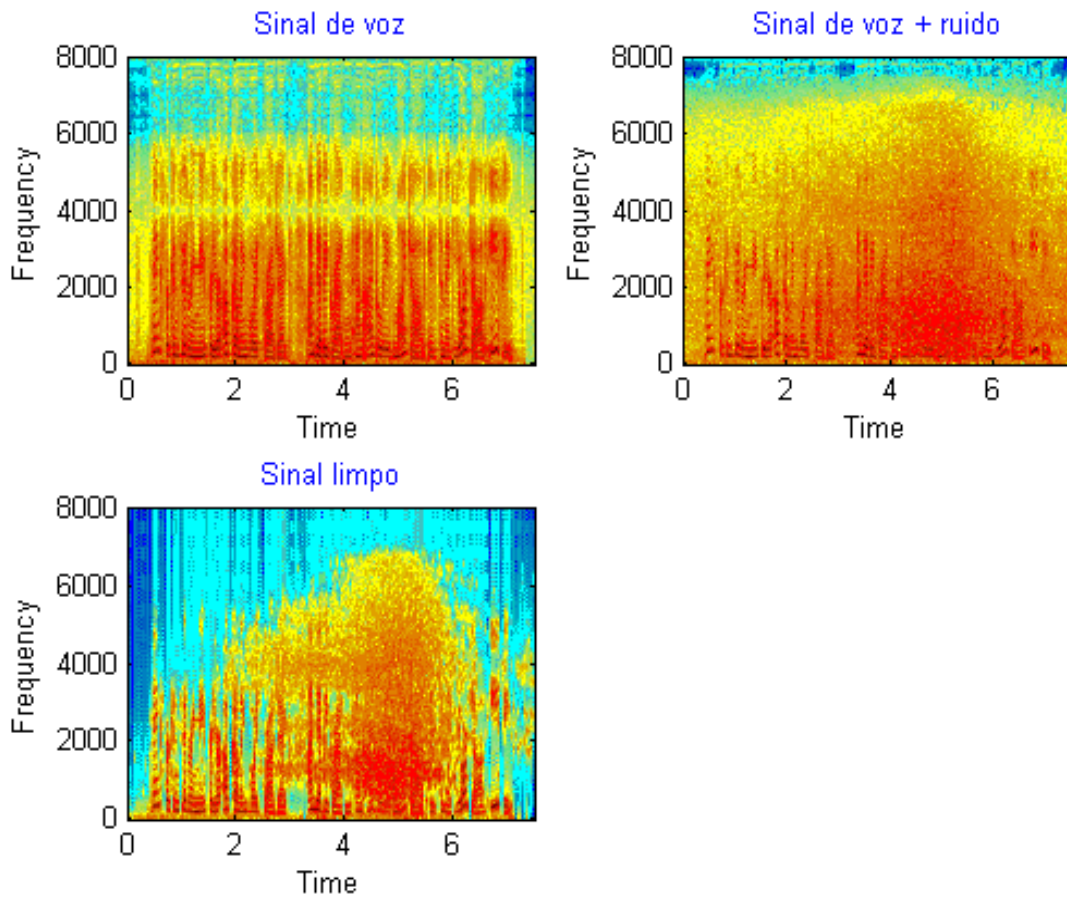


Figura 3.11: Espectrograma do sinal de voz com ruído de carro aplicando-se o algoritmo Subtração Espectral

3.2.2 Filtro de Wiener

A Figura 3.12 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste e do sinal voz depois de ter sido aplicado o Filtro de Wiener.

A Figura 3.13 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.12.

Semelhante ao algoritmo de subtração espectral, o Filtro de Wiener não conseguiu evitar o surgimento do ruído musical. Analisando os espectrogramas, percebe-se que houve perda de informação do sinal de voz na faixa de frequência entre 4 kHz e 6 kHz.

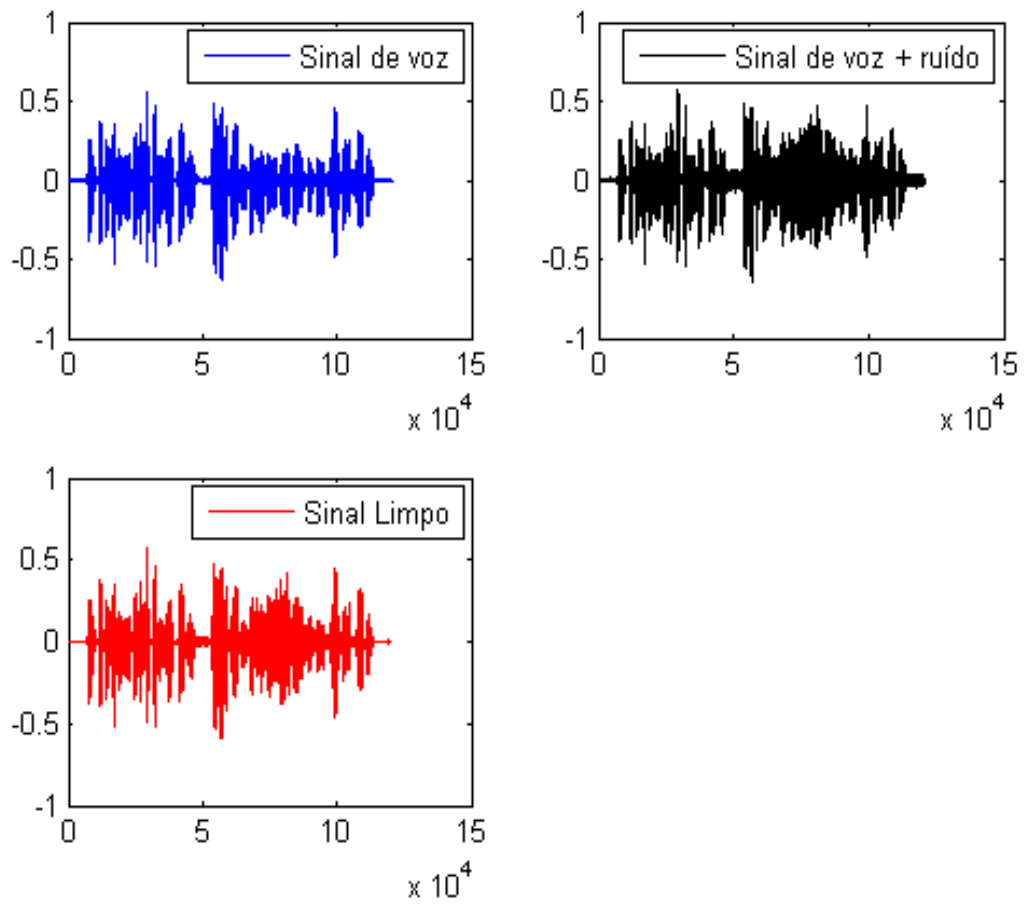


Figura 3.12: Filtro de Wiener aplicado ao sinal de voz com ruído de carro

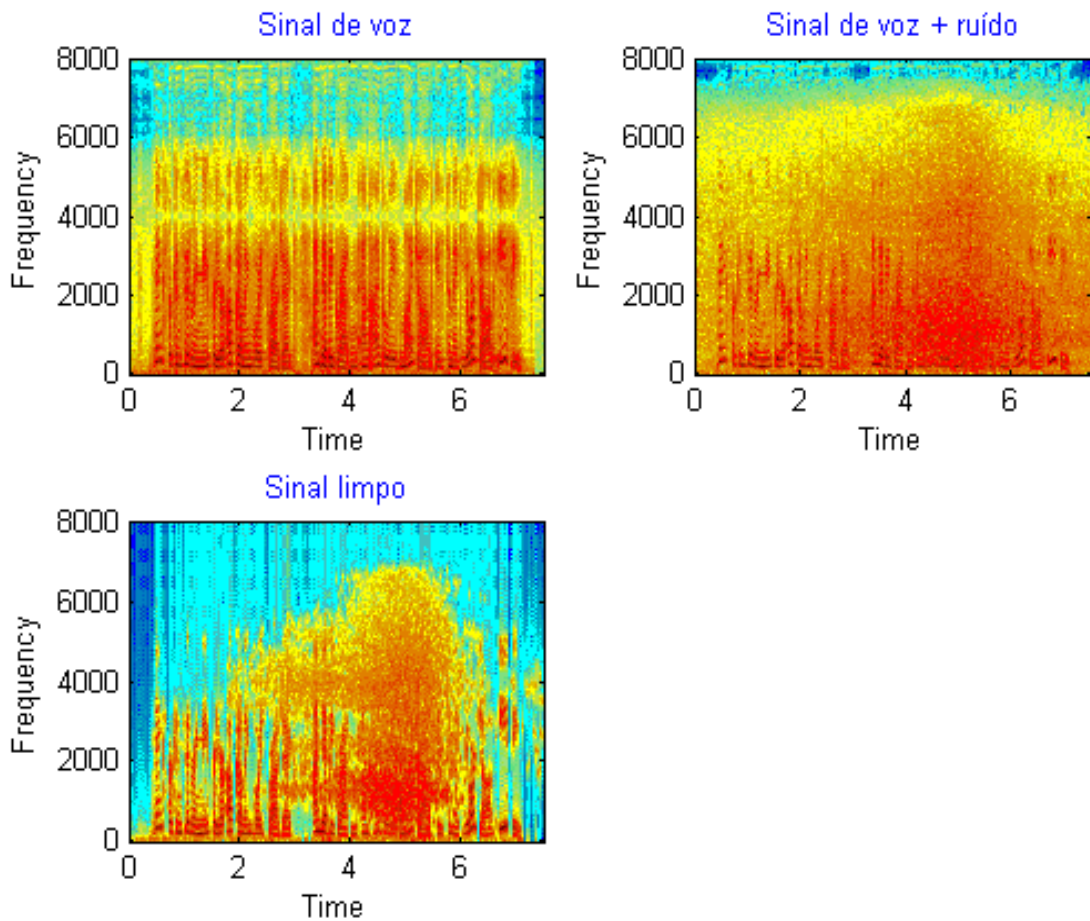


Figura 3.13: Espectrograma do sinal de voz com ruído de carro aplicando-se o algoritmo Filtro de Wiener

3.2.3 Subspace Tracking

A Figura 3.14 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal voz depois de ter sido aplicado o algoritmo de Subspace tracking.

A Figura 3.15 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.14.

O sinal de voz resultante do algoritmo subspace tracking, não apresenta o “musical noise”. Porém, novamente temos a introdução de componentes de altas frequências, o que ocasionou distorções no sinal de voz. O ruído de carro também não foi muito reduzido.

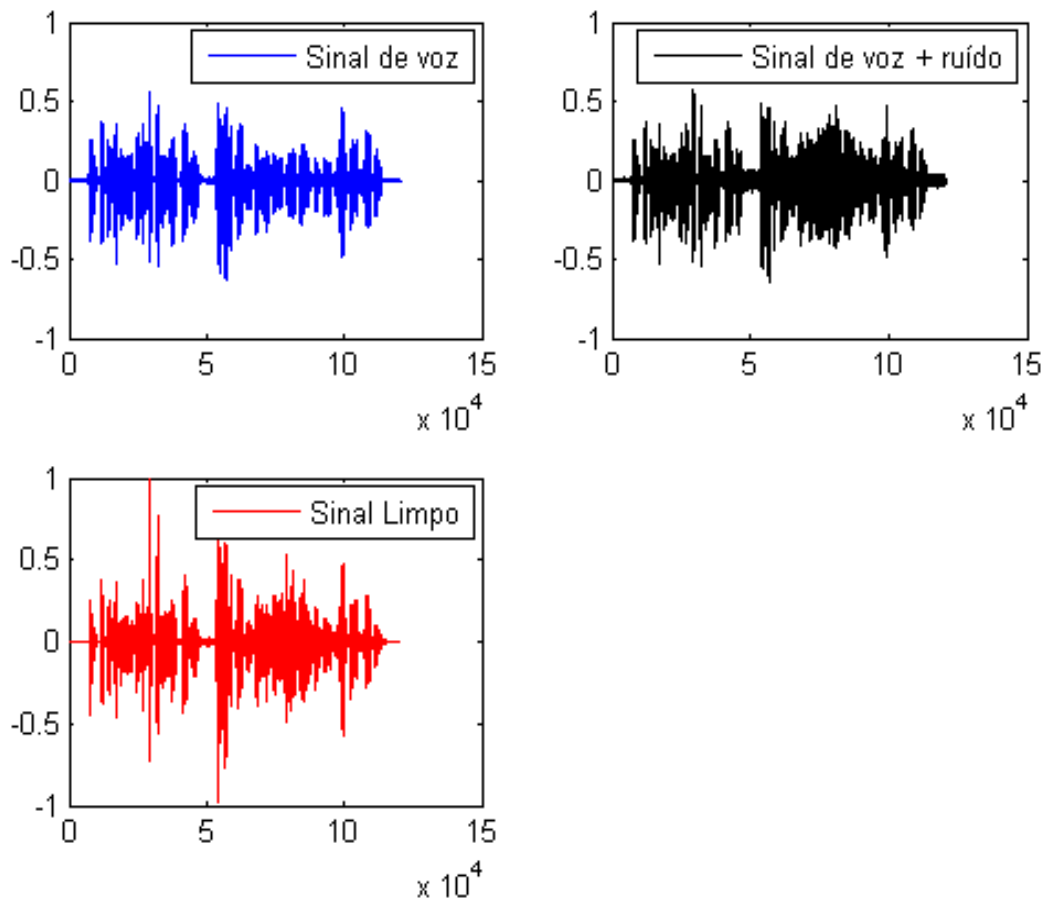


Figura 3.14: Subspace Tracking aplicado ao sinal de voz com ruído de carro

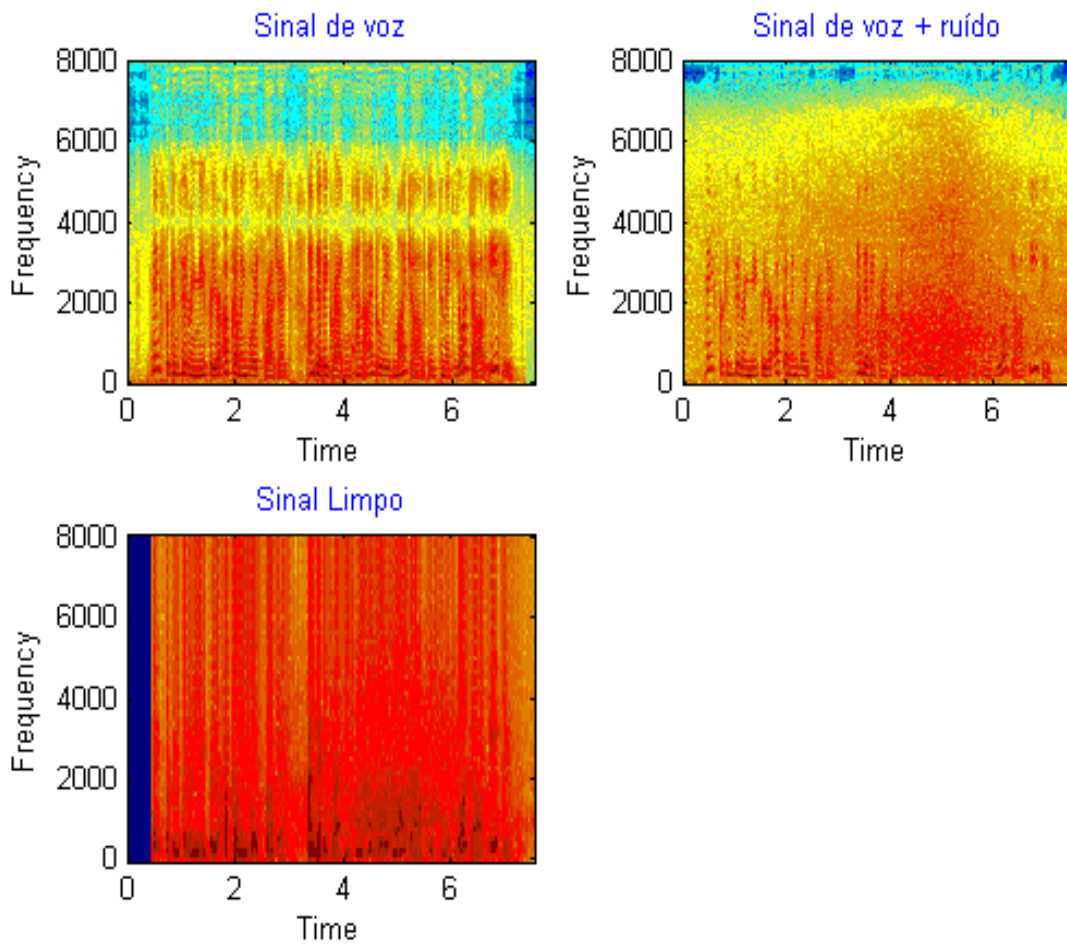


Figura 3.15: Espectrograma do sinal de voz com ruído de carro avião aplicando-se o algoritmo Subspace Tracking

3.2.4 Subspace Tracking com Transformada Wavelet

A Figura 3.16 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo subspace tracking com transformada wavelet.

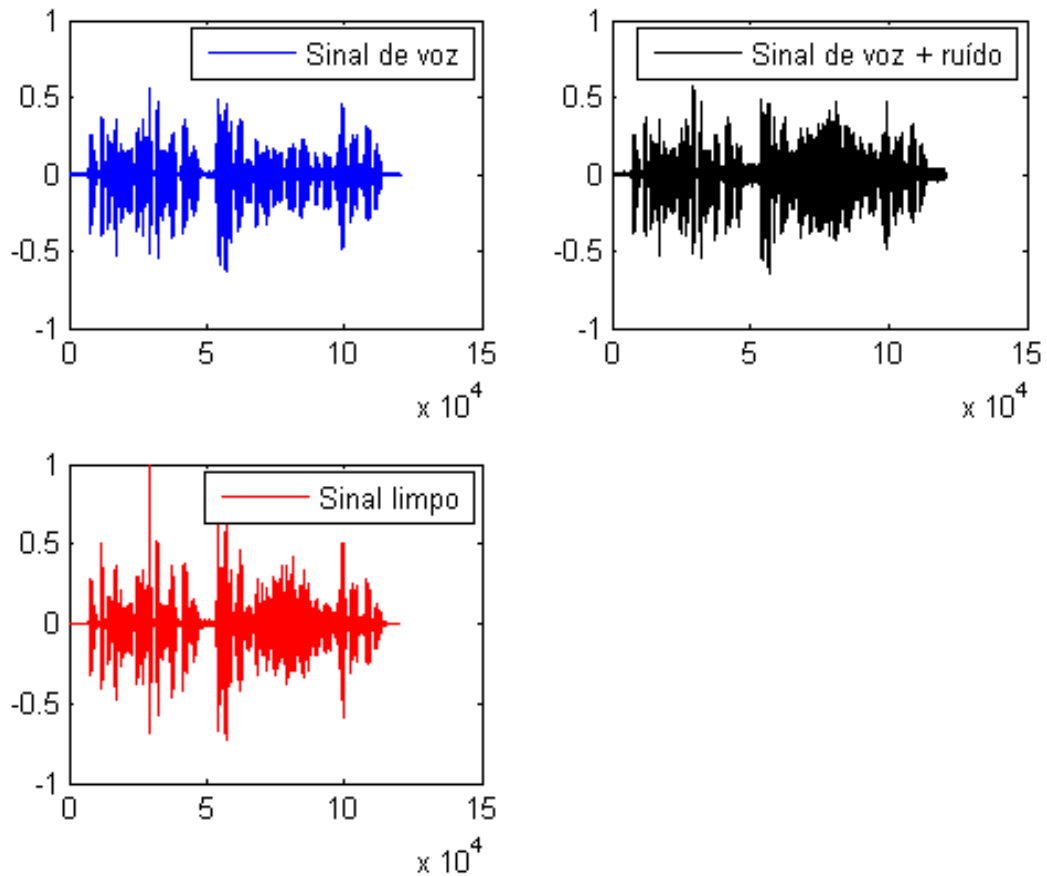


Figura 3.16: Subspace tracking com a transformada wavelet aplicado ao sinal de voz com ruído de carro

A Figura 3.17 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.16.

No sinal resultante do algoritmo subspace tracking usando a transformada wavelet mantém-se a inexistência do “musical noise”. O ruído ainda é forte no sinal de voz, mas menor do que sem a utilização da transformada wavelet. A introdução de componentes em alta frequência ainda existe, mas de forma mais branda. Logo, é possível concluir que a decomposição do sinal de voz usando a wavelet no algoritmo subspace tracking melhorou o seu desempenho, resultando numa maior redução do ruído e diminuindo as distorções no sinal de voz provocadas pelo surgimento de componentes em alta frequência.

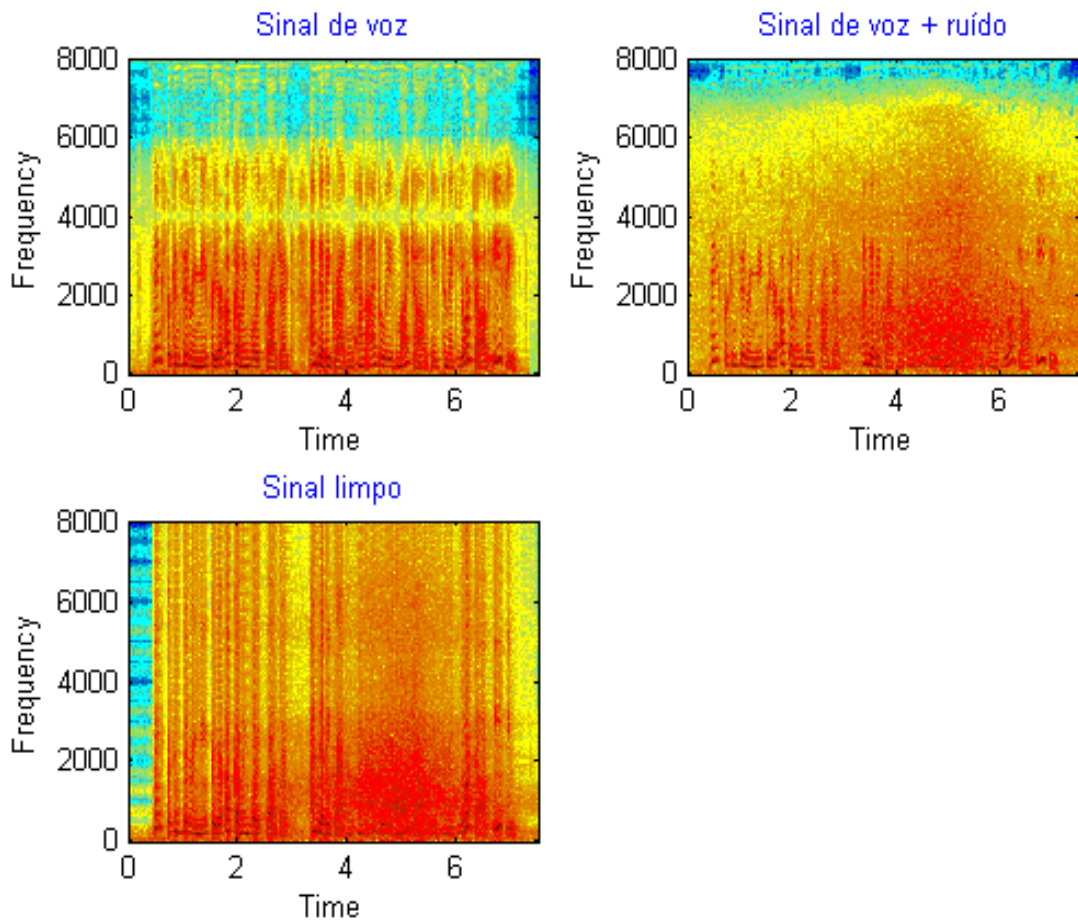


Figura 3.17: Espectrograma do sinal de voz com ruído de carro aplicando-se o algoritmo subpace tracking com a transformada wavelet

3.2.5 Comparação entre os Algoritmos

A Figura 3.18 apresenta os resultados obtidos com cada algoritmo para a redução de ruído de carro no sinal de voz.

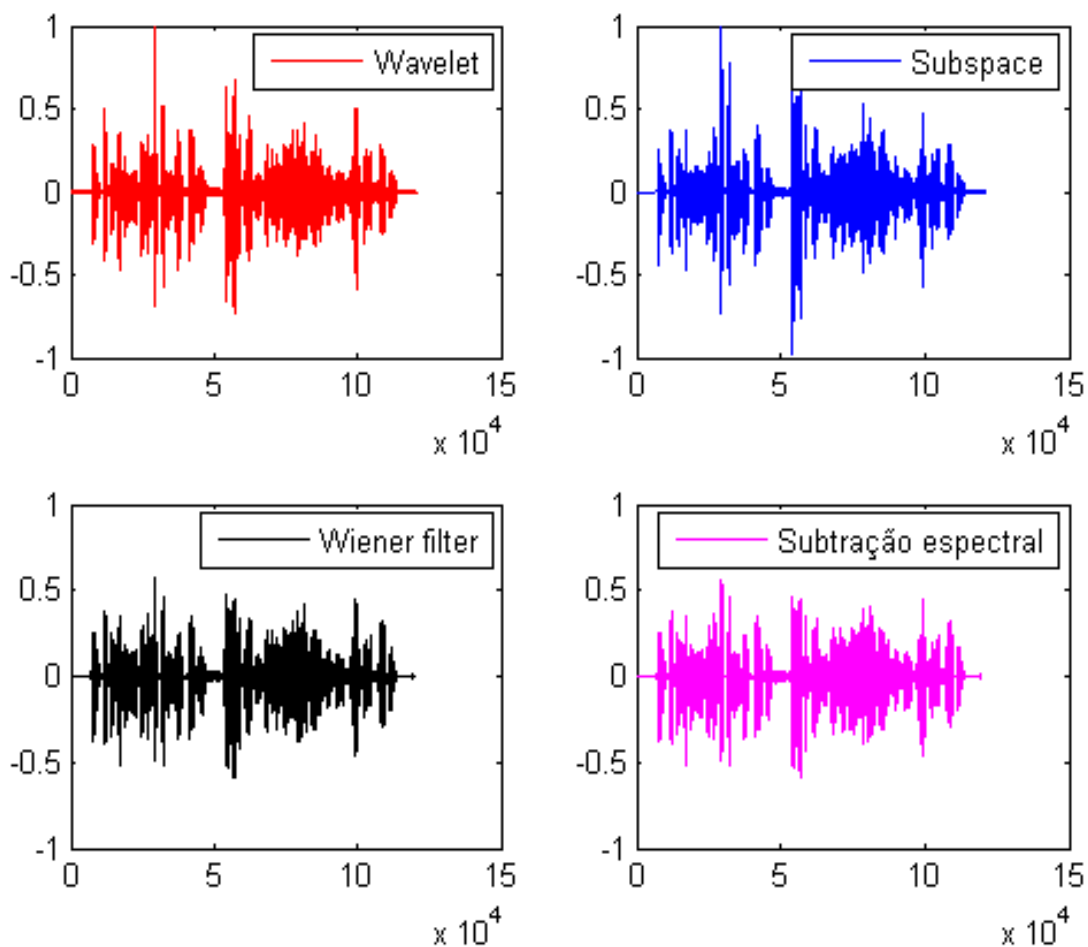


Figura 3.18: Comparação entre os algoritmos aplicados ao sinal de voz com ruído de carro

Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído de carro. Destacamos a presença do “musical noise” nos sinais resultantes do Filtro de Wiener e da Subtração Espectral, e o surgimento de componentes em alta frequência no Subspace Tracking e no mesmo com a transformada wavelet. Do ponto de vista subjetivo, os ouvintes não distinguiram entre o Filtro de Wiener e a Subtração espectral. No caso do Subspace Tracking ficou claro ao ouvir o sinal de voz que a utilização da transformada wavelet melhorou consideravelmente a qualidade do sinal resultante em relação ao do algoritmo original.

3.3 Ruído de Rua

Iremos discorrer acerca da performance de cada algoritmo em relação ao sinal de voz contaminado pelo ruído de rua.

3.3.1 Subtração Espectral

A Figura 3.19 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo da subtração espectral.

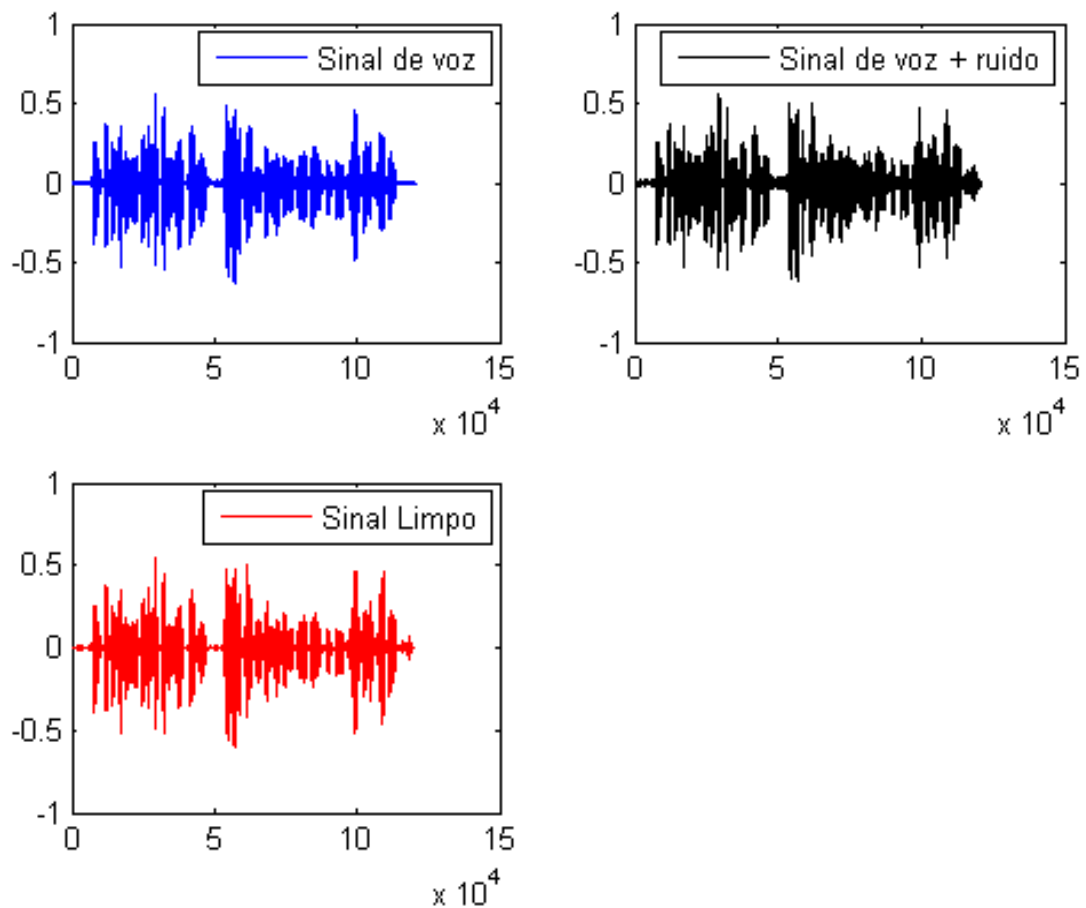


Figura 3.19: Subtração espectral aplicada ao sinal de voz com ruído de rua

A Figura 3.20 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.19.

Para o ruído de rua, o algoritmo de subtração espectral conseguiu eliminar um pouco do ruído, sem provocar o aparecimento do “musical noise”.

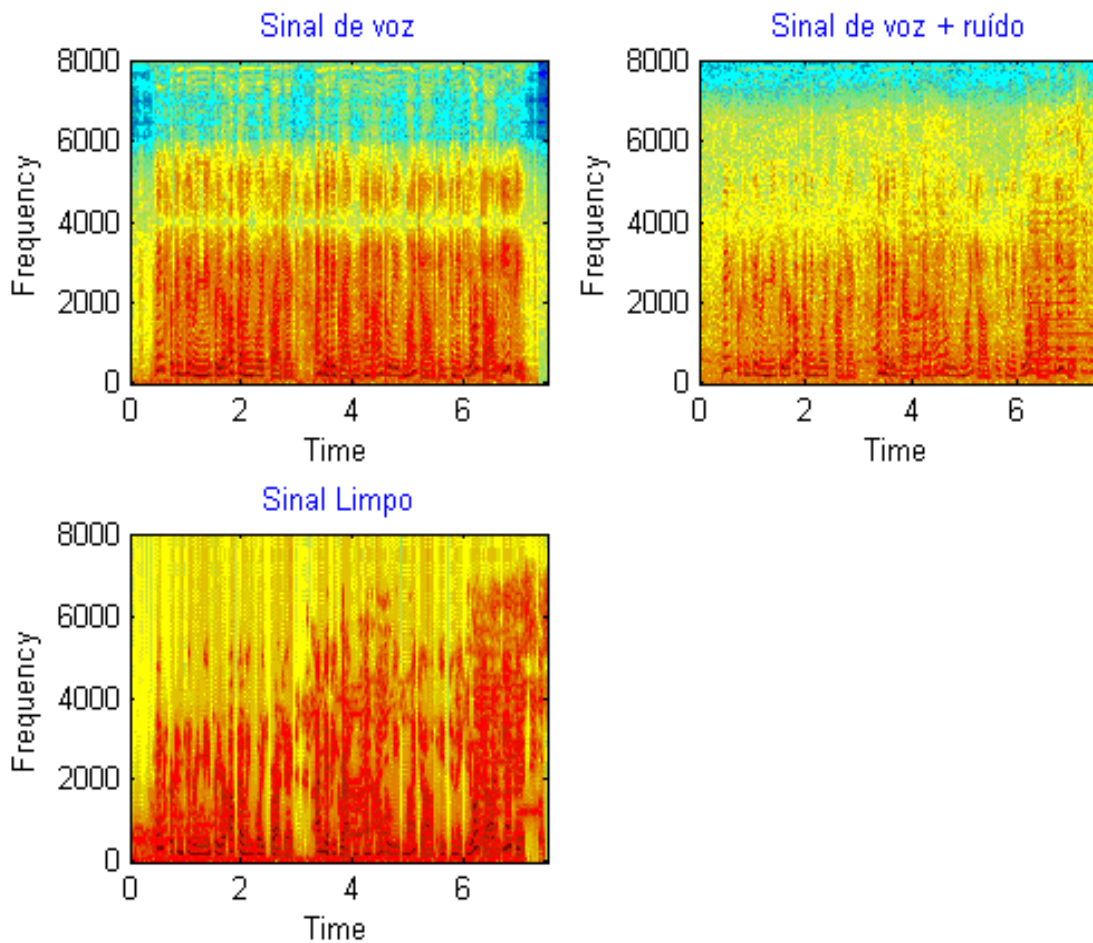


Figura 3.20: Espectrograma do sinal de voz com ruído de rua aplicando-se a subtração espectral

3.3.2 Filtro de Wiener

A Figura 3.21 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o Filtro de Wiener.

A Figura 3.22 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.21.

Semelhante ao algoritmo de subtração espectral, o Filtro de Wiener conseguiu evitar o surgimento do ruído de fundo. Nos dois casos, os sinais de voz paralelos ao sinal original não foram eliminados como esperado, pois os algoritmos não conseguem distinguir qual dos sinais de voz corresponde a ruído.

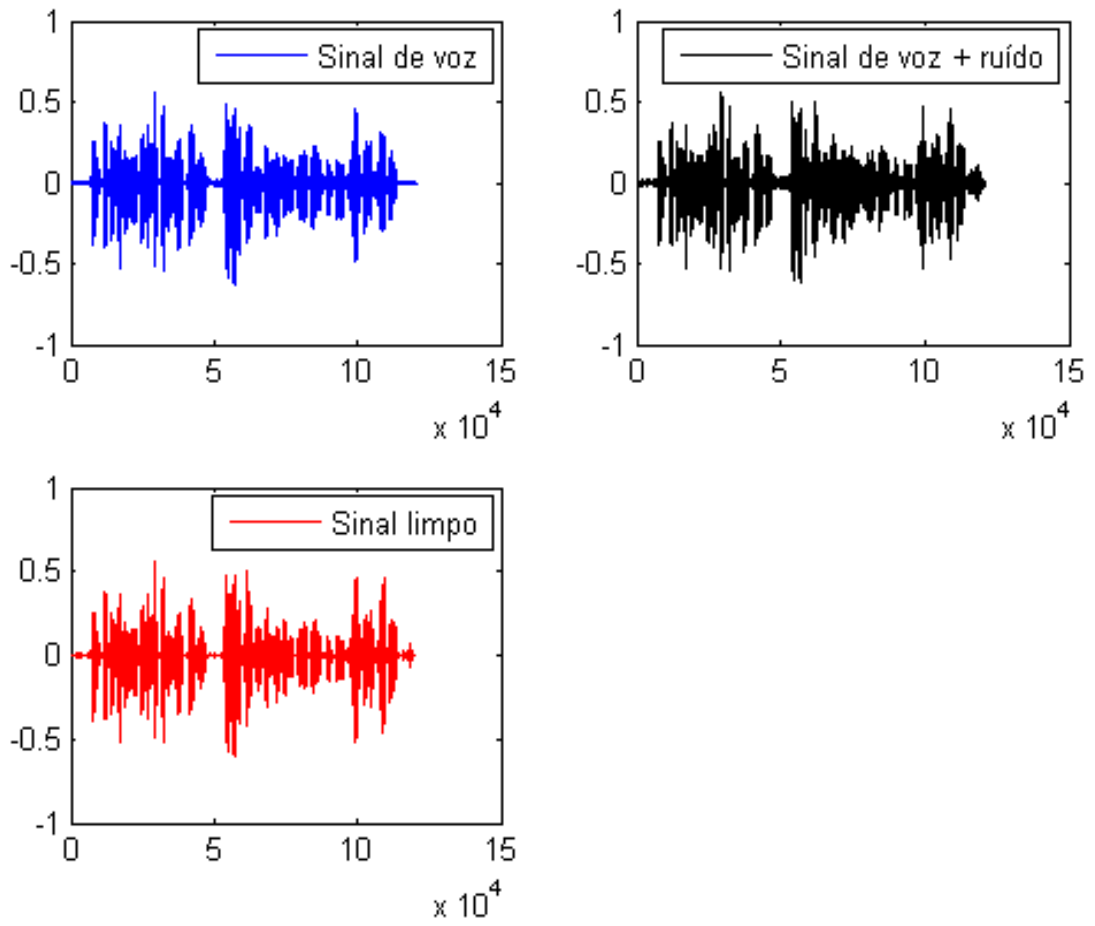


Figura 3.21: Filtro de Wiener aplicado ao sinal de voz com ruído de rua

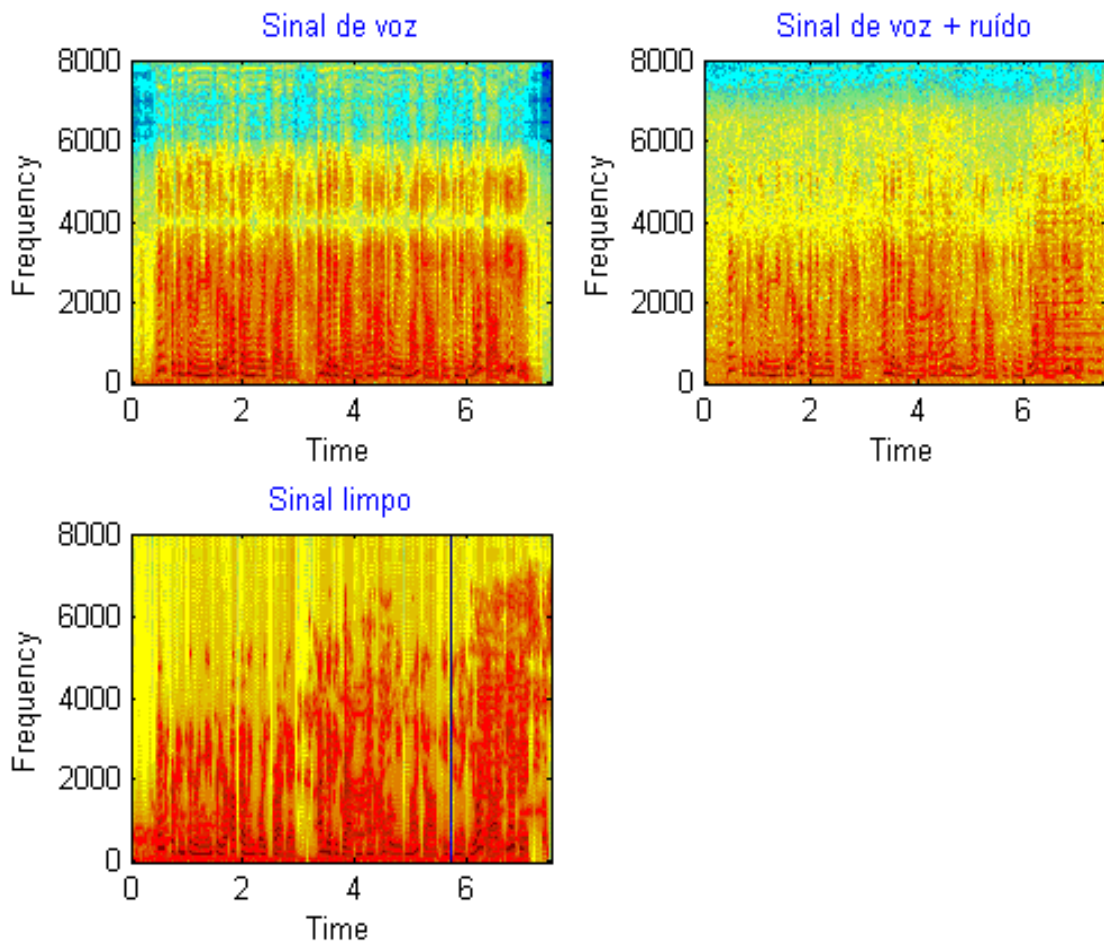


Figura 3.22: Espectrograma do sinal de voz com ruído de rua aplicando-se o Filtro de Wiener

3.3.3 Subspace Tracking

A Figura 3.23 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo subspace tracking.

A Figura 3.24 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.23.

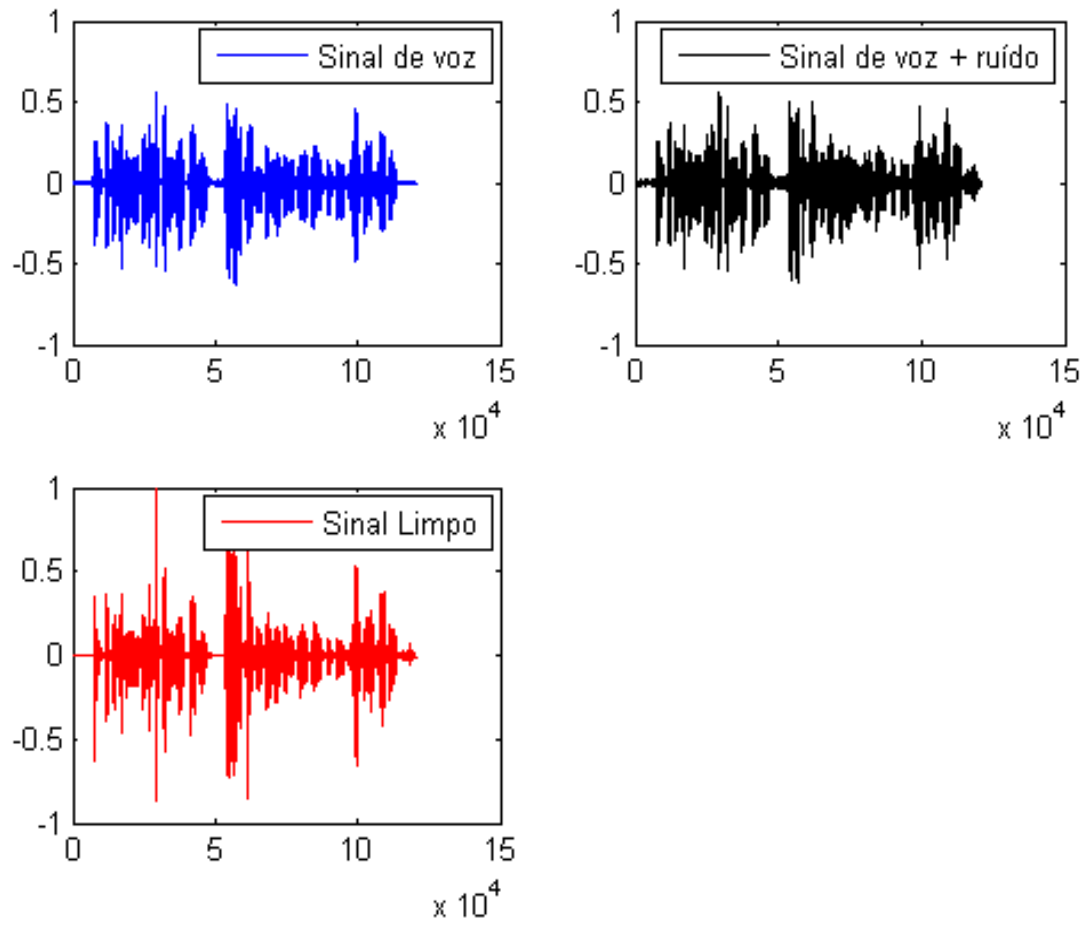


Figura 3.23: Subspace Tracking aplicado ao sinal de voz com ruído de rua

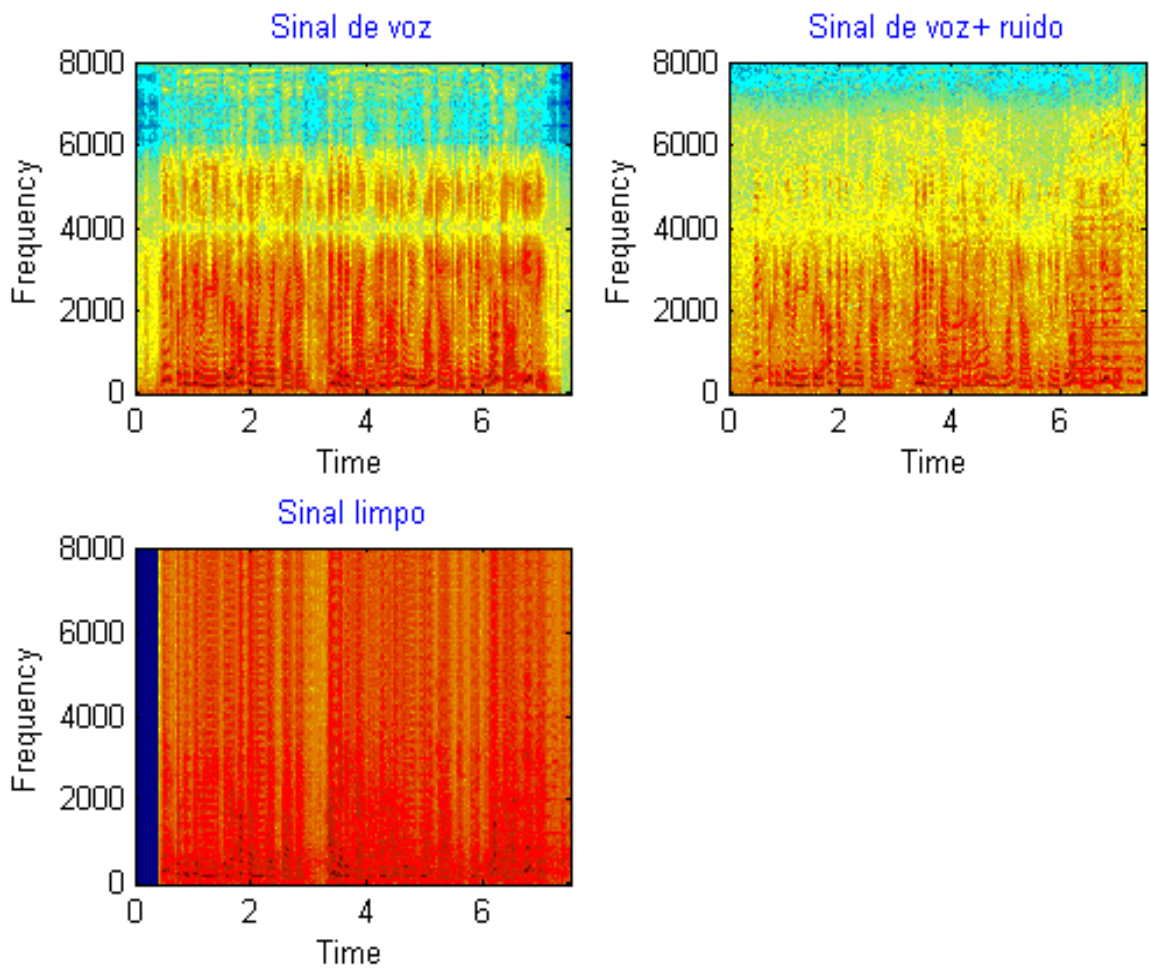


Figura 3.24: Espectrograma do sinal de voz com ruído de rua aplicando-se o algoritmo Subspace Tracking

No subspace tracking, o sinal de voz não apresenta o “musical noise”. Porém, novamente temos a introdução de componentes em alta frequência no sinal de voz, o que ocasionou um sinal de voz distorcido.

3.3.4 Subspace Tracking com Transformada Wavelet

A Figura 3.25 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo subspace tracking com transformada wavelet.

A Figura 3.26 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.25.

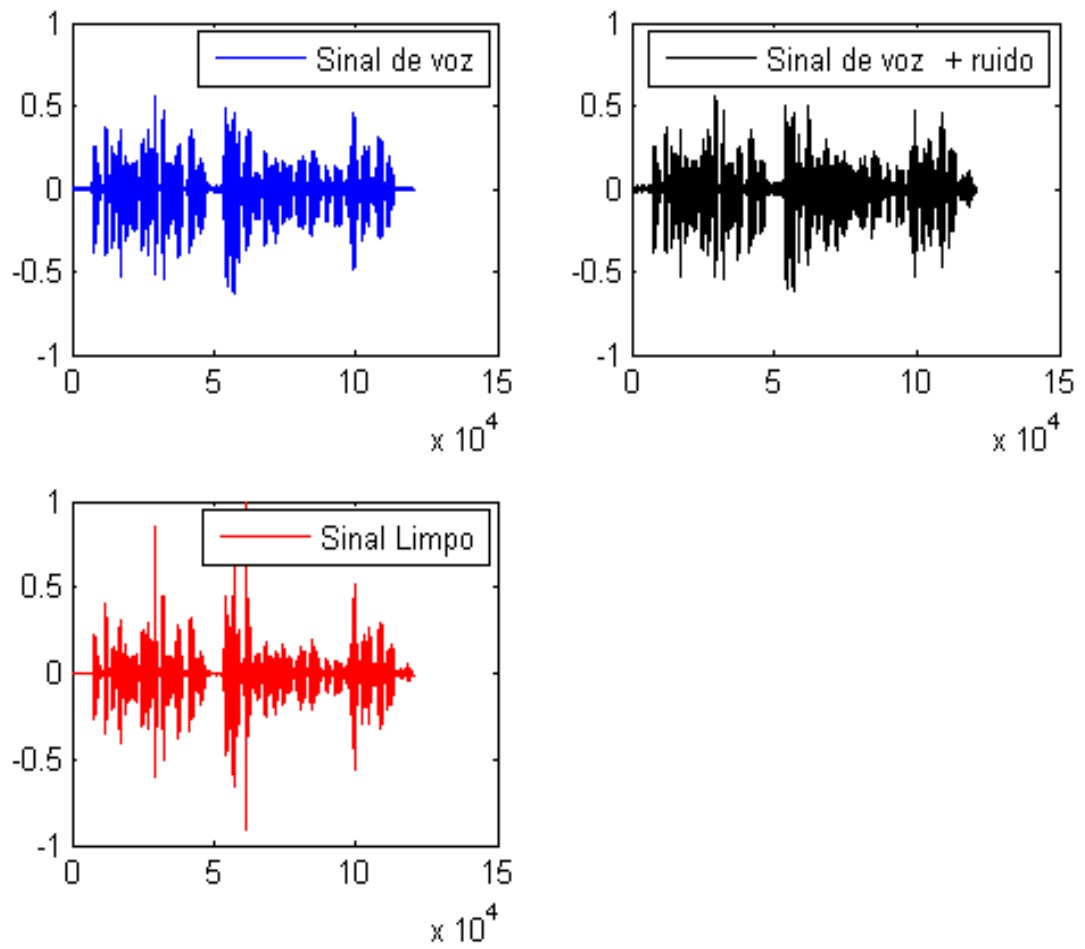


Figura 3.25: Subspace tracking com a transformada wavelet aplicado ao sinal de voz com ruído de rua

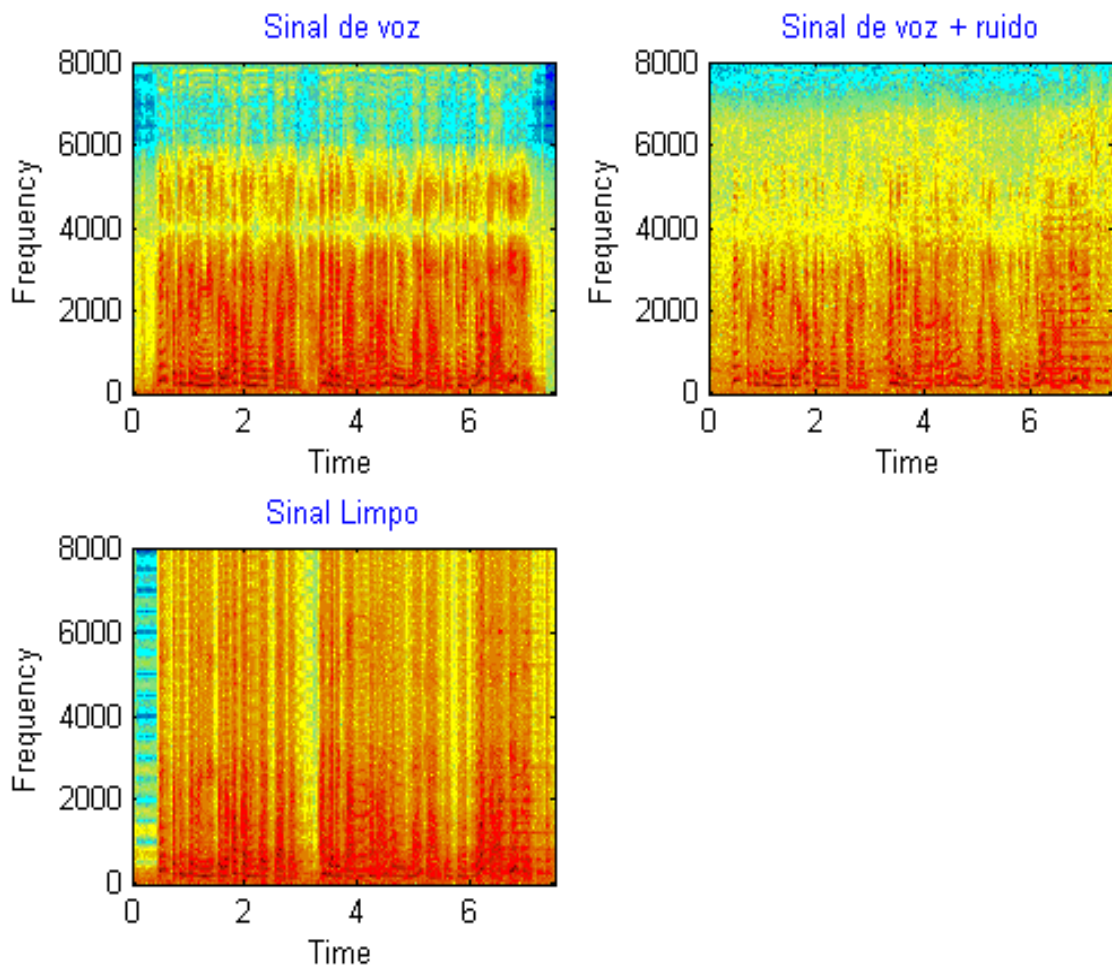


Figura 3.26: Espectrograma do sinal de voz com ruído de rua aplicando-se o algoritmo de Subspace Tracking com Transformada Wavelet

No algoritmo subspace tracking usando a transformada wavelet mantém-se a inexistência do “musical noise”. A introdução de componentes em alta frequência ainda existe, mas de forma mais branda. Logo, é possível concluir que a decomposição do sinal de voz usando a wavelet no algoritmo subspace tracking reduziu mais acentuadamente o ruído, introduzindo menos distorções no sinal de voz.

3.3.4 Comparação entre os Algoritmos

A Figura 3.27 apresenta os resultados obtidos com cada algoritmo para a redução de ruído de rua no sinal de voz.

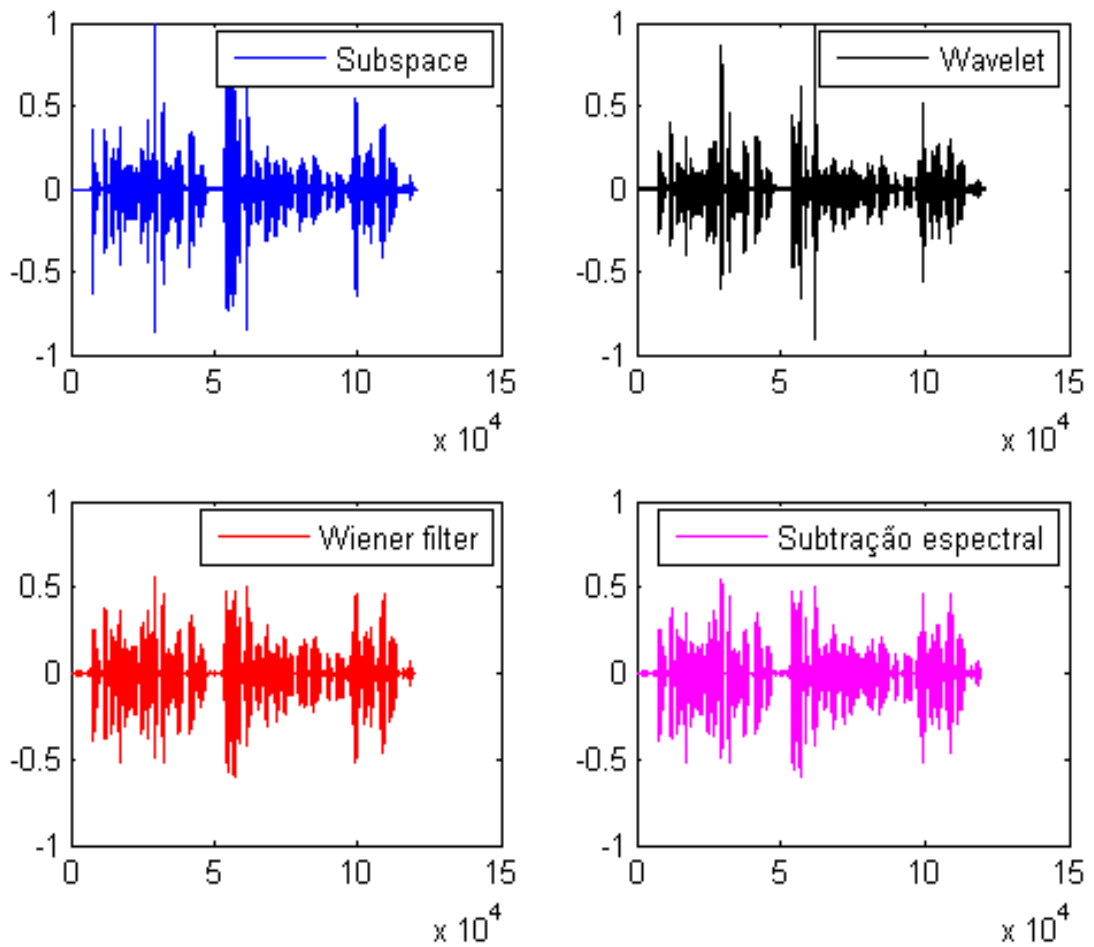


Figura 3.27: Comparação entre os algoritmos aplicados ao sinal de voz com ruído de rua

Pelo aspecto objetivo, nenhum dos quatro algoritmos suprimiu completamente o ruído, pois se observa a existência de vozes paralelas ao sinal de voz de interesse, que não são eliminadas pelo algoritmo. Ainda ocorre a introdução de componentes em alta frequência, antes não existentes, no sinal resultante do algoritmo subspace tracking e no mesmo com a transformada wavelet.

3.4 Ruído Branco

Iremos discorrer acerca da performance de cada algoritmo em relação ao sinal de voz contaminado por ruído branco.

3.4.1 Subtração Espectral

A Figura 3.28 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal voz de com ruído depois de ter sido aplicado o algoritmo da subtração espectral.

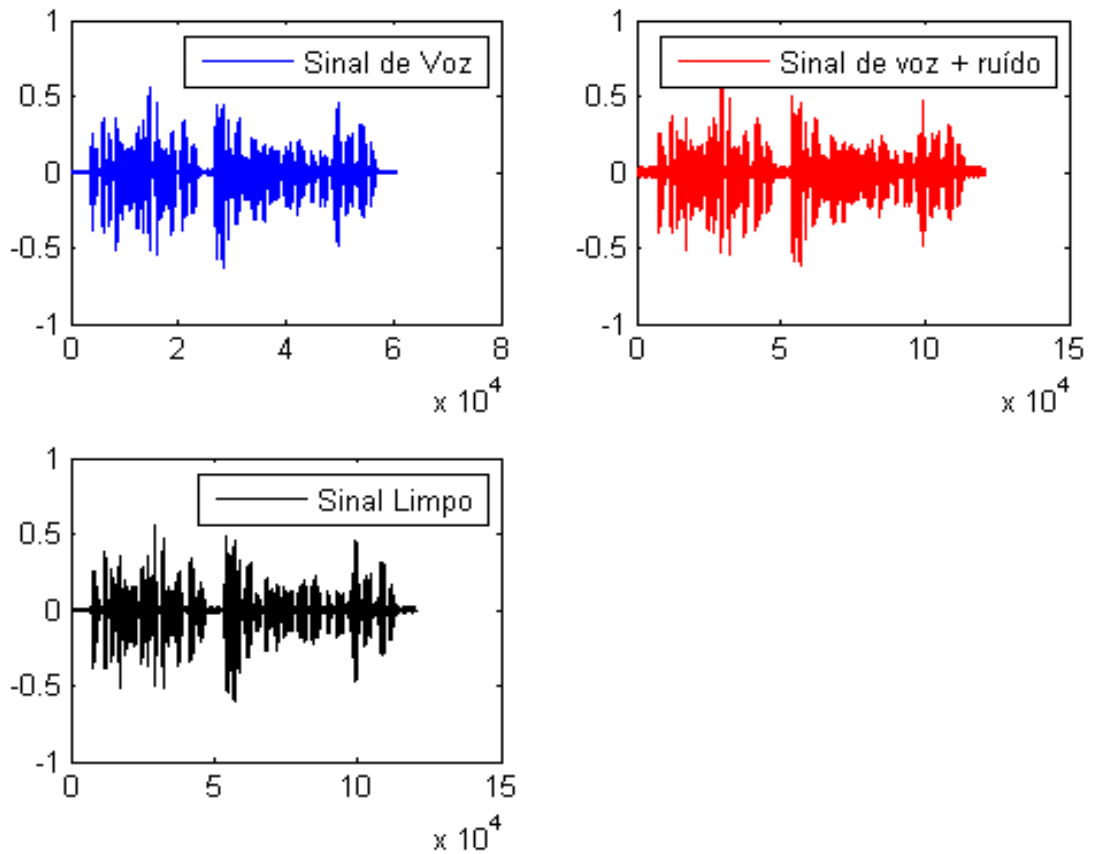


Figura 3.28: Subtração espectral aplicada ao sinal de voz com ruído branco

A Figura 3.29 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.28.

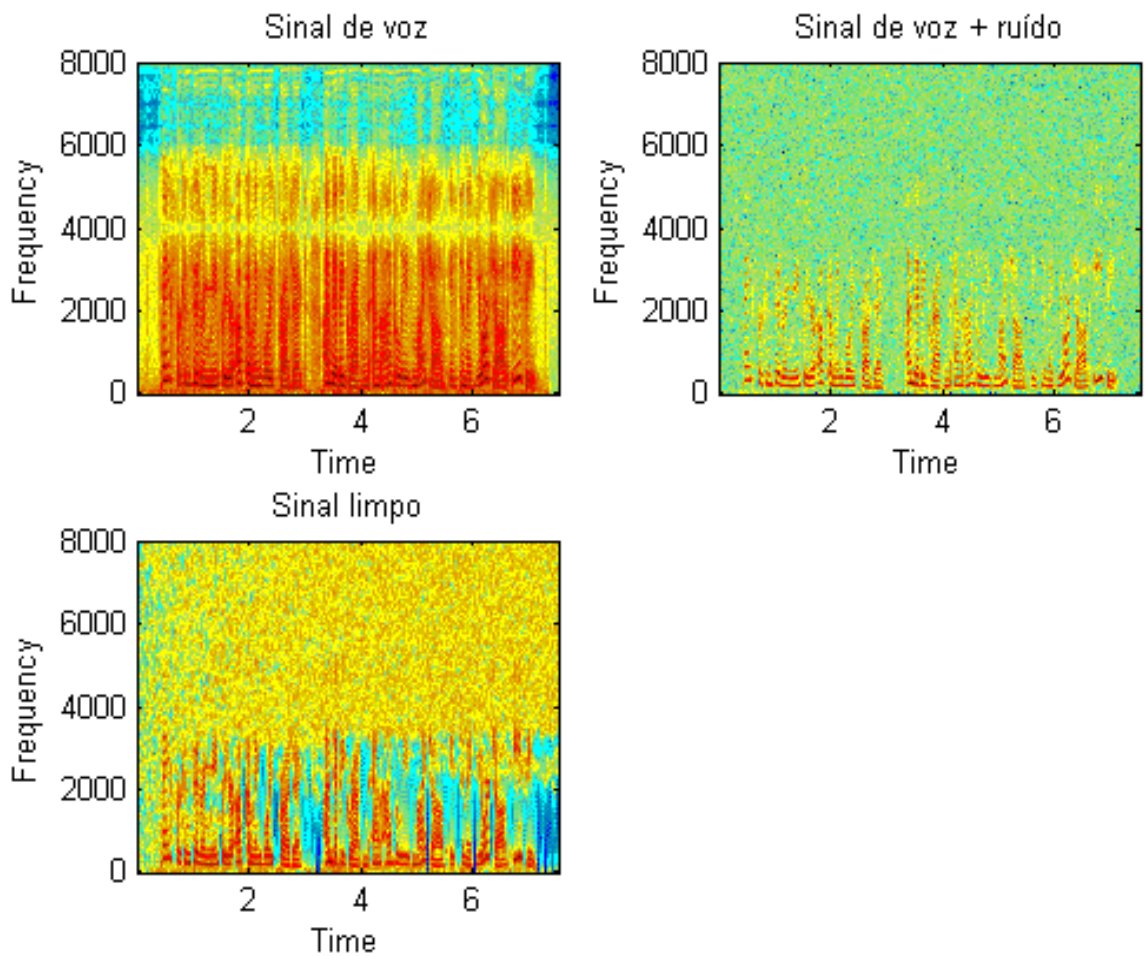


Figura 3.29: Espectrograma do sinal de voz com ruído branco aplicando-se o algoritmo de Subtração Espectral

O algoritmo de subtração espectral conseguiu reduzir o ruído. Entretanto, temos presente o ruído de fundo (“musical noise”) no sinal de voz resultante.

Pelos espectrogramas vemos que o ruído concentra-se em alta frequência. As informações em componentes de baixa frequência foram recuperadas, porém as de alta frequência permanecem ainda com ruído na forma de “musical noise”.

3.4.2 Filtro de Wiener

A Figura 3.30 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o Filtro de Wiener.

A Figura 3.31 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.30.

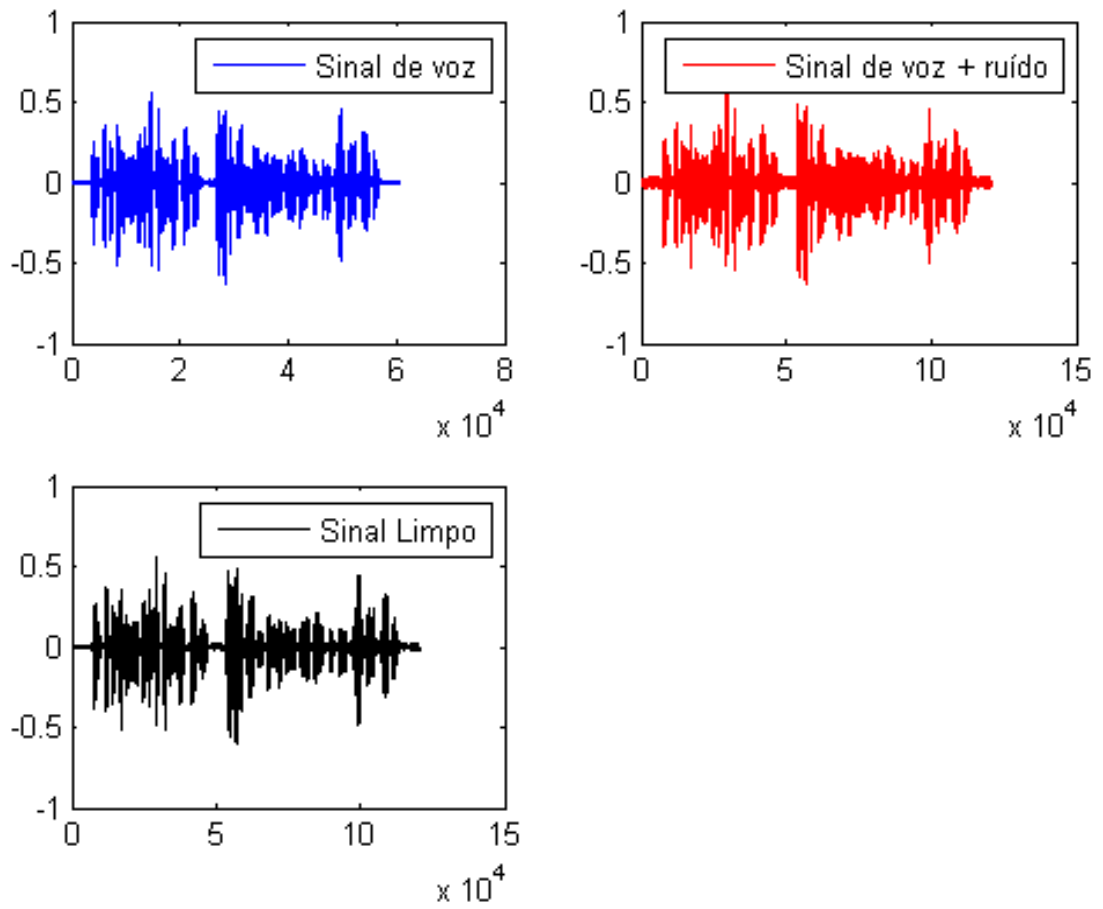


Figura 3.30: Filtro de Wiener aplicado ao sinal de voz com ruído branco

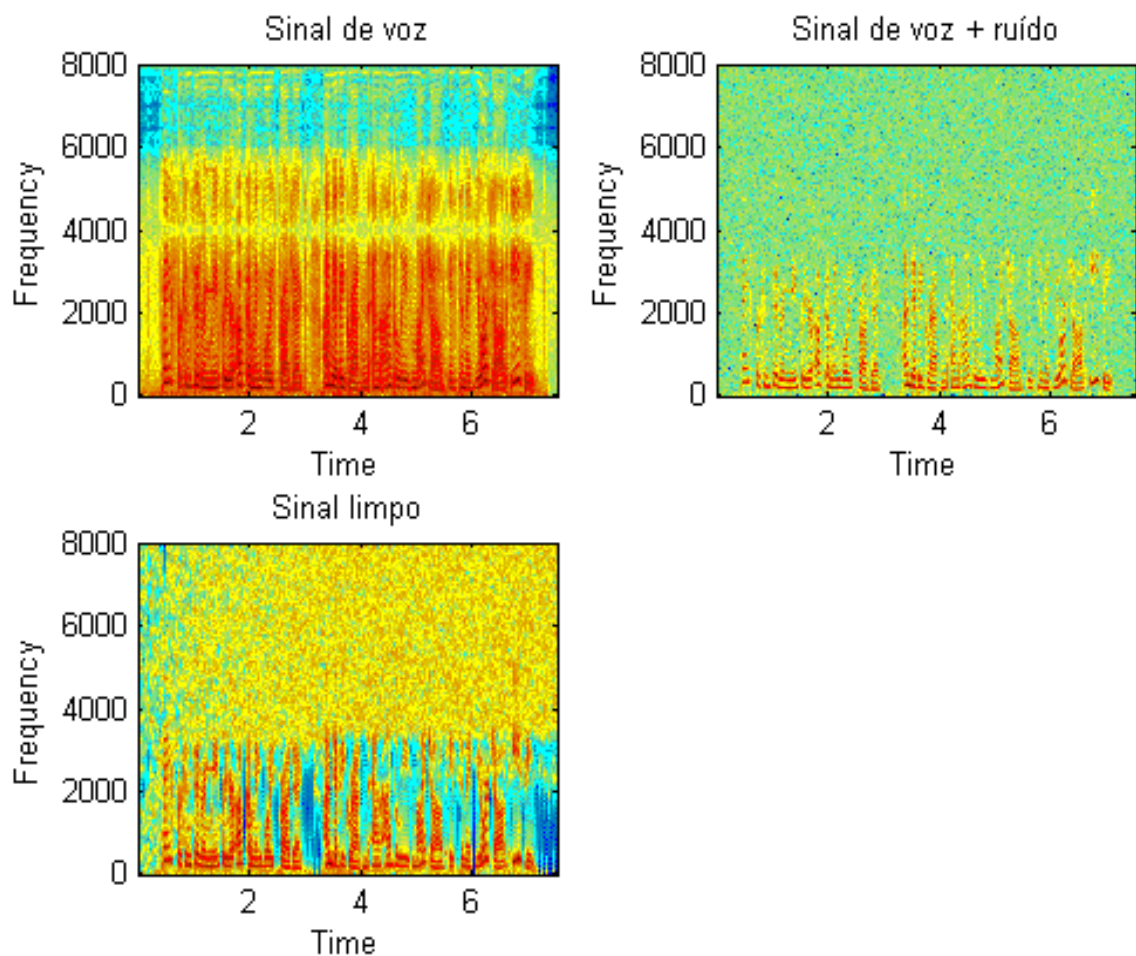


Figura 3.31: Espectrograma do sinal de voz com ruído branco aplicando-se o Filtro de Wiener

Semelhante ao algoritmo de subtração espectral, o Filtro de Wiener não conseguiu evitar o surgimento do ruído de fundo.

Pelos espectrogramas vemos que o ruído concentra-se em alta frequência. As informações em componentes de baixa frequência foram recuperadas, porém as de alta frequência permanecem ainda com ruído na forma de “musical noise”. Comparando com a subtração espectral, este apresenta um pouco mais de informação da voz em alta frequência do que o filtro de Wiener.

3.4.3 Subspace Tracking

A Figura 3.32 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo Subspace Tracking.

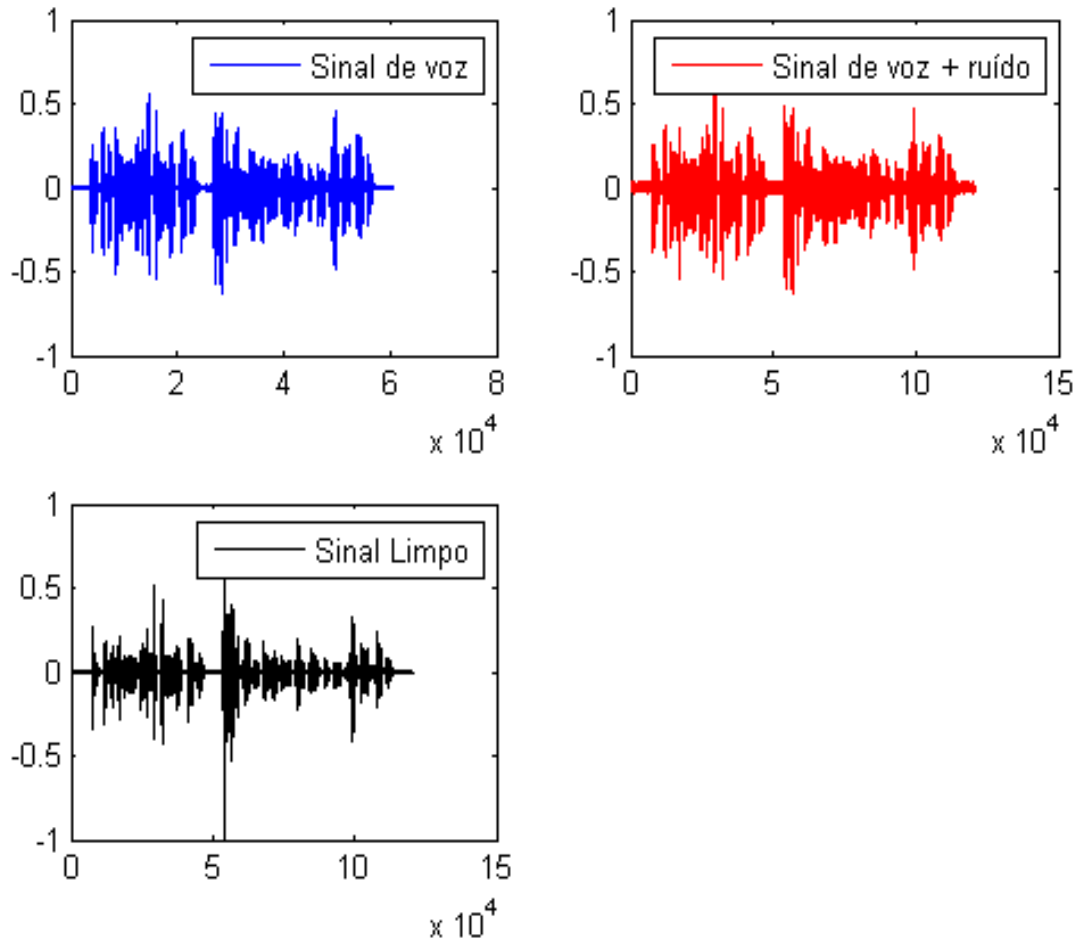


Figura 3.32: Subspace tracking aplicado ao sinal de voz com ruído branco

A Figura 3.33 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.32.

No algoritmo subspace tracking, o sinal de voz resultante não apresenta o “musical noise”. Porém, as componentes em alta frequência não foram recuperadas corretamente, de modo que, o sinal de voz ficou distorcido e abafado.

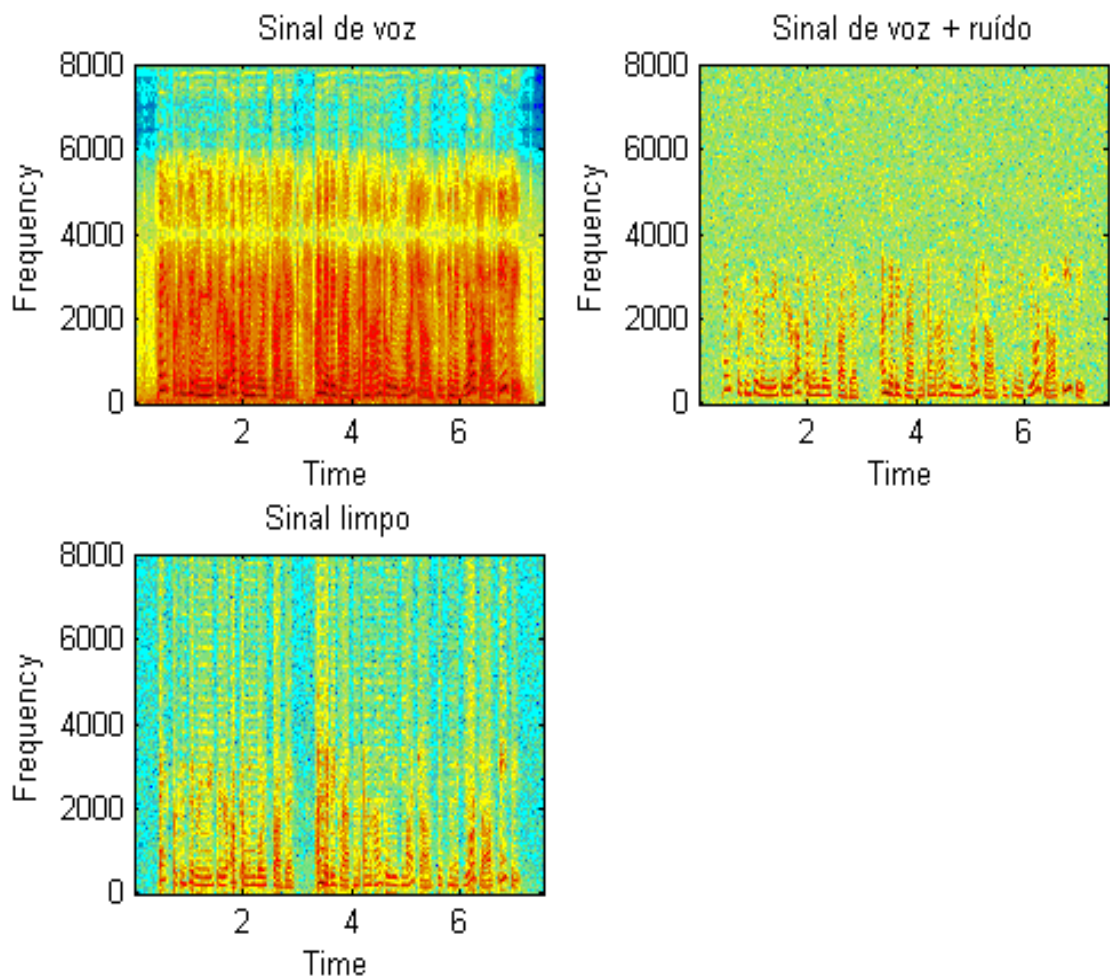


Figura 3.33: Espectrograma do sinal de voz com ruído aplicando-se o algoritmo de Subspace Tracking

3.4.4 Subspace Tracking com Transformada Wavelet

A Figura 3.34 apresenta, respectivamente, os gráficos do sinal antes da introdução do ruído, após a introdução deste, e do sinal de voz depois de ter sido aplicado o algoritmo subspace tracking com transformada wavelet.

A Figura 3.35 apresenta os gráficos dos espectrogramas dos sinais da Figura 3.34.

No algoritmo subspace tracking usando a transformada wavelet, mantém-se a inexistência do “musical noise”. O ruído ainda é forte no sinal de voz, mas menor do que sem a utilização da transformada wavelet. A perda de componentes em alta frequência ainda existe, mas também é menos acentuada, resultando em menores distorções no sinal de voz.

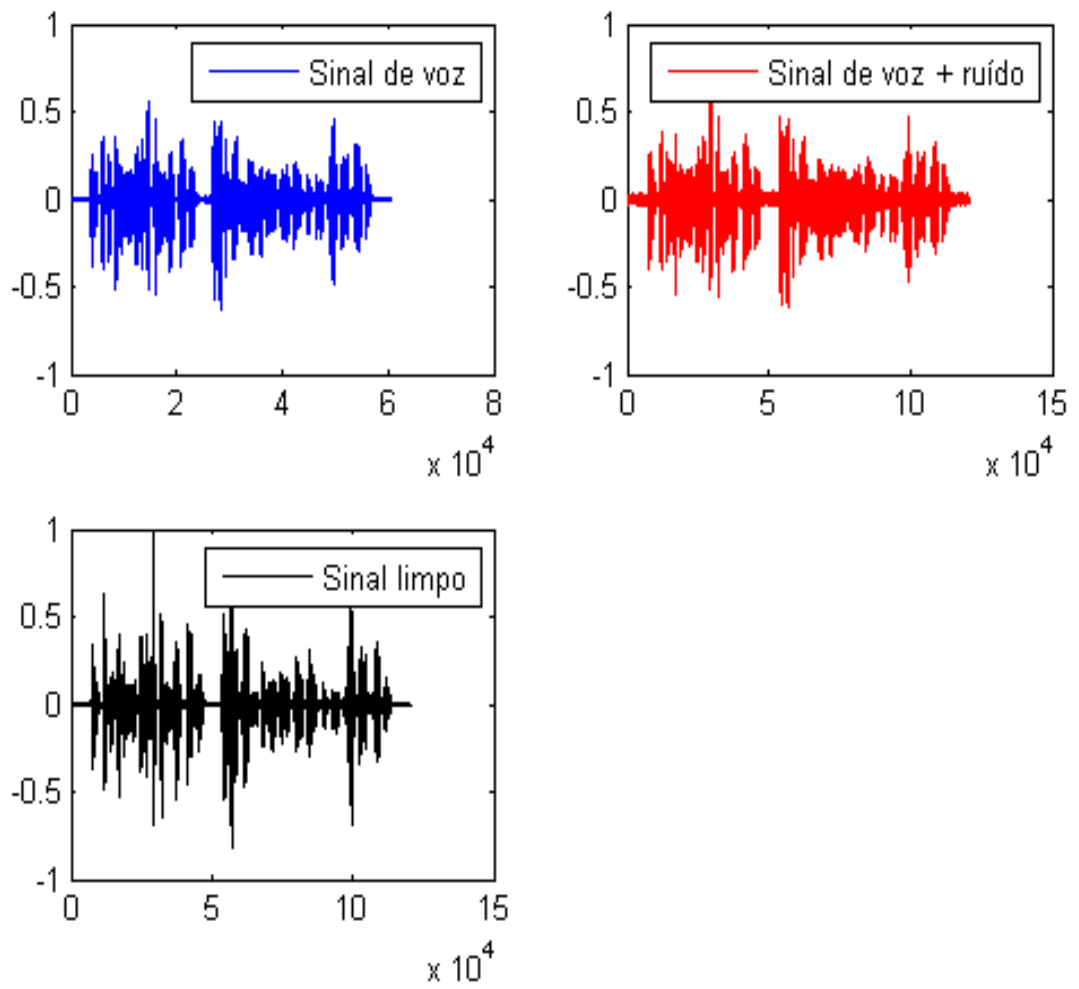


Figura 3.34: Subspace Tracking com transformada Wavelet aplicado ao sinal de voz com ruído branco

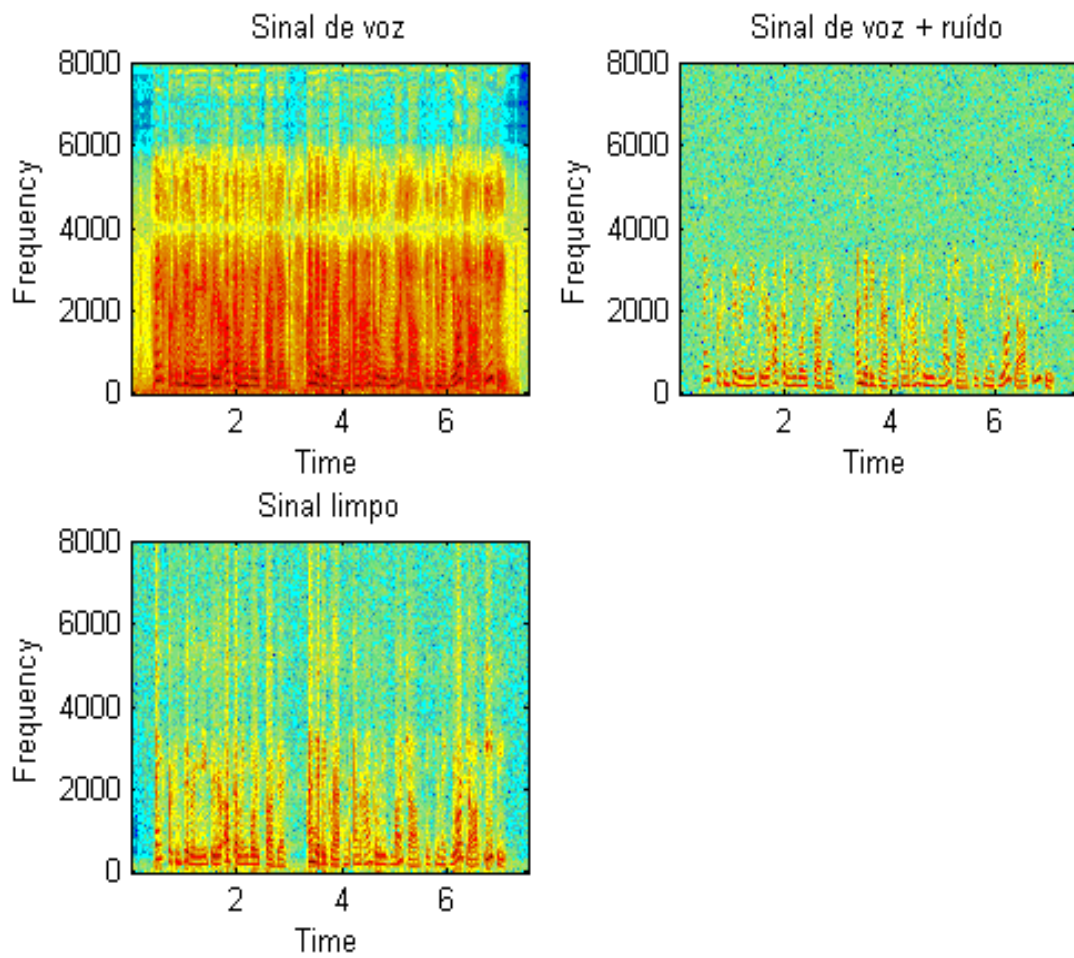


Figura 3.35: Espectrograma do sinal de voz com ruído branco aplicando-se o algoritmo subspace tracking com transformada wavelet

3.4.5 Comparação entre os Algoritmos

A Figura 3.36 apresenta os resultados obtidos com cada algoritmo para a redução de ruído branco no sinal de voz.

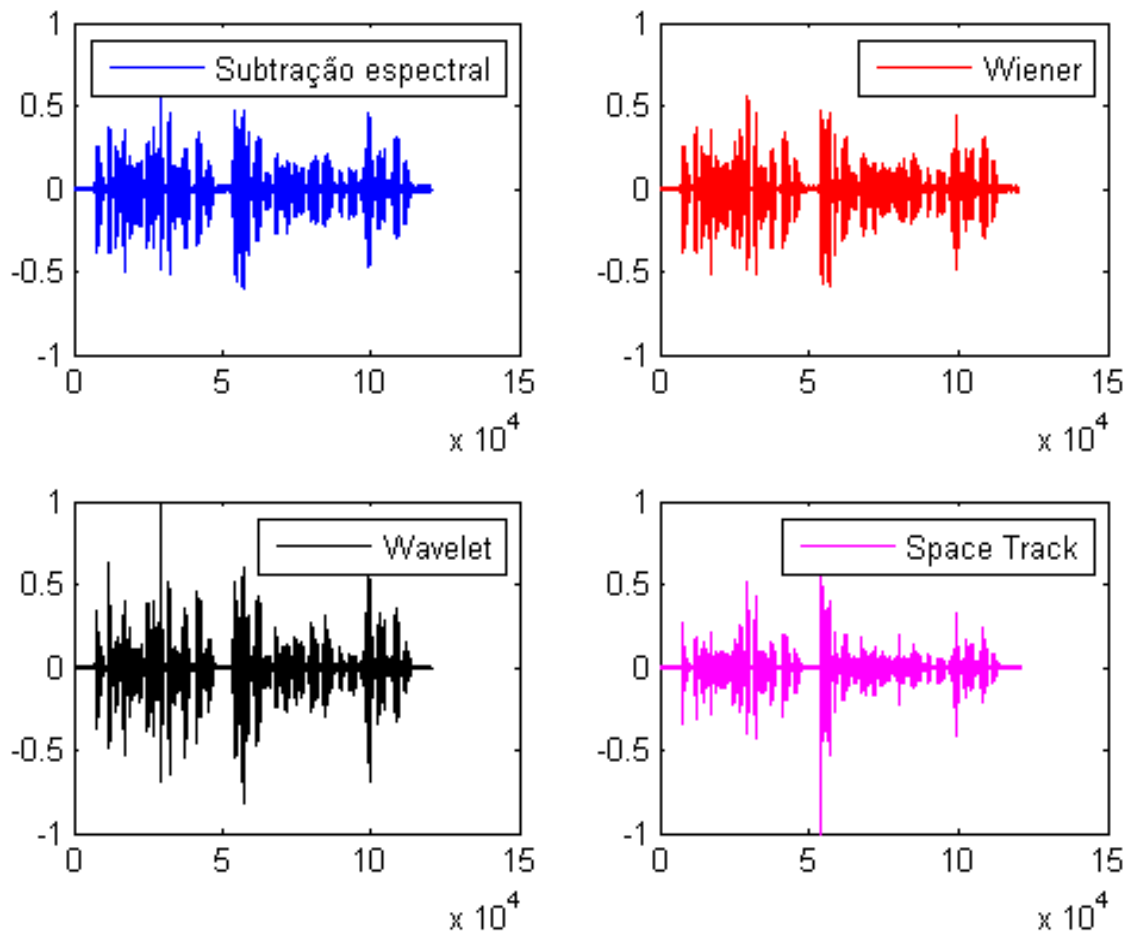


Figura 3.36: Comparação entre os algoritmos aplicados ao sinal de voz com ruído branco

Destacamos a presença do “musical noise” no Filtro de Wiener e Subtração Espectral, e a perda de componentes em alta frequência no Sbspace Tracking e no mesmo com a transformada wavelet. Do ponto de vista do aspecto subjetivo, o Filtro de Wiener foi o que obteve o melhor resultado na redução do ruído branco. Ficou claro, mais uma vez, que a utilização da transformada wavelet melhorou consideravelmente o resultado do algoritmo Subspace Tracking.

4 CONCLUSÕES

O objetivo principal deste trabalho é demonstrar que é possível fazer uso de algoritmos para implementar métodos de supressão de ruído em sinais que sofreram degradação de vários tipos. Este objetivo foi alcançado, já que pudemos observar, através dos gráficos e da avaliação dos arquivos de áudio, uma diminuição considerável do ruído no sinal de voz.

Os quatro algoritmos implementados não conseguiram reduzir completamente os ruídos utilizados nos testes (ruído de avião, carro, rua e branco). Porém, isso não significa que os algoritmos são ineficientes. Os resultados dependem também da intensidade do ruído aplicado e dos parâmetros iniciais em cada algoritmo. Devido à similaridade dos algoritmos de Wiener e Subtração Espectral, seus respectivos resultados foram muito semelhantes em todos os casos. Só foi possível discernir o melhor resultado ouvindo os arquivos de áudio (e não através de uma avaliação gráfica dos sinais), o que torna a análise subjetiva e varia para cada ouvinte.

Pudemos constatar que existe uma considerável diferença no desempenho dos algoritmos de Subspace Tracking com o uso da transformada Wavelet e sem o uso desta. A decomposição do sinal em bandas utilizando a Wavelet de Daubechies contribuiu para uma redução maior do ruído. Contudo, não foi suficiente para evitar a modificação de componentes em alta frequência fazendo com o sinal de voz ficasse abafado e/ou distorcido.

Para melhorar ainda mais o desempenho dos algoritmos, diversificações nas condições iniciais dos algoritmos podem ser feitas: utilizar melhores estimativas das estatísticas do ruído, selecionar de forma mais criteriosa os parâmetros dos algoritmos como o número de decomposições na transformada Wavelet e as constantes das recursões (α e β), entre outros. Pode-se também aplicar a transformada Wavelet no Filtro de Wiener e na Subtração Espectral a fim de amenizar a presença do “musical noise” naqueles.

5 REFERÊNCIAS

- [1] J. F. Wang, C. H. Yanh e K. H. Chang, “Subspace tracking for speech enhancement in car noise environments”, pp. 789-792, ICASSP 2004.
- [2] Y. Ephraim e H. L. Van Trees, “A signal subspace approach for speech enhancement”, IEEE Transactions on Speech and Audio Processing, vol 3, no 4, July 1995.
- [3] R. McAulay e M. Malpass, “Speech enhancement using a soft-decision noise suppression filter”, IEEE Transactions on Speech and Audio Processing, vol ASSP-28, no 2, April 1980.
- [4] S.F.Boll ,”Supression of acoustic noise in speech using spectral subtraction” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol.ASSP-29, pp. 113-120, Apr. 1979.
- [5] S. F. Boll, “Speech enhancement in the 1980’s: Noise suppression with pattern matching,” in Advances in Speech Signal Processing (S. Furui and M. M. Sondhi, Eds.), New York: Marcel Dekker, 1992.
- [6]] J. F. Wang, C. H. Yanh and K. H. Chang, “Multiband Subspace Tracking Speech Enhancement for In-Car Human Computer Speech Interaction Mean Square Error Estimator”, Journal of Information Science and Engineering, no 22, pp. 1093-1107, 2006.
- [7] Regis Rossi A. Faria, “Aplicação de Wavelets na análise de gestos Musicais”, Tese de Mestrado, Unicamp, São Paulo, 1997.
- [8] J. Benesty, S. Makino and J. Chen (Eds.), “Speech Enhancement“, Springer-Verlag, 2005.