

UNIVERSIDADE DE SÃO PAULO ESCOLA POLITÉCNICA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA

FILTRO PARA SEPARAR PERFUSÃO OU RESPIRAÇÃO EM  
IMAGENS DE TOMOGRAFIA POR IMPEDÂNCIA ELÉTRICA

Caetano Belda Martinez

SÃO PAULO

- 2005 -

UNIVERSIDADE DE SÃO PAULO ESCOLA POLITÉCNICA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA

FILTRO PARA SEPARAR PERFUSÃO OU RESPIRAÇÃO EM  
IMAGENS DE TOMOGRAFIA POR IMPEDÂNCIA ELÉTRICA

Trabalho de formatura apresentado à Escola  
Politécnica da Universidade de São Paulo para  
obtenção do título de Graduação em Engenharia

Caetano Belda Martinez

Orientador: Raul G. Lima

Área de concentração:  
Bioengenharia

SÃO PAULO

– 2005 –

## AGRADECIMENTOS

Ao meu orientador, Prof. Raul Gonzalez Lima, pela confiança e incentivo no desenvolvimento de meu trabalho, pelos ensinamentos, pela paciência e pela compreensão.

Ao Prof. Walter Ponge, pelas conversas, dicas, aulas e sugestões, instigando sempre a minha curiosidade e incentivando o desenvolvimento de um olhar mais crítico.

Aos demais professores da Escola Politécnica, que me forneceram através de seus conhecimentos e experiências, importantes ferramentas para o meu desenvolvimento acadêmico, profissional e pessoal.

À Escola Politécnica e à USP, pelo investimento na minha graduação.

Aos meus amigos, pelas conversas, festas, companheirismo, força e disposição.

À minha família, pelo apoio, pela amizade, pela razão, pelo ombro e pela base.

À minha namorada, Beatriz, pelo apoio incondicional, paciência, conselhos, colo, compreensão e presença em minha vida.

## RESUMO

O presente trabalho é a compilação do estudo que, motivado pela possibilidade de adicionar clareza a imagens de tomografia por impedância elétrica, foi realizado para a disciplina de Projeto Integrado, junto ao Departamento de Engenharia Mecânica da EPUSP.

O vídeo de estudo é resultante da aquisição de 30 sinais via Tomógrafo por Impedância Elétrica seguido da estimação de parâmetros via modelo de elementos finitos e apresenta imagens de uma sessão transversal do torax que compreende movimentos relacionados à perfusão e à respiração.

Esses efeitos, da respiração e da perfusão sanguínea, aparecem conjugados nos vídeos e o trabalho baseou-se na possibilidade de separá-los.

Para tanto, são utilizados filtros digitais, em particular os filtros do tipo FIR ("Finite Impulse Response") e Wavelet, através de ferramentas computacionais matemáticas.

Uma das dificuldades agregadas à filtragem desses sinais consiste no fato de estas serem posteriores à estimação dos parâmetros, ou seja, sobre o vídeo teoricamente finalizado. Assim, ao invés de serem filtrados 30 sinais, deverão ser filtrados (no caso de uma imagem 640x560) 358.400 sinais, relativos aos pixels da imagem do vídeo, e que são ainda divididos nas 3 cores do sistema RGB (Red, Green, Blue), totalizando mais de 1 milhão de sinais a serem filtrados.

O trabalho traz não apenas a comparação entre os dois tipos de filtros, mas também uma análise crítica de cada um deles individualmente procurando ainda ressaltar aplicações dessa ferramenta relativamente recente e ainda pouco explorada que é a transformada Wavelet. São ainda executadas comparações entre algumas famílias de Wavelet, como por exemplo, Haar e Daubechies.

Outro ponto de vista abordado trata da complexidade computacional, aferida em termos de horas de processamento de um mesmo hardware para as diferentes formulações de filtragem.

## ABSTRACT

This work is the compilation of the studies that, motivated by the possibility to clarify medical videos, was realized to discipline of Integrated Project together with the Mechanical Engineering Department of the EPUSP (Polytechnic School of the Sao Paulo University).

The case study results of the signal acquisition in 30 channels of an Electrical Impedance Tomograph, followed by the finite elements estimative parameters that show images of a thoracic chop with blood perfusion and breathing movements.

Such effects, breathe and blood perfusion, appear together in the videos and this work is based on the possibility to separate them.

For these purposes, digital filters were used, particularly the FIR ("Finite Impulse Response") and Wavelet kinds, using mathematical computer tools.

One found difficulty to the filtering action of this is the fact of realization after the finite element procedure, that increases the number of signals from 30 to 358,400 signals (to a 640x560 image size), relative to each pixel of the video image, that are multiplied by the 3 RGB colors (Red, Green, Blue) totalizing more than 1 million signals to be filtered.

Not only does this work bring the comparison between the two filtering methods, but a critical analysis of each one individually, trying to emphasize the good and the not so good points, and detaching other applications from this recently used method, that has a lot to increase, the Wavelet transform. Have been done yet, comparisons between different forms of Wavelet, like Haar and Daubechies families.

Another point of view has been done about the computation complexity, measured in terms of processing hours in a same hardware to the different filtering formulations.

## SUMÁRIO

1. INTRODUÇÃO .....	1
2. DESENVOLVIMENTO: .....	2
2.1. Tomografia por Impedância Elétrica .....	2
2.1.1. Comparações e aplicações .....	2
2.1.2. Imagens adquiridas .....	3
2.2. Frequências de corte .....	5
2.2.1. Respiração: .....	5
2.2.1.1. Mecanismo da inspiração: .....	5
2.2.1.2. Frequência respiratória .....	6
2.2.2. Perfusão: .....	6
2.2.2.1. Frequência cardíaca .....	7
2.2.3. Determinação da frequência de corte para o filtro FIR: .....	7
2.2.4. Sobre a ordem do filtro FIR: .....	7
2.3. Filtro FIR .....	8
2.3.1. Implementação do filtro FIR .....	9
2.4. Filtro Wavelet .....	11
2.4.1. Introdução à transformada Wavelet .....	12
2.4.1.1. Histórico .....	13
2.4.2. Funções de base .....	15
2.4.3. Funções de escala .....	16
2.4.4. Transformada de Fourier X Transformada Wavelet .....	16
2.4.4.1. Similaridades .....	16
2.4.5. Particularidades .....	17
2.4.6. Bases Wavelet .....	18
2.4.6.1. Diferenciação das famílias Wavelet .....	20
2.4.6.2. Escolha da família Wavelet .....	23
2.4.7. A transformada discreta Wavelet (DWT) .....	23
2.4.8. Introdução a DWT .....	24
2.4.9. A transformada Wavelet discreta (DWT) .....	26

2.4.10.	A codificação de sub-banda e a análise de multi-resolução .....	26
2.4.11.	Os coeficientes DWT .....	29
2.4.12.	A transformada rápida Wavelet (FWT) .....	29
2.4.13.	Transformada Wavelet Packet (WPT) .....	30
2.4.14.	Formas de onda adaptadas .....	30
2.4.15.	Exemplos de aplicação.....	31
2.4.16.	Algoritmo da implementação Wavelet unidimensional em Matlab .....	32
2.4.16.1.	Primeiro passo.....	32
2.4.16.2.	A transformada inversa.....	34
2.4.17.	Implementação .....	35
2.4.17.1.	O sinal .....	35
3.	TRATAMENTO DE UM SINAL .....	38
3.1.	Determinação da ordem do filtro FIR.....	39
3.1.1.	Determinação da ordem do filtro Wavelet:.....	39
3.2.	Estendendo o tratamento para o vídeo .....	40
3.2.1.	Manipulação do vídeo.....	40
3.2.2.	Separação dos arquivos de imagem .....	41
3.2.3.	Criação das matrizes .....	41
3.2.4.	Redução do processamento.....	42
3.2.5.	Testando os filtros para um pixel.....	43
4.	RESULTADOS E ANÁLISES:.....	44
4.1.	Sinal do eletrodo com filtro FIR: .....	44
4.2.	Sinal do pixel com filtro FIR: .....	47
7		
4.4.	Diferentes filtragens Wavelet para o pixel: .....	54
4.5.	Passagens da decomposição Wavelet do pixel: .....	57
4.6.	Plano da filtragem Wavelet:.....	66
4.7.	Resultados da filtragem do vídeo com FIR: .....	67
4.8.	Resultados da filtragem do vídeo com Wavelet: .....	67
	CONCLUSÕES .....	68

BIBLIOGRAFIA RECOMENDADA: .....	71
ANEXO I – Código Fonte Scilab – Rotina FIR.....	72
ANEXO II – Código Fonte Matlab – Rotina para separar as matrizes RGB.....	74
ANEXO III – Código Fonte Matlab – Redução em quadrantes da área filtrada .....	75
ANEXO IV – Código Fonte Matlab – Filtro FIR para todos os pixels.....	77
ANEXO V – Código Fonte Matlab – Recomposição das imagens .....	80
ANEXO VI– Código Fonte Matlab – Recomposição do vídeo em AVI.....	81
ANEXO VII – Código Fonte Matlab – Filtragem Wavelet para todos os pixels .....	82



## 1. INTRODUÇÃO

A partir de um Tomógrafo por Impedância Elétrica são obtidas imagens onde é possível verificar a sobreposição dos fenômenos da perfusão e da respiração de um paciente, além é claro de um determinado nível de ruído. Utilizando-se da aplicação de filtros digitais é possível a obtenção de imagens que acondicionam e, conseqüentemente, permitem a visualização de cada um dos fenômenos citados, de maneira isolada. É evidente que para o diagnóstico ou acompanhamento de um paciente, a clareza das imagens pode ser bastante útil permitindo ainda que se analise localmente o fluxo de sangue ou de ar, reduzindo inclusive alguma possível margem para erros médicos.

Os dois modelos de filtro a serem aplicados, testados e comparados para o tratamento da imagem do tomógrafo são baseados nos princípios FIR ("Finite Impulse Response") e Wavelet.

O método FIR utiliza-se de uma convolução, baseado na inversa da transformada de Fourier. Em geral o seu custo computacional é alto, apresentando porém linearidade e estabilidade.

O conceito Wavelet por sua vez é relativamente novo (data da década de 80) e apresenta infindável campo de aplicações em sinais de duas ou mais dimensões (como no caso de imagens), sendo atualmente a tendência no que tange o processamento computacional de sinais. O procedimento Wavelet mostra-se indicado para o tratamento de sinais complexos de forma personalizada para cada caso e apresenta de maneira geral custo computacional mais baixo.

## **2. DESENVOLVIMENTO:**

Nesse item serão abordados alguns aspectos relevantes ao projeto, relativos aos dois filtros: FIR e Wavelet, ao Tomógrafo por Impedância Elétrica e ao tratamento de imagens. .

### **2.1. Tomografia por Impedância Elétrica**

O processo de Tomografia por Impedância Elétrica funciona pela estimação do campo interno de condutividade, com o menor erro possível entre medidas e valores previstos por um modelo, através da medição de potenciais (ou de correntes) impostos por correntes (ou potenciais) aplicadas aos eletrodos em determinados pontos da superfície em estudo.

#### **2.1.1. Comparações e aplicações**

Existem atualmente, diversas frentes de pesquisas relacionadas à área médica que buscam desenvolver equipamentos, métodos e processos eficientes de aquisição, manipulação, tratamento e resolução de imagens. Uma dessas empreitadas está relacionada à aquisição de sinais vitais através da Tomografia por Impedância Elétrica (EIT). Em geral, as imagens adquiridas através da EIT, quando comparadas ao ultrassom ou a Tomografia Computadorizada podem ser consideradas de baixa resolução apresentando, no entanto, diversas vantagens perante as técnicas citadas.

Primeiramente, a EIT é não invasiva, não emite radiações e, portanto, não é nociva ao paciente e apresenta custo estrutural reduzido, pois não há necessidade da instalação de uma infra-estrutura especial, como por exemplo, blindagem. O custo computacional também é relativamente baixo e principalmente, por tratar-se de um equipamento com dimensões reduzidas, pode ser transportado, esse fato elimina possíveis inconvenientes de deslocamento no caso de pacientes impossibilitados (muitas vezes em leito de UTI, por exemplo) isso implica ainda que os equipamentos ligados ao paciente não precisem ser desligados ou transportados com o mesmo para a realização da tomografia. Outro

fator que agrega vantagens ao processo é a possibilidade de realizar o procedimento em momento anterior ao caráter emergencial, uma vez que o mesmo pode ser feito inclusive na própria UTI.

A EIT possibilita o monitoramento do paciente em tempo real abrindo espaço para uma aplicação bastante interessante no controle de unidades de ventilação artificial, que são atualmente controladas manualmente e, portanto, sujeita a muito mais erros humanos.

### **2.1.2. Imagens adquiridas**

As imagens, da seção do tórax, adquiridas pela EIT apresentam dois padrões bem definidos e marcantes, um devido à respiração, e outro relacionado ao ritmo cardíaco denominado de agora em diante perfusão. A relação entre perfusão e respiração é também um interessante aspecto a se observar.

As imagens a seguir ilustram quatro instantes distintos de um vídeo onde é possível visualizar a sístole, a diástole, a inspiração e a expiração em fenômenos combinados em uma seção do tórax.

A primeira figura apresenta o pulmão preenchido e a pulsação cardíaca em duas situações, à esquerda em sístole e a direita em diástole. A segunda figura mostra o pulmão em momento de expiração, conseqüentemente com baixo volume e a pulsação cardíaca de diástole e sístole.

Fig. 2.1 – Inspiração e pulsação cardíaca

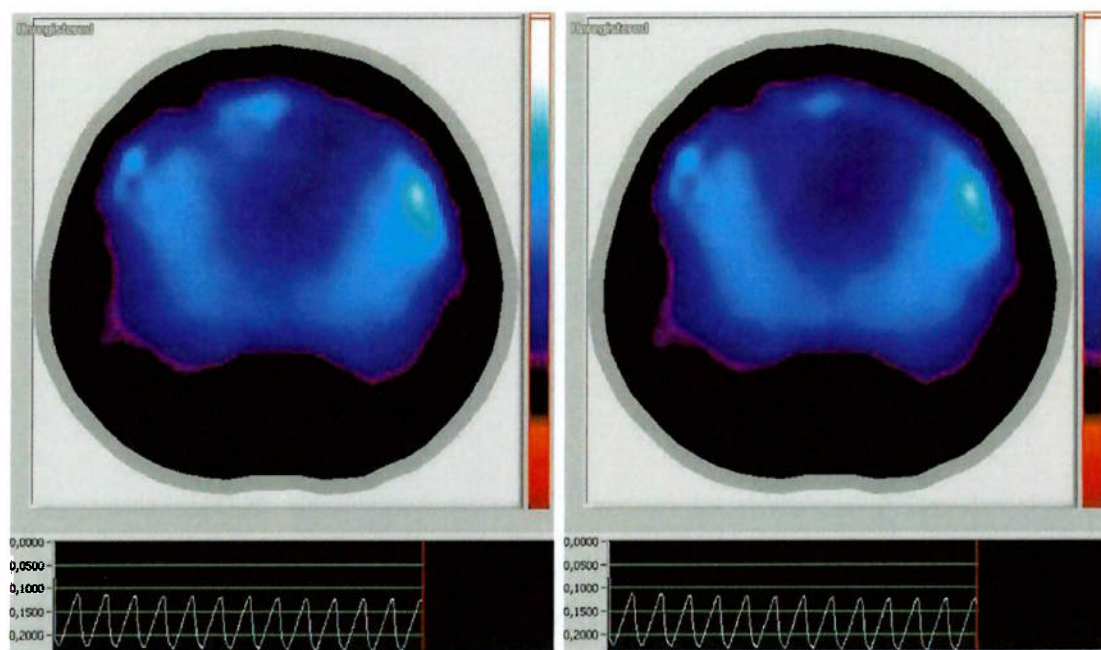
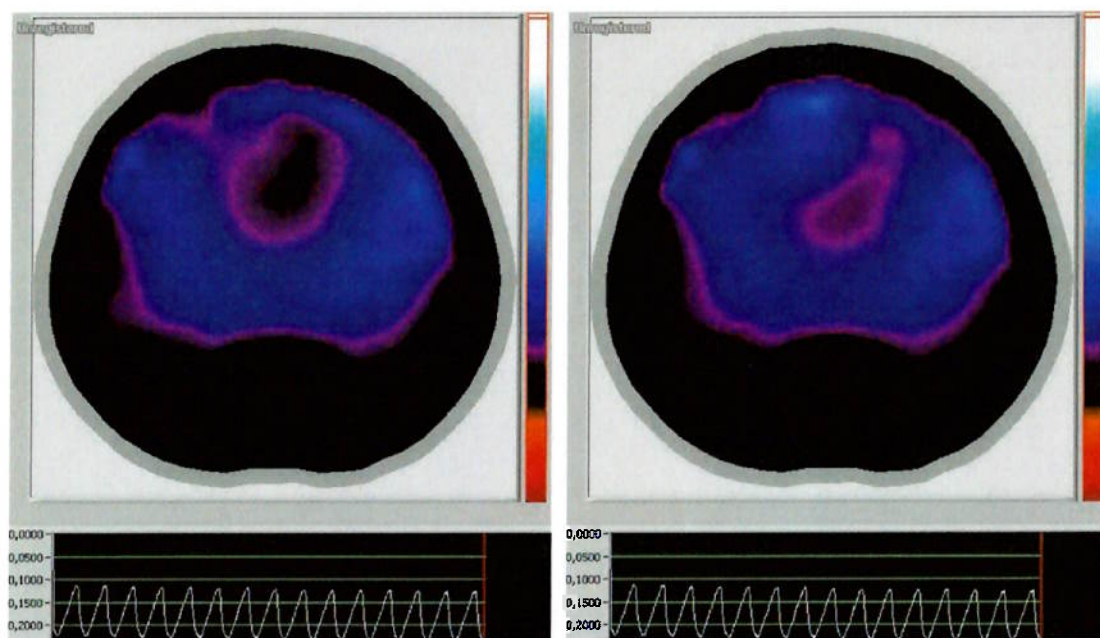


Fig. 2.2 – Expiração e pulsação cardíaca



Estas imagens foram retiradas de um arquivo que não foi filtrado, e através das figuras é possível perceber as duas regiões em questão, formadas pelos pulmões na lateral e

pelo coração na região central superior, a idéia é que esses dois efeitos sejam separados e observados individualmente em um novo arquivo de vídeo, perceba que esta observação só será bem detectada através da comparação dos filtros.

## **2.2. Freqüências de corte**

Como serão tratadas nesse estudo as freqüências de respiração e perfusão, é conveniente que seja explicado um pouco o funcionamento desses mecanismos:

### **2.2.1. Respiração:**

A intensidade e o padrão da respiração são determinados pela contração combinada dos grupos musculares do diafragma, da caixa torácica, do abdômen e das estruturas circundantes. A regulação da respiração é basicamente um sistema de resposta alimentado por entradas providas de múltiplas regiões sensoras. O ciclo respiratório é então controlado pelo centro respiratório (localizado no tronco cerebral) e modificado em resposta às informações obtidas pelos centros cerebrais superiores e receptores sistêmicos. O centro de controle respiratório está localizado no bulbo e trata-se de um conglomerado de grupos nervosos distintos anatomicamente e capazes de gerar um padrão ventilatório.

A atividade respiratória pode ser alterada, em resposta a uma infinidade de estímulos, tais quais visuais, emocionais ou voluntários.

#### **2.2.1.1. Mecanismo da inspiração:**

Visto que o padrão respiratório pode contar com inúmeras variáveis, serão consideradas como ponto de partida, as situações de normalidade em termos de corpo humano, ou seja, o repouso.

Na situação basal, células auto-excitáveis através de um sinal em rampa (comandado pelo centro inspiratório) produz por cerca de dois segundos, a inflação dos pulmões e volta a agir após três segundos de súbita parada. Dessa região parte um conjunto de fibras que através da medula se dirige aos neurônios motores responsáveis pelos grupos musculares da inspiração.

#### **2.2.1.2. Frequência respiratória**

Na condição acima explicitada pode-se considerar a atividade pulmonar como um sinal cíclico de frequência da ordem de 0,2 Hz, ou seja, período de duração da ordem de 5 s.

#### **2.2.2. Perfusão:**

Como a respiração, o ritmo da perfusão se altera ao longo das atividades cotidianas segundo alguns sistemas de controle, basicamente divididos em intrínsecos e extrínsecos:

##### **Controle Intrínseco:**

As fibras musculares cardíacas quando recebem o volume sanguíneo proveniente do sistema venoso distendem-se, quanto maior essa distensão, um conjunto de fibras chamado fibras de Purkinje tornam-se mais sensíveis e conseqüentemente mais excitáveis conforme o grau dessa distensão. Essa excitabilidade acarreta no aumento da frequência de descarga rítmica na despolarização de tais fibras, verificam-se então as alterações da própria frequência cardíaca.

**Controle extrínseco:**

O coração também altera a sua atividade de acordo com o grau de atividade do Sistema Nervoso Autônomo (SNA). O SNA exerce grande influencia no funcionamento de diversos tecidos do nosso corpo através dos sistemas Simpático e Parassimpático.

Um predomínio da atividade simpática do SNA provoca, no coração, um significativo aumento na frequência e da força de contração cardíaca.

Já um predomínio da atividade parassimpática do SNA, através da liberação de acetilcolina provoca o efeito oposto no coração: redução na frequência e na força de contração cardíaca.

**2.2.2.1. Frequência cardíaca**

Para a atividade cardíaca o prospecto é o mesmo de quando se falou em respiração, cada caso apresenta deve ser analisado em particular, porem, para um passo inicial pode ser considerada a frequência em repouso de um adulto médio da ordem de 60 a 80 batimentos por minuto, ou seja, algo superior a 1 Hz.

**2.2.3. Determinação da frequência de corte para o filtro FIR:**

A partir do apresentado, é de se imaginar que a frequência de corte escolhida deva se situar entre as faixas de 0,2 Hz e 1Hz. Nesses padrões a frequência de 0,5Hz é, por exemplo, uma boa escolha.

**2.2.4. Sobre a ordem do filtro FIR:**

A partir da frequência de corte e da taxa de amostragem (característica do sinal) pode-se definir a ordem do filtro FIR. A ordem do filtro representa o tamanho do vetor de frequências do sinal, e conforme o comprimento do vetor (ordem) e a taxa de aquisição (máxima frequência observável) será definida a largura de cada banda, ou seja, a



amplitude da faixa de frequência compreendida por cada ponto discreto do filtro, e a qual será imposto um determinado ganho. É importante lembrar sempre que o custo computacional está diretamente ligado à ordem do filtro e que por tanto a menor ordem possível sem prejudicar a eficiência é a melhor alternativa.

No devido momento essa definição será abordada novamente.

### 2.3. Filtro FIR

A grande vantagem da filtragem de sinais pelo filtro FIR (“Finite Impulse Response”) se dá pela sua resposta linear em relação à fase da entrada, quando temos um sistema causal, ou seja, independente dos valores futuros da entrada. Este fato permite a filtragem em tempo real do sinal de entrada.

O custo implicado pela filtragem via FIR por sua vez é dado pela alta ordem do mesmo a fim de permitir bons resultados.

Sendo o filtro de ordem  $M-1$ , a resposta do sinal à filtragem é obtida a partir das  $M-1$  amostras do sinal anteriores ao instante considerado e o próprio instante. O filtro FIR não é, portanto, re-alimentado sendo chamado também de filtro de média móvel ou filtro não recursivo.

Chamando de  $x$  a entrada e de  $y$  a saída do sinal, para as  $n$  amostras ( $n$  sempre variando de 0 até  $M-1$ ) teremos a seguinte equação de diferenças:

$$y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + \dots + b_{M-1} \cdot [n - (M-1)]$$

Ou seja:

$$y(n) = \sum_{k=0}^{M-1} b_k \cdot x(n-k)$$

Onde  $b_k$  representa os coeficientes do filtro FIR.



Alternativamente podemos representar o sinal de saída como a convolução entre a resposta em frequência imposta para cada “sample”  $h(n)$  com o sinal de entrada, obtendo:

$$y(n) = \sum_{k=0}^{M-1} h_k \cdot x(n-k)$$

Comparando as duas equações anteriores, é claro que os coeficientes  $b_k$  são a resposta em frequência  $h_k$ .

A função de transferência do filtro FIR (no domínio da frequência) apresenta-se então da seguinte maneira:

$$H(z) = \sum_{k=0}^{M-1} h_k \cdot z^{-k}$$

Por não apresentar denominador é notável que a resposta é sempre estável.

Para que o sistema tenha resposta linear de fase, é necessária ainda uma condição de simetria, explicitada pela seguinte condição:

$$h(n) = \pm h(M-1-n)$$

Incorporada a condição de simetria é possível ainda comprovar algebricamente que a fase de resposta é linear e tem a seguinte forma:

$$\Theta(\omega) = \frac{\beta \cdot \pi}{2} - \frac{\omega \cdot (M-1)}{2}.$$

### 2.3.1. Implementação do filtro FIR

Na prática a implementação do filtro FIR faz-se por meio de convoluções, possibilitadas justamente devido ao número finito de amostras. Basicamente a síntese

do filtro FIR é feita através da Transformada Inversa de Fourier (TFI) de uma seqüência que representa a amplitude do filtro no domínio da freqüência, gerando a resposta ao impulso do filtro, que é uma seqüência finita no tempo. O filtro FIR é uma convolução do sinal de entrada com a resposta ao impulso do filtro.

Será agora descrito o algoritmo da filtragem para um sinal  $v(t)$  qualquer obtido por amostragem. Os códigos fonte utilizados nas simulações em “Scilab” e “Matlab” estão em anexo.

Lembrando sempre que a faixa de freqüência observável é composta por metade do espectro, respeitando o critério de Nyquest, para que não sejam observados efeitos de “aliasing”, uma vez que a máxima freqüência observável deve ser determinada por um período onde ocorram no mínimo duas aquisições, ou seja, o sinal só pode ser corretamente avaliado se forem capturados dois instantes referentes a um mesmo ciclo (um positivo e um negativo em relação à média).

#### **Algoritmo:**

$v$  = Sinal  $\rightarrow$  adquirido por um certo período (T) que possui um número de amostras (n).

$f_s = n / T \rightarrow f_s$  é a taxa amostragem do sinal por período de tempo.

$M$  = Ordem do filtro  $\rightarrow$  deve ser impar, foi adotada ordem 21

$w(i) = 2 * \pi * (i-1) / M$ , com  $i$  variando de 1 até a  $(M/2+1) \rightarrow$  anti-aliasing

define a resposta em freqüência do "sistema linear"

Para  $i = f_c \rightarrow H(i) = 0.4$  para aliviar o truncamento

se o filtro for passa baixa:

para  $i = 1 : f_c \rightarrow H(i) = 1;$

para  $i = 7 : 11 \rightarrow H(i) = 0.0;$

se o filtro for passa alta

para  $i = 1 : f_c \rightarrow H(i) = 0;$

para  $i = 7 : 11 \rightarrow H(i) = 1.0;$

Para  $n$  variando de 1 a 10 e  $k$  variando de 1 a 11:

$a(k,n) = 2 * \cos(w(k) * ((M-1)/2 - (n-1)))$ ;

Para  $k$  variando de 1 a 11;

$a(k,11)=1;$

Determinando os coeficientes do filtro FIR,  $h=FFT^{-1}(H)$ .

$h=inv(a)*H$

Para satisfazer a condição de simetria nos coeficientes:

Para  $i$  variando de 1 a 11;

$ht(i)=h(i);$

Para  $i$  variando de 1 a 10;

$ht(21-(i-1))=h(i);$

E aplicando o filtro ao sinal  $v \rightarrow$  obtem-se  $v_{filtrado}$

$vf(1)=v(1)*h(1);$

$vf(2)=v(2)*h(1)+v(1)*h(2);$

$vf(3)=v(3)*h(1)+v(2)*h(2)+v(1)*h(3);$

para  $i=1:21$

$aux=0.0;$

para  $j=1:i$

$aux=aux+v(i-(j-1))*ht(j);$

$vf(i)=aux;$

para  $i=22:456;$

$aux=0.0;$

para  $j=1:21;$

$aux=aux+v(i-(j-1))*ht(j);$

$vf(i)=aux;$

## 2.4. Filtro Wavelet

Nessa seção serão definidos alguns conceitos importantes relativos à transformada Wavelet, em seguida, serão expostas notas sobre as bases Wavelet e com respeito à sua implementação propriamente dita.

### 2.4.1. Introdução à transformada Wavelet

Wavelets são funções que satisfazem certos pressupostos matemáticos e são usados em representações de dados ou funções. A idéia em si não é nova, a aproximação por superposição de funções data do início do século XIX através da formulação por Fourier da transformada em base de senos e cossenos. Todavia, na análise Wavelet, a escala utilizada para analisar o fragmento de sinal desempenha importante papel.

Os algoritmos Wavelet processam os dados em diferentes resoluções (escalas), se um sinal for analisado através de janelas amplas, serão pronunciados os efeitos (características) de grande escala, e em contrapartida utilizando janelas mais finas, são pronunciados os efeitos mais sutis. Assim, o objetivo da análise via transformada Wavelet é conseguir observar tanto os efeitos globais, quanto os efeitos locais.

É justamente essa possibilidade de análise ampla que faz da transformada Wavelet uma ferramenta interessante e bastante útil.

Por décadas os cientistas buscaram funções mais apropriadas do que os senos e cossenos (base da FFT) na aproximação dos fragmentos de um sinal. Através da análise via Wavelet é possível aproximar funções contidas em domínio finito, diferentemente de um sinal senoidal, assim, a análise de descontinuidades é muito bem vista pela transformada Wavelet.

O procedimento de análise leva em consideração a adoção de uma função mãe, (“Wavelet Mother”, também chamada “Analyzing Wavelet”). A análise temporal é feita por uma componente de alta frequência (contraída) enquanto a análise em frequência é obtida por uma componente dilatada da mesma função.

Como o sinal original pode ser representado em termos da expansão Wavelet (utilizando uma combinação linear dos coeficientes das funções), a operação dos dados pode ser efetuada através da utilização dos coeficientes correspondentes. Escolhendo a família de funções mais apropriada para os dados a serem analisados, ou através de truncamentos eficientes, a transformada se mostra uma excelente ferramenta de compressão do sinal.

As funções Wavelet apresentam ótima funcionalidade para estudos astronômicos, acústicos, nucleares, processamento de imagens e sinais, predição de terremotos, radares, resolução de equações diferenciais entre tantas outras aplicações.

#### 2.4.1.1. Histórico

Historicamente, a análise Wavelet apresenta diversas origens que convergem para a atual situação da ferramenta, pode-se dividir esse desenvolvimento entre os períodos anteriores e posteriores à década de 30.

##### Antes de 1930

Indiscutivelmente a análise Wavelet iniciou-se a partir da transformada de Fourier (século XIX), que leva uma função do domínio do tempo ao domínio da frequência (já explicado nos capítulos anteriores). A transformada associa a qualquer função a seguinte formulação:

$$a_0 + \sum_{k=1}^{\infty} (a_k \cdot \cos(k \cdot x) + b_k \cdot \sin(k \cdot x)),$$

E com os coeficientes obtidos por:

$$a_0 = \frac{1}{2 \cdot \pi} \int_0^{2\pi} f(x) dx, \quad a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cdot \cos(kx) dx \quad \text{e} \quad b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cdot \sin(kx) dx$$

A partir dessa formulação por Fourier, abriu-se um novo universo no que se refere ao tratamento de dados e funções. A partir de 1807 os matemáticos começaram a explorar convergências das series de Fourier, sistemas ortogonais e gradualmente foram se redirecionando da análise em frequência para a análise em escala. O que significa analisar  $f(x)$  através de estruturas matemáticas que variam em escala, da seguinte maneira: Aplica-se essa determinada estrutura ao sinal sucessivamente porem em diferentes escalas, e às escala correlacionam-se os coeficientes obtidos. Esse leque de

escalas se mostrava menos suscetível a ruído, pois eram medidas as médias de flutuação do sinal em cada escala particular.

A primeira menção a teoria das Wavelets aparece no apêndice da tese de A. Haar (1909), uma das propriedades da Wavelet de Haar que a torna de uso limitado consiste no fato de ela não ser continuamente diferenciável.

### **Os anos 30**

Muitos grupos trabalhavam em pesquisas independentes a representação de funções de base em variação de escala.

Paul Levi notou a superioridade na análise de sinais randômicos pela base de funções de Haar em relação à transformada de Fourier.

Ainda em 1930 Littlewood, Paley e Stein envolvidos no cálculo da energia de uma função  $f(x)$  que mostrava diferentes resultados quando se tratavam de casos de energia distribuída ou concentrada ao redor de alguns pontos, o que indicava uma não conservação dessa energia.

Essas pesquisas levaram à descoberta de uma função que podia variar em escala e conservar a propriedade da energia. Esse trabalho serviu de base para o trabalho de David Marr, com um algoritmo efetivo para o processamento de imagens, já no princípio da década de 80.

### **Décadas de 60 a 80**

Nas décadas que compreendem o período entre 1960 e 1980, os matemáticos Guido Weiss e Ronald R. Coifman estudaram elementos simples de uma função espaço, chamados átomos, com a meta de encontrar uma função comum a essa classe e que possibilitasse a reconstrução de todos os elementos da função de espaço utilizando esses átomos. Em 1980, Grossman and Morlet, definiram Wavelets no contexto da física quântica, criando uma nova mentalidade para o conceito da transformada.

### **Após 1980**

Em 1985, Stephane Mallat deu às Wavelets um impulso adicional, através de seu trabalho no processamento de sinais digitais. Mallat descobriu relações entre algoritmos de pirâmide, filtros “quadrature mirror” e as bases ortonormais Wavelet. Inspirado nesses resultados, Y. Meyer construiu o primeiro Wavelet não trivial. Diferentemente das Wavelets de Haar, as Wavelets de Meyer são continuamente diferenciáveis, embora não apresentem uma estrutura muito compacta. Alguns anos depois, Ingrid Daubechies utilizou-se do trabalho de Mallat para construir uma nova família de funções Wavelet em bases ortonormais, que podem ser consideradas talvez como a mais elegante das soluções e a que melhor reflete as atuais aplicações.

#### **2.4.2. Funções de base**

Toda função que não seja unidimensional pode ser considerada como a combinação de uma base de vetores (no caso digital) ou de funções (no caso analógico), a melhor base possui ainda a propriedade de ortogonalidade entre os seus elementos.

Apenas de maneira ilustrativa podemos considerar um vetor de duas dimensões  $(x,y)$  como sendo a soma do vetor da multiplicação de  $x$  por  $(1,0)$  com o vetor da multiplicação de  $y$  por  $(0,1)$ . Dessa maneira o elemento  $(x,y)$  foi decomposto na base ortogonal composta pelos vetores  $(1,0)$  e  $(0,1)$ .

No caso de um elemento analógico, como uma função  $f(x)$ , por exemplo, podemos relembrar o caso da transformada de Fourier, que decompõe em uma soma de senos e cossenos através da correta aproximação de frequências e amplitudes. Essas funções são ainda ortogonais (dado que o produto escalar das mesmas é igual a zero).

### **2.4.3. Funções de escala**

Uma função de base varia na escala de maneira que uma mesma função ou espaço seja escalonado em diferentes classes. Por exemplo, imagine um sinal no domínio de 0 a 1. Este sinal pode ser dividido como duas funções, uma da etapa em que a escala varia entre 0 a 1/2 e outra com a escala variando de 1/2 a 1. O mesmo sinal pode também ser dividido em quatro funções, de 0 a 1/4, 1/4 a 1/2, 1/2 a 3/4, e 3/4 a 1. E assim por diante. Cada uma das representações codifica o sinal original com uma definição particular, ou seja, escalonando-o de maneiras distintas.

### **2.4.4. Transformada de Fourier X Transformada Wavelet**

Como dito anteriormente, a transformada de Fourier antecede a transformada Wavelet em pelo menos um século e pode-se dizer que a segunda veio em função de uma necessidade percebida pela utilização da transformada de primeira, vale lembrar que em nada a questão é comparar as duas quanto à validade ou não. A transformada de Fourier apresenta em determinadas situações excelentes resultados, onde muitas vezes é pouco conveniente que se tenha um tratamento mais complexo (caso da Wavelet), o objetivo desse capítulo é pura e simplesmente o de levantar similaridades e particularidades de cada uma das transformadas, de maneira a facilitar o entendimento dos dois tratamentos.

#### **2.4.4.1. Similaridades**

As duas transformadas que são chamadas “Fast Fourier e Transform” (FFT) e a “Discrete Wavelet Transform” e (DWT), são operações lineares que geram a partir de estruturas de dados que contem  $\log_2 n$  segmentos de diversos comprimentos em vetores de dados de comprimento  $2^n$ .



As propriedades matemáticas que envolvem as matrizes de transformação são similares, a matriz de transformada inversa é a inversa da matriz de transformação. Embora as bases da transformada Wavelet (“Wavelet Mother”) sejam um pouco mais complexas do que as senoidais da transformada de Fourier, a mecânica é bastante parecida.

Uma outra similaridade se encontra na localização das funções de base no domínio da frequência, fazendo delas ferramentas que analisam a energia contida em um intervalo de frequência (análise espectral) sendo muito úteis na determinação da distribuição de energia de um sinal.

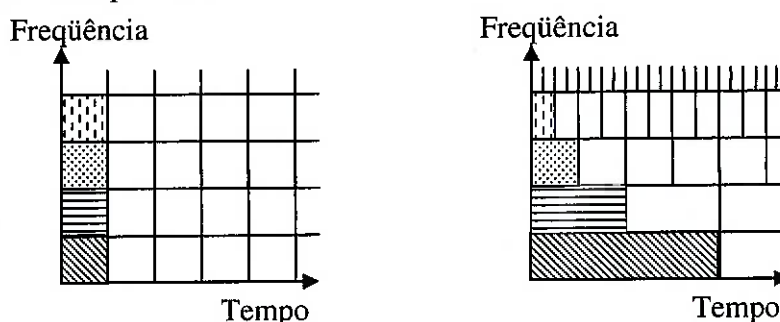
#### 2.4.5. Particularidades

A transformada por ondulas (Wavelet), gera espectros ao longo do tempo, ao passo que a transformada de Fourier gera um espectro único para toda a série histórica.

Para melhor ilustrar as duas transformadas (FFT e Wavelet), aqui serão representadas no plano do tempo-frequência as regiões de cobertura das funções de base para ambos os casos.

À esquerda a representação da transformada de Fourier e à direita da transformada Wavelet.

Fig. 2.3 – Plano comparativo



Uma das grandes vantagens conforme observado na figura acima é o tamanho da janela, maior para menores frequências, e menor para maiores frequências, o que reduz também o efeito de truncamento da janela aplicada à análise.

Essa disposição permite ainda uma análise direta à informação, ou seja, a sua localização no tempo, enquanto a transformada de Fourier obscurece esse aspecto.

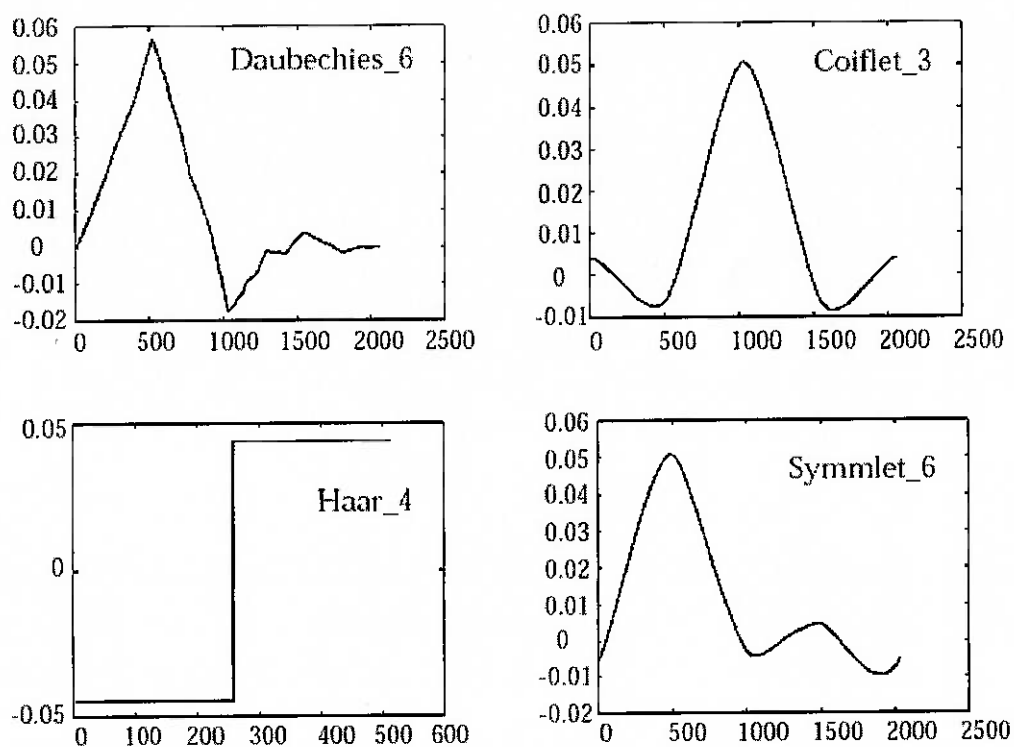
Outra grande vantagem se dá ao fato de a transformada Wavelet possuir infinitas bases e não se limitar a uma família de funções, apresentando assim uma melhor adequação a cada tipo de sinal, muito embora isso exija uma análise quanto à melhor base para cada aplicação.

#### **2.4.6. Bases Wavelet**

Conforme discutido, muitas são as famílias de ondas que compreendem a denominação Wavelet (na realidade são infinitas). Esta seção apresenta algumas delas (especialmente as supracitadas) e em especial aborda algumas particularidades.

Lembrando que para cada aplicação uma diferente família apresentará os melhores resultados, de acordo com os objetivos de cada estudo. Para uma primeira abordagem é conveniente exemplificar através de imagens algumas dessas famílias, as figuras a seguir são elementos representantes de algumas famílias representativas de Wavelets:

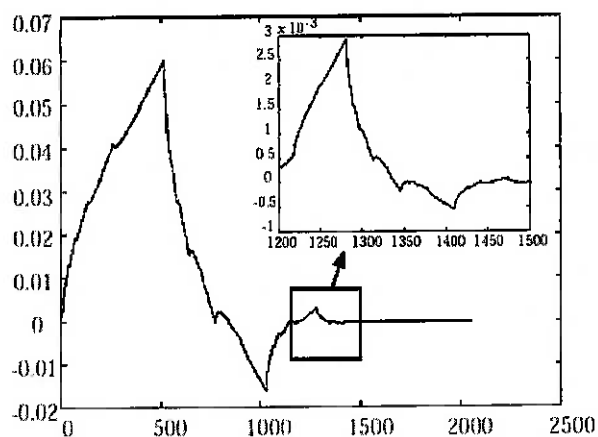
Fig. 2.4 – Exemplos de famílias



Observe que os elementos têm duração finita e são bem diferentes entre si.

Repare ainda na figura a seguir, que ampliando o segundo pico da função de Daubechies, é perceptível uma repetição com menor intensidade do primeiro pico:

Fig. 2.5 – Detalhe da função de Daubechies



### 2.4.6.1. Diferenciação das famílias Wavelet

Conforme explicitado anteriormente, para executar a análise Wavelet, é necessário escolher uma família que se adapte ao sinal analisado. Para tanto será feita uma breve análise de algumas famílias pertinentes, ressaltando que a nomenclatura das subdivisões é meramente ilustrativa.

Fonte: Matlab Wavelet Toolbox

#### 1. Famílias simples

Famílias: Gaussiana, Morlet e Chapéu Mexicano (Mexican Hat)

Propriedades: Análise não ortogonal, sem fase  $\phi$ , sem suporte a compactação, a reconstrução não se apresenta muito segura.

Análises: Decomposição continua.

Características: Simetria, algoritmo rápido e reconstrução inviável.

#### 2. Famílias infinitamente regulares

Família: Meyer

Propriedades: Análise ortogonal, existe fase  $\phi$ , sem suporte a compactação, indefinidamente derivável.

Análises: Decomposição continua, transformada discreta.

Características: Simetria, regularidade.

Família: Meyer discreta

Propriedades: Aproximação FIR da Wavelet Meyer.

Análises: Decomposição continua, transformada discreta.

#### 3. Ortogonal e com suporte a compactação

Famílias: Daubechies, Symlets, Coiflets.

Propriedades: Análise ortogonal, existe fase  $\phi$ , com suporte a compactação, quantidade dada de instantes de descontinuidade.

Análises: Decomposição continua, transformada discreta via FWT.

Características: Descontinuidades, filtros FIR, regularidade pobre, cada família apresenta ainda algumas propriedades específicas.

#### **4. Bi-ortogonal e com suporte a compactação**

Família: B-splines.

Propriedades: Análise bi-ortogonal, existe fase  $\phi$ , com suporte a compactação, descontinuidades, regularidade conhecida.

Análises: Decomposição continua, transformada discreta via FWT.

Características: Simetria com filtros FIR, perda da ortogonalidade.

#### **5. Complexas**

Famílias: Gaussiana complexa, Morlet complexa, Shannon complexa, B-spline complexa.

Propriedades: Análise não ortogonal, não existe fase  $\phi$ , sem suporte a compactação.

Análises: Decomposição continua complexa.

Características: Simetria, algoritmo rápido e reconstrução inviável.

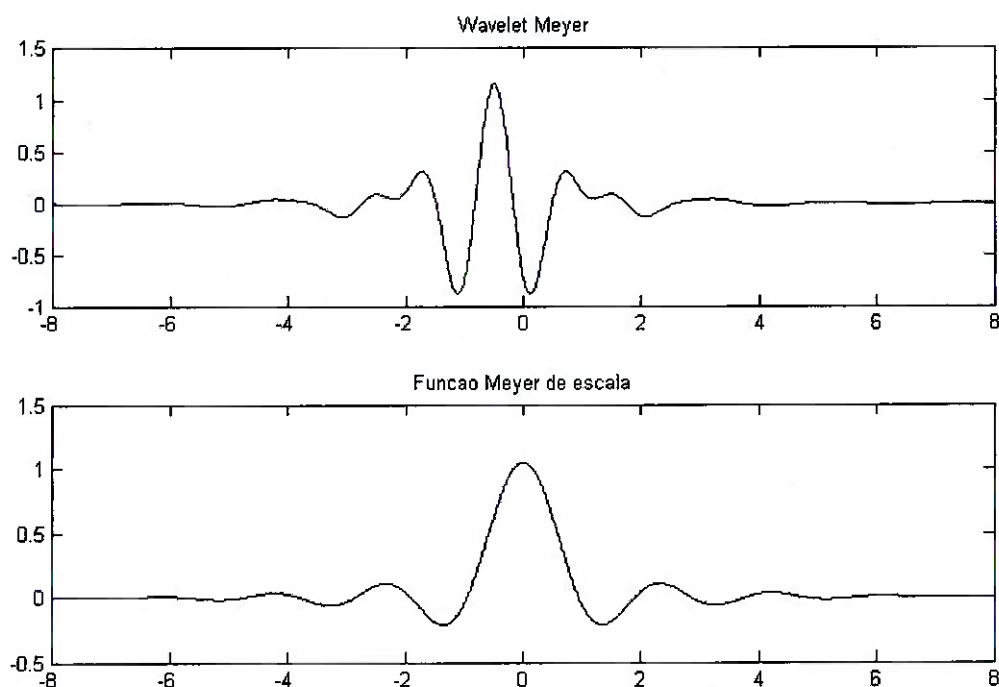
O quadro a seguir traduz um resumo de algumas particularidades de cada família, fornecendo uma boa base de comparação entre as diferentes famílias.



#### 2.4.6.2. Escolha da família Wavelet

Muitas são as possibilidades na escolha da família a ser utilizada no tratamento do sinal. Dentre todas, apenas algumas são convenientes e dentre estas, foi escolhida para este caso particular a transformada discreta de Meyer (D-Meyer), para a decisão foi levado em consideração a questão do domínio discreto, velocidade de processamento e da boa reconstrução dom sinal a partir de testes. A Wavelet discreta de Meyer apresenta o a seguinte estrutura:

Fig. 2.7 – Funções Wavelet e de escala:



#### 2.4.7. A transformada discreta Wavelet (DWT)

Embora a transformada Wavelet contínua discretizada (em termos computacionais) nos pareça em primeira vista uma transformada discreta, em fato ela não passa de uma simples versão em samples da transformada continua, com informações redundantes em termos da análise de sinais. Essa redundância demanda alta capacidade computacional e

conseqüentemente: muito tempo e recurso. A transformada Wavelet discreta (DWT) por sua vez provê informação suficiente para a análise e síntese do sinal original, com uma significativa redução de tempo computacional.

A DWT tem implementação consideravelmente simples se comparada a CWT (contínua). Os conceitos básicos e propriedades serão introduzidos nesta seção, visando analisar de maneira voltada para a implementação computacional..

#### 2.4.8. Introdução a DWT

Dilatações e translações da função mãe ( $\phi(x)$ ), definem uma base ortogonal, a base Wavelet a seguir representada:

$$\phi_{(s,l)} = 2^{\frac{-s}{2}} \phi(2^{-s}x - l)$$

As variáveis  $s$  e  $l$  são inteiros que dilatam e escalonam a função mãe  $\phi$  para que sejam geradas as Wavelets, como por exemplo nessa análise, a família Wavelet de Meyer.

O coeficiente  $s$  representa a escala e conseqüentemente determina a largura da função e o coeficiente  $l$ , a posição. Note que a função mãe é redimensionada (ou dilatada) em potencias de dois e transladada por números inteiros. O que faz da Wavelet especialmente interessante é justamente essa similaridade dentro de uma família, causada pelos redimensionamentos e reposicionamentos. Ou seja, a partir da Wavelet mãe, temos um amplo conhecimento da base.

Para expandir o domínio para diferentes resoluções, a Wavelet mãe é arranjada em uma equação de escala:

$$W(x) = \sum_{k=-1}^{N-2} (-1)^k c_{k+1} \phi(2x + k)$$



Aqui,  $W(x)$  é a função de escala para a Wavelet mãe ( $\phi$ ), e  $c_k$  os coeficientes Wavelet, que devem satisfazer:

$$\sum_{k=0}^{N-1} c_k = 2, \quad \text{e} \quad \sum_{k=0}^{N-1} c_k c_{k+2l} = 2\delta_{l,0}$$

Onde  $\delta$  é a função delta e  $l$  é o índice de localização.

Uma característica bastante útil do sistema Wavelet é a facilidade com que é possível escolher os coeficientes do sistema, para adaptá-lo ao problema específico. Um exemplo a ser considerado se dá no desenvolvimento por Daubechies em uma de suas publicações, de famílias específicas de Wavelet que apresentam excelente fidelidade para a representação do comportamento de polinômios.

Dessa maneira, é conveniente olhar os coeficientes como coeficientes do filtro Wavelet, que se encontram localizados em uma matriz de transformação e são aplicados ao vetor do sinal (dados de entrada). A funcionalidade do filtro pode ser entendida como a superposição de dois efeitos, um funcionando de maneira a suavizar o sinal (de maneira similar a uma media móvel) e o segundo trabalhando no detalhe do sinal e permitindo ou não a sua passagem.

A matriz com os coeficientes da transformação do DWT é aplicada ao vetor de dados por um algoritmo chamado hierárquico ou ainda piramidal. Os coeficientes são arranjados de forma que as linhas ímpares contêm os coeficientes que agem no sentido de suavizar e as linhas pares no detalhamento. Inicialmente, a matriz é aplicada ao vetor original, então o vetor suavizado é dividido pela metade e a matriz é aplicada novamente, e em seguida o vetor é reduzido a um quarto e assim por diante, até o dado trivial remanescente. Assim, a aplicação da matriz chega a um nível de detalhamento de alta resolução, suavizando-o sucessivamente. A saída desse processo consiste no item remanescente adicionado dos componentes de detalhe.

#### **2.4.9. A transformada Wavelet discreta (DWT)**

A fundamentação da DWT remete a 1976, quando Croiser, Esteban e Galand desenvolveram uma técnica para decompor sinais discretos. Crochiere, Weber e Flanagan haviam feito trabalho similar na codificação de sinais de fala no mesmo ano. Este esquema de análise foi nomeado codificação de sub-banda. Em 1983, Burt definiu uma técnica bastante parecida e nomeou codificação piramidal, o que é também reconhecido por análise de multi-resolução. Após isto, em 1989, Vetterli e L  Gall desenvolveram o esquema, removendo a redund ncia existente no m todo piramidal.

#### **2.4.10. A codifica  o de sub-banda e a an lise de multi-resolu  o**

A id ia b sica aqui   a mesma do CWT. Uma representa  o de um sinal digital em um plano de tempo e escala, que   obtido atrav s do uso de t cnicas de filtragem digital. Relembrando que a CWT   a correla  o entre a fun  o Wavelet em diferentes escalas e o sinal em termos de escala (freq  ncia), uma forma de medi  o de similaridades. A an lise continua   computada modificando-se a escala da janela de an lise, dividindo a janela no tempo, multiplicando pelo sinal e integrando em todo o tempo.

No caso discreto, filtros de diferentes freq  ncias de corte s o usados para analisar o sinal em diferentes escalas. O sinal   submetido a uma s rie de filtros (passa-alta e passa-baixa).

A resolu  o do sinal, que no fim   uma medida da quantidade de informa  es de detalhe no sinal,   modificada pelas opera  es de filtragem e a escala   alterada pelas opera  es de “upsampling” e “downsampling”, que remetem respectivamente a adi  o de samples intermedi rios (geralmente zero ou a m dia) ou a redu  o (elimina  o) dos mesmos, ampliando ou reduzindo conseq entemente a resolu  o do sinal.

O procedimento inicia-se com a filtragem por um filtro passa-baixa de meia banda, com resposta ao impulso  $h(n)$ . A filtragem do sinal corresponde   opera  o matem tica de convolu  o com a resposta ao impulso do filtro. E pode ser assim representada:

$$x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot h[n-k]$$

O filtro passa-baixa de meia banda remove todas as frequências de ordem superior a metade da mais alta frequência do sinal. Se um sinal apresenta uma componente máxima de 1000 rad/s, equivale dizer que a filtragem remove qualquer componente superior a 500 rad/s.

De acordo com o critério de Nyquist, pode-se entender que após a filtragem de meia-banda, é possível eliminar metade dos “samples”, uma vez que a máxima frequência foi reduzida à metade. É executada então a operação de “subsample”, que elimina todos os samples ímpares, reduzindo a metade a taxa do sinal e conseqüentemente o comprimento. Note que o filtro mantém a taxa do sinal, embora corte as altas frequências, e ainda que na realidade a operação de “subsample” executa no sinal um efeito de escala. É importante perceber que a perda de informações aqui se dá apenas pela filtragem do sinal, e não pelo “subsample”, que é apenas uma adequação à nova composição do sinal e que respeita o critério de Nyquist.

O procedimento pode ser expresso matematicamente por:

$$y[n] = \sum_{k=-\infty}^{\infty} h[k] \cdot x[2n-k]$$

Uma importante propriedade da transformada discreta Wavelet é a relação entre a resposta em impulso dos filtros passa-alta e passa-baixa, estes não são independentes e se relacionam por:

$$g[L-1-n] = (-1)^n \cdot h[n]$$

Onde  $g[n]$  é o filtro passa-alta,  $h[n]$  o passa-baixa filter, e  $L$  o comprimento do filtro (em número de pontos). Perceba que um é uma versão inversa do outro. A conversão do passa-baixa para o passa-alta é dada pelo termo  $(-1)^n$ . Filtros que satisfazem essa

condição são comumente usados no processamento de sinais, e conhecidos por QMF (Quadrature Mirror Filters). Os dois filtros e o “subsampling” são expressos por:

$$y_{high}[k] = \sum_n x[n] \cdot g[-n + 2k]$$

$$y_{low}[k] = \sum_n x[n] \cdot h[-n + 2k]$$

A reconstrução do sinal torna-se simples quando os filtros passa-banda formam bases ortogonais. O procedimento é então seguido em ordem reversa para a reconstrução. O sinal a cada level é submetido ao “upsample” por dois e sintetizado pelos filtros  $g'[n]$  e  $h'[n]$  (passa-alta e passa-baixa respectivamente) e então somado ao detalhe. Um ponto interessante é a similaridade exceto pela questão temporal (reversa) entre ambos os casos. Então a reconstrução é executada para cada camada segundo:

$$x[n] = \sum_{k=-\infty}^{\infty} (y_{high}[k] \cdot g[-n + 2k]) + (y_{low}[k] \cdot h[-n + 2k])$$

Contudo, quando o filtro não é um passa-banda ideal, torna-se impossível uma perfeita reconstrução. Embora não seja possível construir o filtro ideal, sob certas condições é possível encontrar filtros que façam a reconstrução bem próxima do ideal.

Note que para um sucessivo “subsample” de base 2, o comprimento do sinal deve ser uma potencia de 2 ou um múltiplo de uma potencia de 2, de maneira que as decomposições sejam eficientes. O comprimento do sinal determina quantas vezes o mesmo pode ser decomposto. Um sinal de 1024 pontos, por exemplo, pode sofrer uma decomposição de 10 fases.

#### **2.4.11. Os coeficientes DWT**

A interpretação dos coeficientes da DWT nem sempre é muito simples, isso devido à sua apresentação bastante peculiar, onde os coeficientes de cada nível são concatenados, iniciando-se pelo ultimo nível. Este arranjo pode ser melhor explicado através de um exemplo.

Supondo um sinal de 256 samples de comprimento, adquirido a uma taxa de 10MHZ, de onde serão obtidos os coeficientes DWT. Pelo critério de Nyquest, a componente de maior frequência existente no sinal é de 5MHZ. Na primeira iteração com os filtros passa-baixa e passa-alta, a saída de ambos sofre o “subsample” de 2. O sinal do passa-alta compõe o primeiro nível dos coeficientes, ali existirão 128 samples representativos da faixa de frequência [2,5 5]MHZ. O sinal filtrado pelo passa-baixa (também com 128 samples) representa a faixa [0 2,5]MHZ e que será novamente submetido às filtrações e ao “subsample”, de onde surge o segundo nível de coeficientes, com 64 samples, e sucessivamente perfazendo o terceiro nível com 32 samples e assim por diante. Esse procedimento pode ser executado até que haja 1 único coeficiente.

Assim, o que se obtém é uma pirâmide com 1 coeficiente no nível 8, 2 coeficientes no nível 7, 4 coeficientes no nível 6, 8 coeficientes no nível 5, 16 coeficientes no nível 4 e conforme dito anteriormente nos demais níveis.

Assim, para as menores frequências, são usados menos coeficientes, a resolução temporal se reduz conforme a frequência decresce, isto não afeta no entanto a quantidade de informações naquele espectro, uma vez que apenas as informações redundantes são eliminadas.

#### **2.4.12. A transformada rápida Wavelet (FWT)**

Em geral, a matriz DWT não é esparsa o que acrescenta a mesma complexidade que para o caso de uma transformada discreta de Fourier. E a resolução se mostra também de maneira similar, através da fatoração de DWT em produtos de pequenas matrizes

esparsas utilizando-se de propriedades de similaridade. O algoritmo resultante requer apenas  $n$  operações para transformar um vetor de  $n$  dados.

#### **2.4.13. Transformada Wavelet Packet (WPT)**

Atualmente a transformada Wavelet é uma subdivisão de uma transformada mais versátil, a transformada Wavelet Packet.

Wavelets Packets são constituídas por combinações lineares particulares de Wavelets, que formam bases caracterizadas pela retenção de propriedades como, por exemplo, a ortogonalidade e a localização das Wavelets originais. Os coeficientes dessa combinação linear são obtidos através de um algoritmo recursivo que faz de cada coeficiente da Wavelet Packet a raiz da sua própria árvore de análise.

#### **2.4.14. Formas de onda adaptadas**

Dado o infinito leque de escolhas possível em relação às funções de base, seria conveniente encontrar a melhor delas para o problema particular. Uma base de forma de onda adaptada (“adapted waveform”) é a melhor base para a representação de um dado sinal. Essa escolha se baseia em informações substanciais a respeito do sinal, e no fato de a base descrita ser eficiente, ou seja, apresentar poucos termos para conseguir representar o sinal.

De acordo com Wickerhauser, algumas propriedades são desejáveis:

- agilidade do cálculo dos produtos internos entre as funções da base;
- rapidez na superposição das funções de base;
- boa localização espacial;
- boa localização espectral;
- boa independência linear;

No caso da análise a ser executada neste trabalho, não será utilizada uma forma adaptada, mas sim uma família desenvolvida e que satisfaça os anseios do trabalho.

#### 2.4.15. Exemplos de aplicação

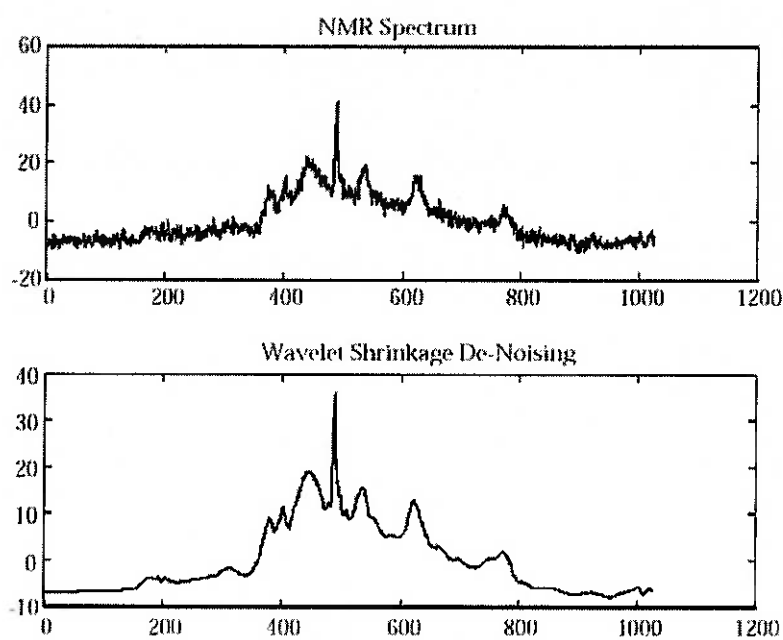
A seguir, apenas para ilustrar o vasto campo de aplicação da transformada Wavelet, serão apresentados alguns exemplos:

Fig. 2.8 – Compactação de imagens



A figura acima mostra uma impressão digital original (a esquerda) e uma impressão reconstruída a partir de uma compressão Wavelet de 26:1, utilizada pelo FBI.

Fig. 2.9 – Redução de ruído



Repare que o sinal superior apresenta alta energia em forma de ruído, e que após a filtragem o mesmo aparece limpo, esse é um exemplo de aplicação da transformada Wavelet como filtro em um sinal de ressonância magnética do VA Medical Center, de São Francisco – EUA.

#### **2.4.16. Algoritmo da implementação Wavelet unidimensional em Matlab**

Nesta seção, o algoritmo Wavelet será explicado com foco na implementação do mesmo em Matlab. O software Matlab foi escolhido para essa execução por possuir uma excelente caixa de ferramentas Wavelet, além da praticidade e versatilidade usual do software.

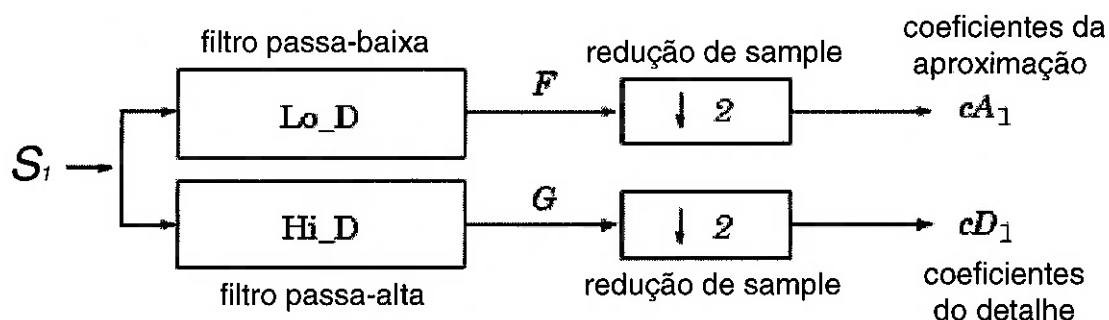
##### **2.4.16.1. Primeiro passo**

Inicialmente, tem-se um sinal de entrada discreto a ser processado, ao qual chamaremos por  $s_e$  este sinal apresenta comprimento  $n$  e assim sendo, a transformada DWT consiste em um máximo de  $\log_2 n$  estágios. Executando a primeira transformação, serão produzidos dois conjuntos de coeficientes: os de aproximação (denominaremos  $cA_1$ ) e de detalhe (denominaremos  $cD_1$ ). Esses vetores, obtidos pela convolução com um filtro passa-baixa (Lo\_D) para a aproximação, e com um filtro passa-alta (Hi\_D) para o detalhe, seguido de uma aproximação média entre dois samples.

De maneira ilustrativa tem-se para a transformação unidimensional:

Fig. 2.10 – Transformada Wavelet



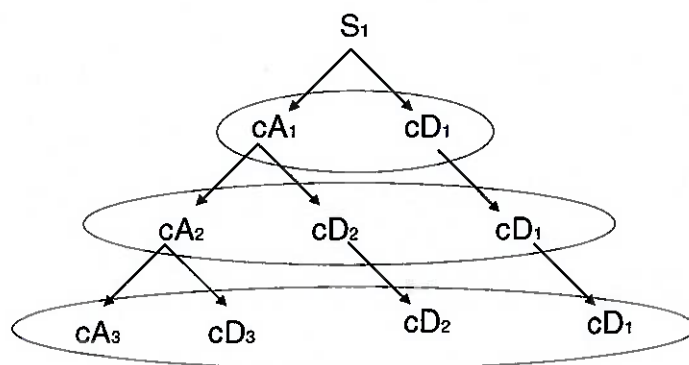


Chamando o comprimento do filtro de  $2N$  (dado que  $s_I$  tem comprimento  $n$ ), os sinais  $F$  e  $G$  têm comprimento  $n+2N-1$ , e os vetores de coeficientes  $cA_1$  e  $cD_1$  tem o comprimento do maior inteiro menor que  $\left(\frac{n-1}{2}\right)$  adicionado de  $N$ .

A partir do mesmo esquema, o segundo passo divide o sinal de aproximação ( $cA_1$ ) em um no vetor de aproximação ( $cA_2$ ) e ( $cD_2$ ), e os passos subseqüentes repetem esta interação com os vetores de aproximação da interação anterior.

O seguinte esquema ilustra esse algoritmo:

Fig. 2.11 – Árvore Wavelet



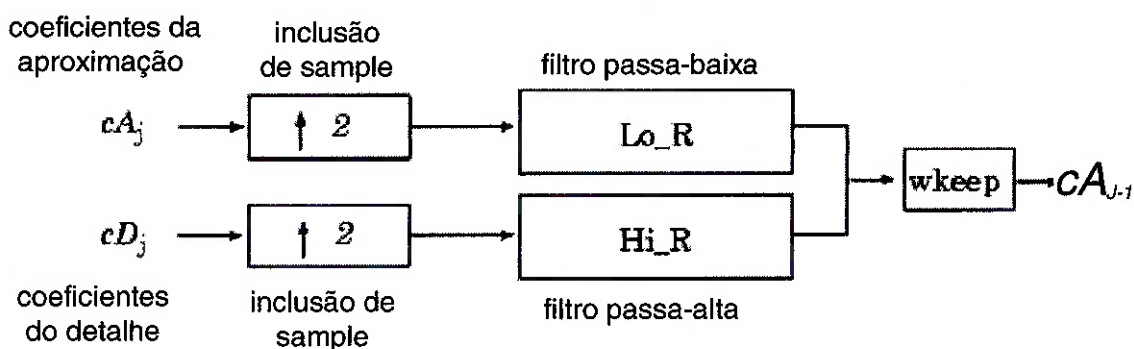
Repare que o sinal  $S_1$  pode ser considerado um vetor de coeficientes iniciais  $cA_0$ .

#### 2.4.16.2. A transformada inversa

De maneira inversa, iniciando de  $cA_j$  e  $cD_j$ , a IDWT reconstitui  $cA_{j-1}$  inserindo zeros no local dos samples ímpares e convolucionando o vetor resultante com os filtros de reconstrução.

De maneira ilustrativa tem-se para a transformação inversa unidimensional:

Fig. 2.12 – Transformada Wavelet inversa



A operação denominada “wkeep” extrai a parcela central do vetor, com o comprimento conveniente e a operação com os filtros indica uma convolução.

No caso bidimensional, um algoritmo similar é possível para Wavelets bidimensionais, obtidas por produtos tensoriais de Wavelets unidimensionais.

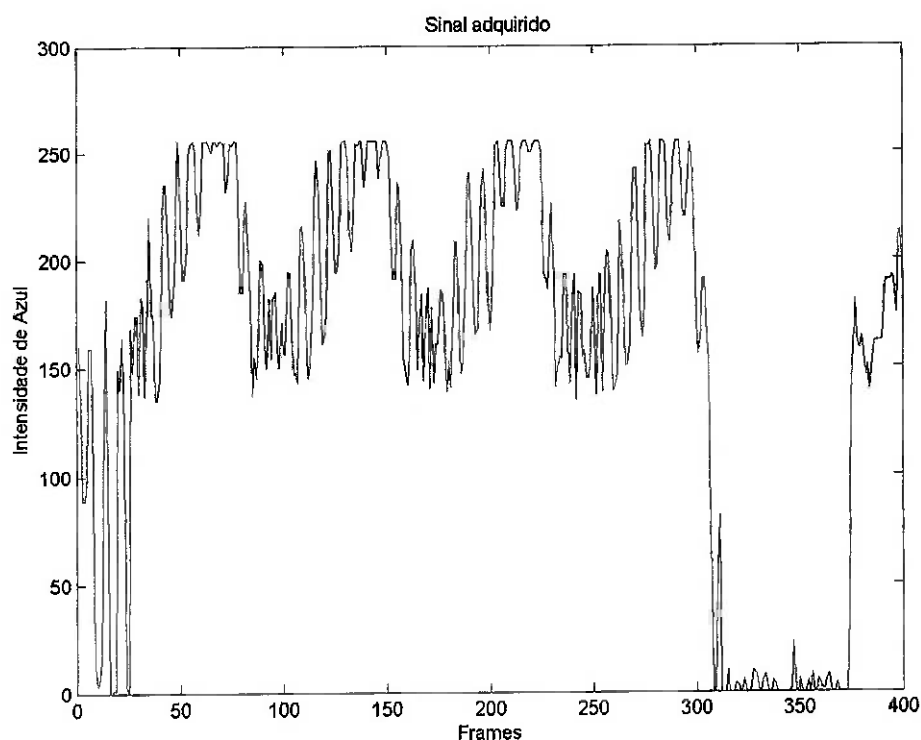
### 2.4.17. Implementação

Nesta seção será apresentada uma primeira filtragem via Wavelet, através da ferramenta matlab, serão ressaltados os comandos e a sequência lógica do procedimento para um pixel. As linhas de comando para o vídeo completo estão em anexo.

#### 2.4.17.1. O sinal

Como já realizado na etapa anterior com a implementação do filtro FIR, inicialmente será apresentado o sinal, conforme se encontra no vídeo ao longo do tempo e aqui em escala de frames.

Fig. 2.13 – Sinal adquirido



Ao sinal será atribuída a nomenclatura  $s$ .

Encontrando o comprimento do sinal:

```
l_s = length(s);
```

Inicialmente, será executada a decomposição de 1ª ordem, para que o procedimento seja documentado passo a passo:

```
[cA1,cD1] = dwt(s, "dmey");
```

Nesse caso, será utilizada a família discreta de Meyers, conforme critérios explicados anteriormente. Futuramente, para fim de comparação serão apresentados resultados utilizando outras famílias.

Repare que aqui foram gerados os coeficientes  $cA_1$  e  $cD_1$ , representando a aproximação e o detalhe respectivamente.

Agora, a partir dos coeficientes serão reconstruídas as funções de aproximação ( $A_1$ ) e de detalhe ( $D_1$ ).

```
A1 = upcoef("a", cA1, "dmey", 1, l_s);
```

```
D1 = upcoef("d", cD1, "dmey", 1, l_s);
```

Caso fosse necessário, a partir dos coeficientes poder-se-ia ainda executar a transformada inversa e obter novamente o sinal.

```
A0 = idwt(cA1,cD1, "dmey", l_s);
```

Para executar a segunda transformação, seria possível executar a mesma transformação utilizando porem como entrada a aproximação do primeiro nível e assim sucessivamente.

Existe ainda uma outra função, que executa uma transformação de nível  $n$  no sinal primário.

$[C,L] = \text{wavedec}(s, n, \text{"dmey"})$ ;

Esse foi o procedimento adotado, por executar de forma compacta a enésima transformação.

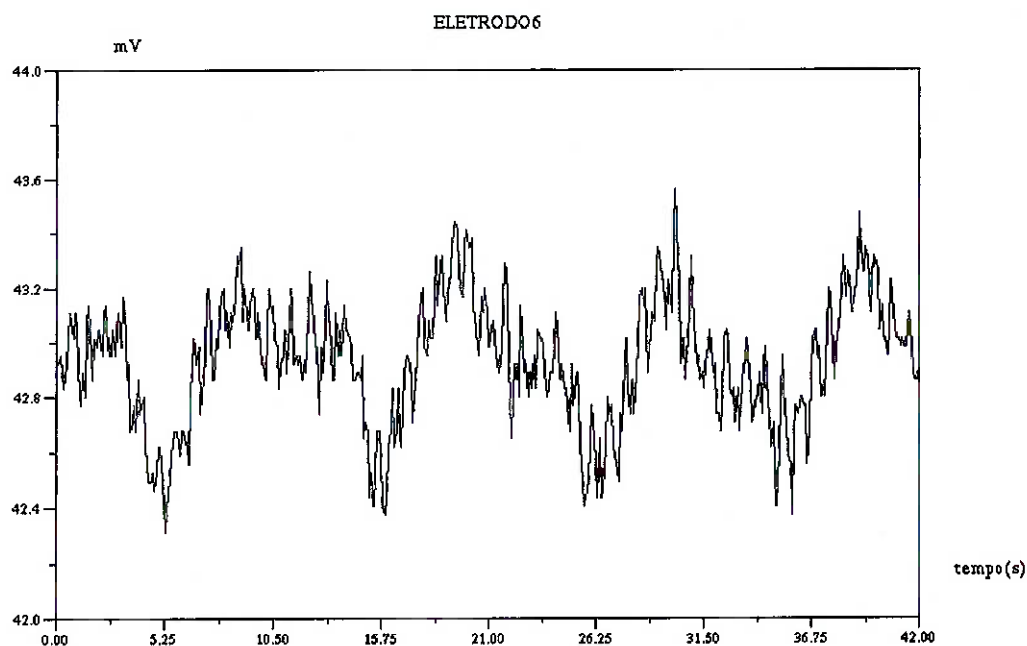
Em seguida se faz necessário apenas observar os resultados e implementar a metodologia para filtrar todos os pixels do vídeo. O algoritmo completo encontra-se no anexo e os resultados obtidos na seção de resultados.

### 3. TRATAMENTO DE UM SINAL

Nesse item será abordada a filtragem de um sinal que representa a variação da tensão em um eletrodo no tempo, ou em termos de imagem, o sinal pode ser considerado equivalente ao pixel no domínio do tempo. A partir da rotina exposta no item anterior o sinal será filtrado através do procedimento FIR, conforme as informações estabelecidas. Para a transformada Wavelet, essa primeira análise não será repetida, os teste para a análise Wavelet serão executados diretamente com o pixel visto que o sinal do eletrodo serve inicialmente para uma primeira avaliação dos parâmetros do filtro FIR e para a primeira aplicação do sistema de processamento.

A seguir será ilustrado o sinal de entrada proveniente de um eletrodo do Tomógrafo:

Fig. 3.1 – Sinal de entrada



O sinal é marcado pela presença de duas componentes de frequência, uma mais baixa que executa aproximadamente 4 períodos no intervalo de aquisição (aproximadamente 0,1Hz) e outra com mais de 10 ciclos em um período de 10 segundos (superior a 1 Hz).

Assim observa-se que a frequência da ordem de 0,5 Hz satisfaz o intervalo entre as duas frequências.

### **3.1. Determinação da ordem do filtro FIR**

A definição da faixa de banda de 0,5 Hz é satisfatória para o caso proposto, pois seriam impostos ganhos diferenciados para frequências de 0,0 a 0,5 Hz (aqui se encaixa o sinal de menor frequência), de 0,5 a 1,0 Hz (faixa de corte do filtro) e de forma análoga compreendendo as faixas sucessivas de amplitude 0,5 Hz. Assim, impondo ganho  $k = 1$  para a faixa de passagem de frequência e  $k = 0$  para a faixa de frequências a serem descartadas, e por fim ganho  $k = 0,4$  (ganho que suaviza o truncamento para filtros desse porte) para a faixa que inclui a frequência de corte.

Assim, para que a faixa de frequência de 12 Hz (imposto pela taxa de aquisição) seja coberta por bandas de 0,6Hz tem-se o filtro de ordem 20, ou seja, 20 faixas de 0,6 Hz compondo todo o espectro. É necessário que exista uma simetria entre por parte do filtro, assim, é conveniente que a ordem implementada seja de valor ímpar, assim será estudada a implementação de um filtro FIR de ordem 21.

#### **3.1.1. Determinação da ordem do filtro Wavelet:**

Conforme mencionado para o caso do filtro FIR, é de se imaginar que a frequência de corte escolhida deva se situar entre as faixas de 0,2 Hz e 1 Hz. Para o mecanismo Wavelet, que funciona sempre como um filtro de meia banda. Teremos então para o sinal de 24 frames por segundo uma frequência máxima de 12 Hz (pelo critério de Nyquist), assim, a primeira atuação do filtro Wavelet reduzirá a 6Hz a frequência da aproximação, em seguida a 3Hz, a terceira aplicação atingirá a faixa dos 1,5Hz e após a quarta aplicação, a aproximação compreenderá o intervalo de 0 a 0,75Hz, faixa de observação que interessa ao propósito do trabalho. Assim, fica definida a quarta filtragem como fonte da saída do sinal filtrado.

### **3.2. Estendendo o tratamento para o vídeo**

Para que a filtragem do vídeo seja completa, a metodologia de filtragem devera ser imposta a cada pixel e para tanto é necessário que o vídeo seja decomposto, filtrado e posteriormente recomposto.

No caso do filtro Wavelet dada a ordem cronológica das simulações (primeiramente foram obtidos os resultados FIR) a etapa de testes com o sinal do eletrodo não será executada, assim, os resultados da filtragem via Wavelet serão apresentados a partir do tratamento deste mesmo pixel.

#### **3.2.1. Manipulação do vídeo**

O vídeo adquirido a partir do Tomógrafo por Impedância Elétrica deve ser tratado nesse trabalho pixel a pixel, o que agrega pequenas dificuldades em termos do processamento da imagem.

A primeira dessas dificuldades trata da desfragmentação do vídeo em um conjunto de imagens seqüenciais em um formato adequado. O segundo aspecto se relaciona ao fato de que a cada imagem corresponde uma matriz ou no caso de um sistema RGB (Red, Green, Blue), três matrizes (nesse caso de 640x560) com definição de oito bits (256 tonalidades), essas matrizes deverão ser tratadas sequencialmente pixel a pixel, ou seja, coordenada a coordenada com as matrizes dos instantes subseqüentes. Dessa maneira, a partir de um sistema robusto de filtragem, a dificuldade do procedimento é computacional, visto que serão primeiramente organizados e em seguida tratados 358.400 (640x560) sinais para cada uma das três cores, correspondendo a um total de 1.075.200 sinais a serem filtrados.

Por fim, os sinais devem ser novamente organizados, de maneira a obter o vídeo tratado.



### **3.2.2. Separação dos arquivos de imagem**

Através do software “VideoMach”, o arquivo de vídeo adquirido foi dividido em uma sequência de imagens, a taxa de aquisição dessas imagens foi fixada em 24 frames por segundo (taxa de aquisição da visão humana) para que não se perdesse em qualidade visual na execução, o que pelo critério de “Nyquist” permite a determinação no sinal de frequências de até 12Hz, suficientemente alta para as frequências do nosso sistema (imaginando uma frequência cardíaca de 120bpm) de ordens dificilmente superiores a 2Hz. O fator limitante às taxas de aquisição mais altas se dá pelas dificuldades computacionais. Perceba que as simulações são executadas em ambiente Windows e através de softwares como o Scilab, o que acarreta em um processamento bastante lento se comparado a linguagens computacionais, porém, em atividade experimental, a utilização desses softwares permite uma maior interatividade e principalmente uma maior observabilidade durante o passo a passo do processamento.

Os frames foram armazenados em arquivos de imagem no formato bitmap (extensão BMP) por se tratar de um formato não compactado e de relativa facilidade de manipulação embora ocupem conseqüentemente um maior espaço físico no disco rígido do computador, alocando menos recursos de processamento.

### **3.2.3. Criação das matrizes**

O formato RGB organiza a imagem em um conjunto de 3 matrizes com valores inteiros que variam de 0 a 255 (sistema de 8 bits) em cada elemento correspondente a um pixel. Cada uma das matrizes representa respectivamente uma das três cores RGB (vermelho, verde e azul) e o valor atribuído a cada posição equivale a uma tonalidade daquela cor, de maneira que um pixel com valores atribuídos de (0, 0, 0) corresponde à cor branca, um pixel atribuído com (255, 255, 255) à cor preta, um pixel com valores (255, 0, 0) à cor vermelha e assim por diante.

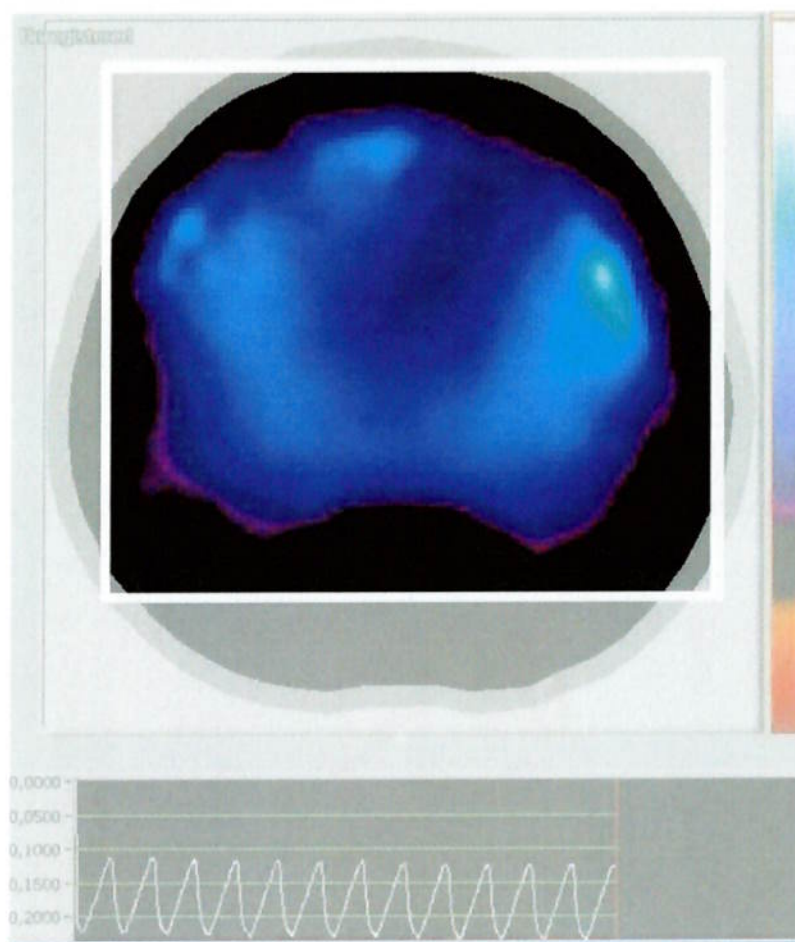
Através do Matlab as imagens em bitmap foram então convertidas nas três matrizes do sistema RGB e unidas em matrizes tri-dimensionais, onde linha e coluna representam a posição de cada pixel e a terceira dimensão representa o tempo, estando uma matriz

separada da outra por um período de 1/24 segundos. Foram então criadas 3 matrizes representativas de cada uma das cores.

#### 3.2.4. Redução do processamento

Para uma redução do processamento, a imagem pode ser dividida de acordo com a necessidade ou não de filtragem, para que só sejam filtrados os pixels que efetivamente variam no tempo de acordo com o objeto de estudo (frequências de respiração e perfusão), assim, é proposta a seguinte divisão da imagem em um quadrante ativo para o filtro, na imagem, o interior da margem branca representa esse quadrante:

Figura 3.2 – Quadrante a ser processado



Dessa maneira o numero de sinais a ser filtrado em cada uma das cores RGB foi reduzido de 35.840 (imagem de 640x560) para 13.900 (340x385), ou seja, cerca de 39% da quantidade inicial. Em termos computacionais essa redução reflete proporcionalmente o tempo de processamento. A comparação visual pode ser feita pela figura acima, onde os pixels filtrados correspondem à área em destaque na figura.

### **3.2.5. Testando os filtros para um pixel**

Para que a rotina seja executada por completo, a fim de reduzir a possibilidade de não funcionamento do processo de filtragem, aleatoriamente foram filtrados alguns pixels, de maneira a observar o comportamento do sinal antes e depois da filtragem. Essa simulação que já havia sido feita com o sinal provindo de um eletrodo do tomógrafo será submetida a um novo teste, uma vez que agora trata-se de um pixel com definição de 8 bits, o que faz com que as variações sejam muito mais discretas do que no sinal anteriormente testado, trata-se agora de um valor inteiro e que varia em intensidade de 0 a 255. Assim, supõe-se evitar eventuais problemas com a simulação que pudessem ser percebidos apenas após todo o processamento.

No caso do filtro Wavelet, a primeira análise será executada já com o sinal do pixel, uma vez que o mecanismo de análise já foi desenvolvido com o filtro FIR.

## 4. RESULTADOS E ANÁLISES:

### 4.1. Sinal do eletrodo com filtro FIR:

A partir do sinal apresentado na seção anterior foram obtidas as respostas ao sistema imposto pela convolução com a inversa da Transformada de Fourier de um sinal imposto. Serão apresentados nessa seção os resultados obtidos em cada etapa dessa passagem.

Para o filtro passa baixa:

O filtro no domínio da frequência:  $H$

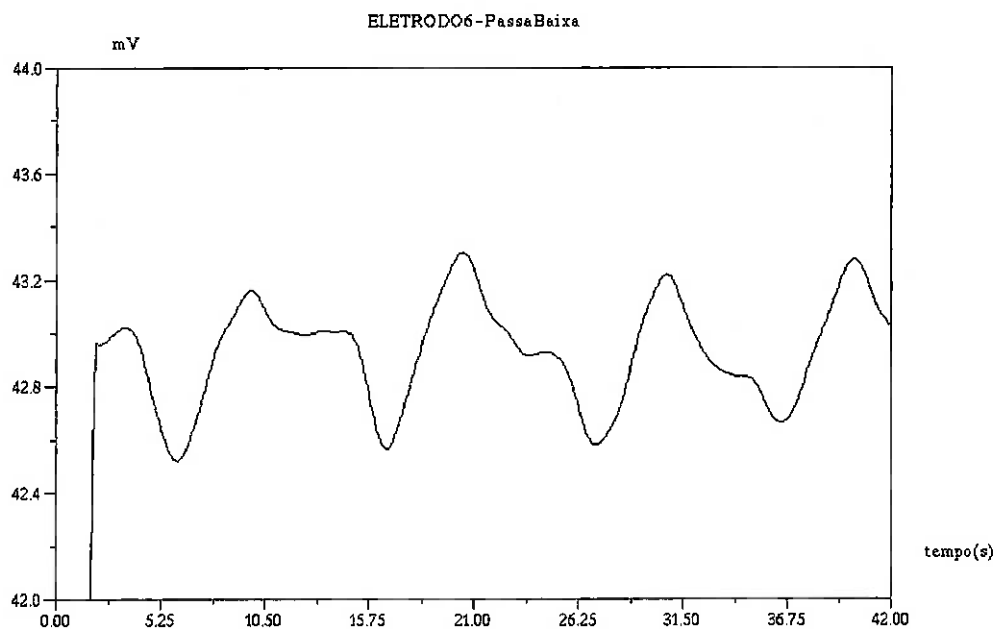
$H = [1.; 0.4; 0.; 0.; 0.; 0.; 0.; 0.; 0.; 0.; 0.]$

O filtro após a aplicação da Transformada Inversa de Fourier:  $h$

$h = [0.0099493; 0.0132964; 0.0196933; 0.0285714; 0.0391421; 0.0504659; 0.0615368; 0.0713710; 0.0790948; 0.0840218; 0.0857143].$

E o sinal filtrado, obtido após a convolução:

Fig. 4.1 – Sinal com Passa Baixa



Note que nos primeiros instantes o valor de tensão encontra-se bem abaixo do valor médio do sinal, isso ocorre porque os primeiras  $M-1$  (sendo  $M$  a ordem do filtro) pontos devem ser descartados quando se implementa o filtro FIR, já que durante as primeiras iterações o sinal ainda não possui um momento passado, e o filtro atua a partir dos  $M-1$  instantes anteriores para fornecer o  $M$ -ésimo instante. Daí a inconveniência de utilizar filtros de ordem muito alta para sinais curtos.

É perceptível que a componente de maior frequência foi eliminada, restando ao sinal a componente de menor frequência.

De modo similar foi aplicado o filtro passa alta, de onde se obteve:

O filtro no domínio da frequência:  $H$

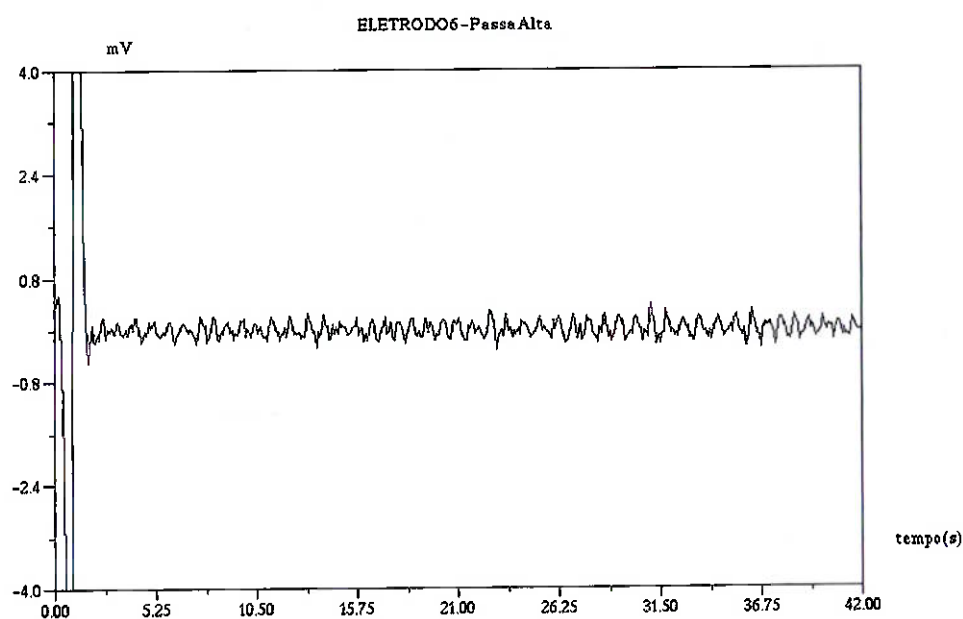
$$H = [0.; 0.4; 1.; 1.; 1.; 1.; 1.; 1.; 1.; 1.; 1.]$$

O filtro após a aplicação da Transformada Inversa de Fourier:  $h$

$$h = [0.0088856; 0.0038649; -0.0057304; -0.0190476; -0.0349036; -0.0518893; \\ -0.0684957; -0.0832470; -0.0948327; -0.1022232; 0.8952381]$$

E o sinal filtrado, obtido após a convolução:

Fig. 4.2 – Sinal com Passa Alta



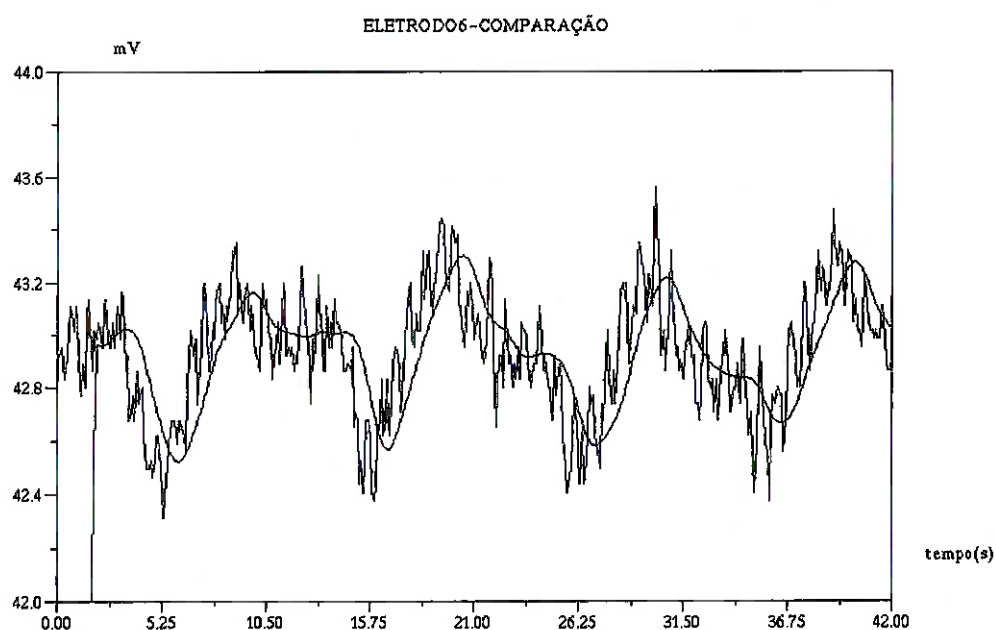
Os resultados obtidos com a aplicação do Filtro Passa Alta foram condizentes com os obtidos anteriormente, a componente de baixa frequência foi eliminada e a componente de maior frequência pode ser bem observada no sinal.

De maneira análoga os primeiros instantes do sinal não podem ser considerados pelo mesmo motivo já explicitado na análise do Filtro Passa Baixa.

Com o intuito de comparar o sinal de entrada e o sinal de saída foram plotados os gráficos dos sinais antes da filtragem e após a filtragem.

Como a componente de baixa frequência tem amplitude bem maior do que a componente de alta frequência, por motivo de escala o sinal de saída a ser comparado foi o filtrado pelo Passa Baixa.

Fig. 4.3 – Comparação de Sinais



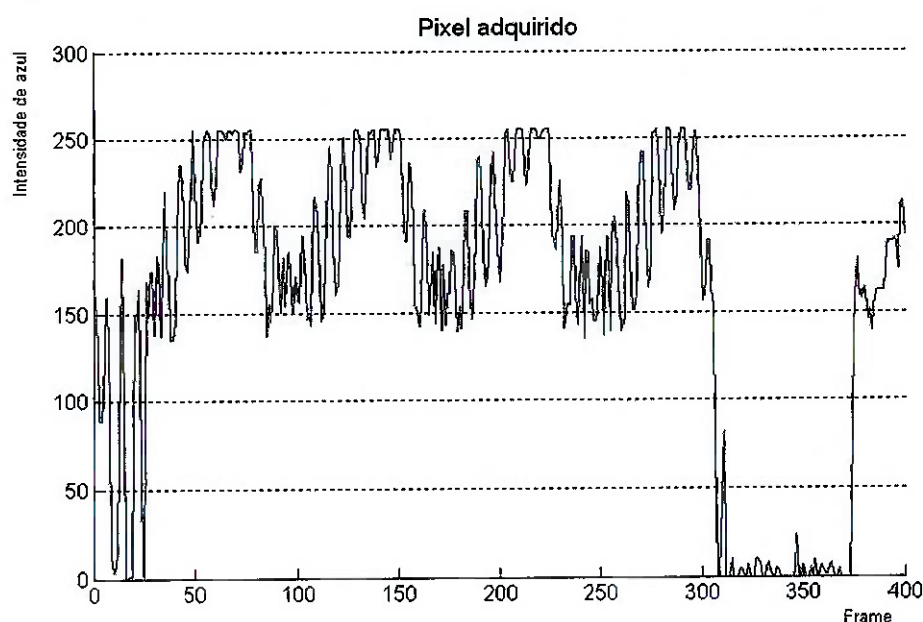
O sinal mais suave (filtrado) sobreposto ao sinal turbulento (não filtrado) nos dá a exata idéia da utilização do filtro.

É notável o atraso de fase do sinal filtrado, que segundo a teoria deve ser linear com a frequência. É perceptível ainda a manutenção da amplitude em relação ao sinal de entrada (ganho  $k = 1$ ) implicado a banda de frequência baixa.

#### 4.2. Sinal do pixel com filtro FIR:

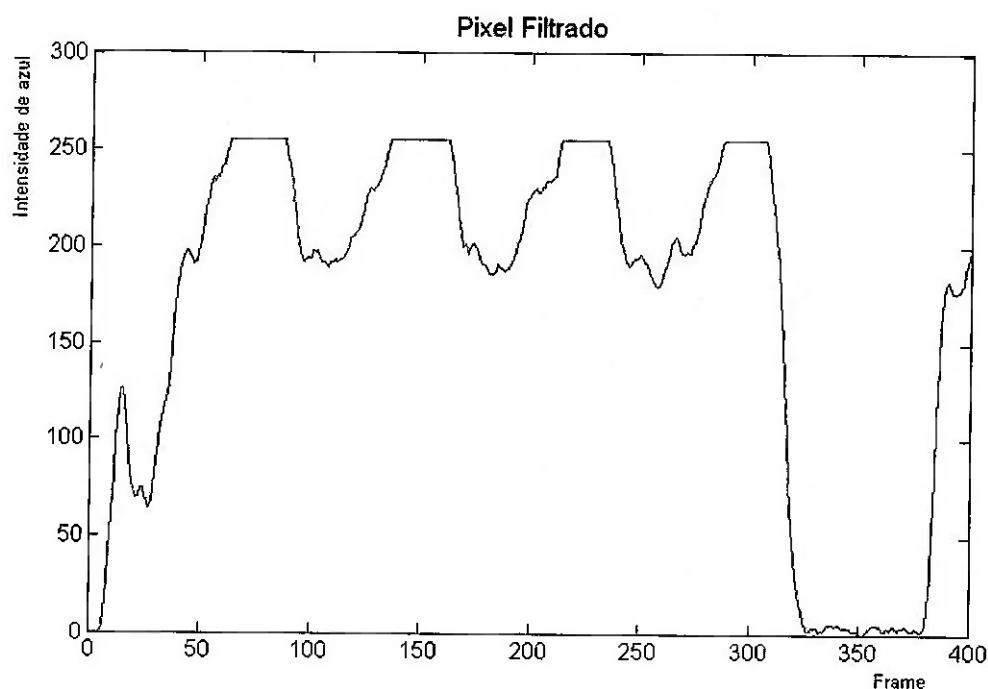
De maneira similar ao apresentado na filtragem do sinal do eletrodo, alguns pixels foram filtrados, a seguir será explicitado o resultado da filtragem do pixel posicionado nas coordenadas (145,215), o único agravante e esse é o motivo para que sejam executados novos testes é o fato de tratarmos agora de uma variável com baixa definição, que sempre possui valores inteiros e na faixa que varia de 0 a 255.

Fig. 4.4 – Sinal do pixel



E após o processamento, de maneira exatamente homologa a anterior, com o sinal filtrado em passa baixa.

Fig. 4.5 – Pixel filtrado



Repare que nos instantes iniciais a intensidade não reflete a real condição de filtragem, devido à ausência de instantes iniciais suficientes para suprir de informações os coeficientes do filtro, o comprimento dessa zona afetada é causado pela ordem do filtro. Nas figuras a seguir serão plotadas as Transformadas de Fourier (curvas FFT) normalizadas do mesmo Pixel anteriormente estudado, a fim de consolidar qualitativamente a análise do efeito do filtro FIR sobre o Pixel.



Figura 4.6 – FFT antes da filtragem

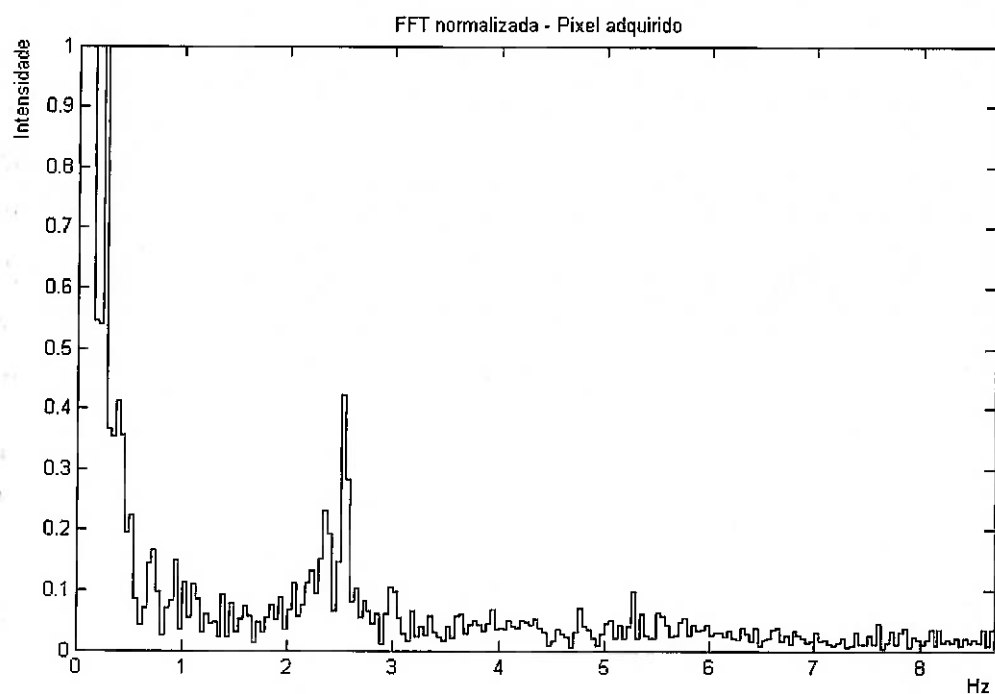
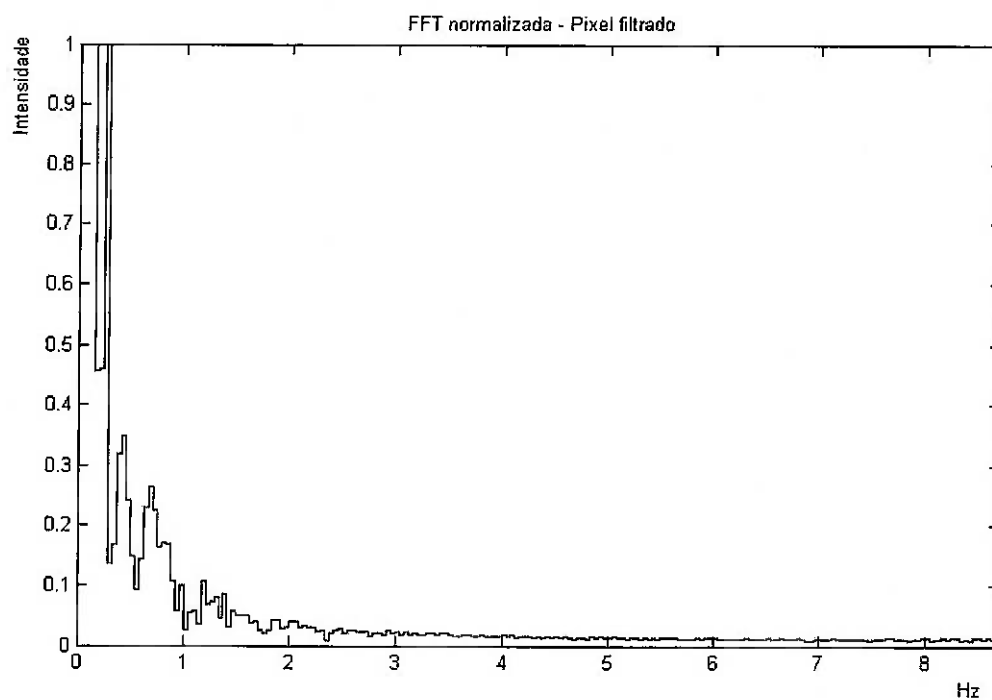


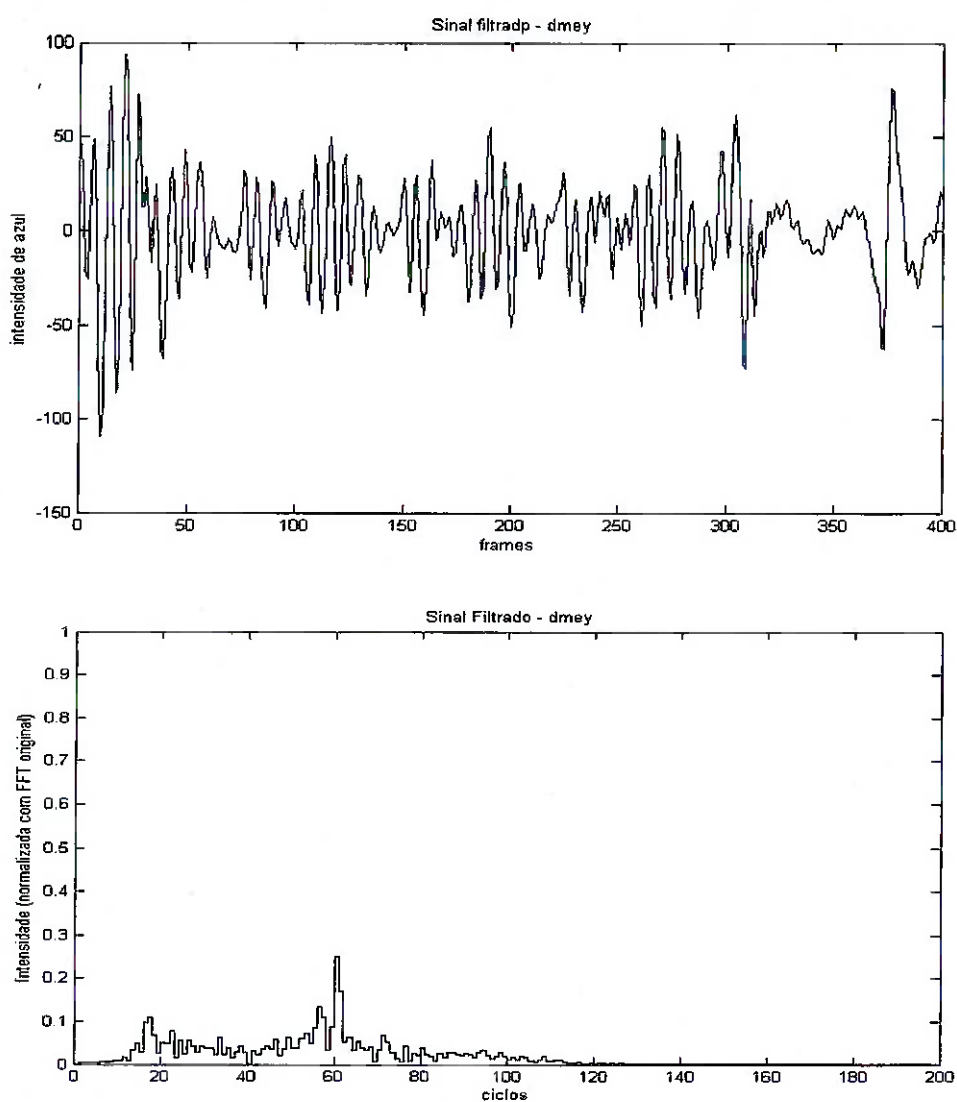
Figura 4.7– FFT após a filtragem



### 4.3. Sinal do pixel com filtro Wavelet:

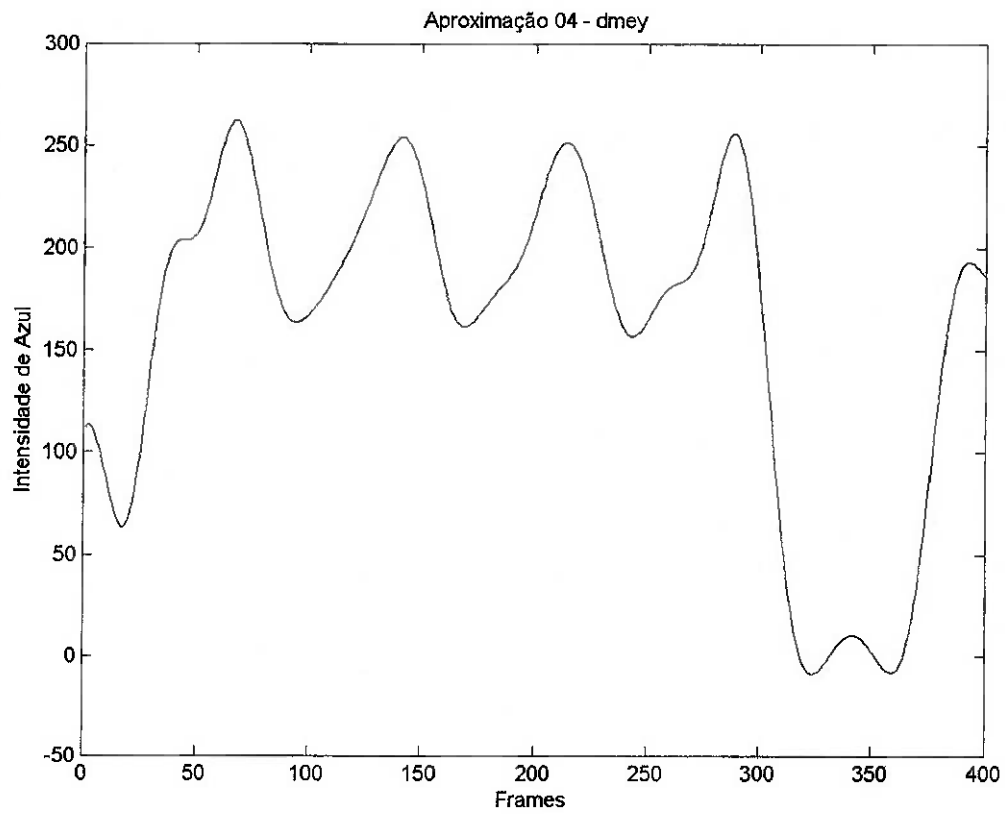
O mesmo pixel filtrado em FIR, foi agora filtrado. A família utilizada para a decomposição foi a da transformada discreta de Meyer, a seguir o sinal do coração (que é a soma dos detalhes da segunda, terceira e quarta filtragem) e a sua respectiva FFT.

Figura 4.8 – Sinal filtrado e FFT



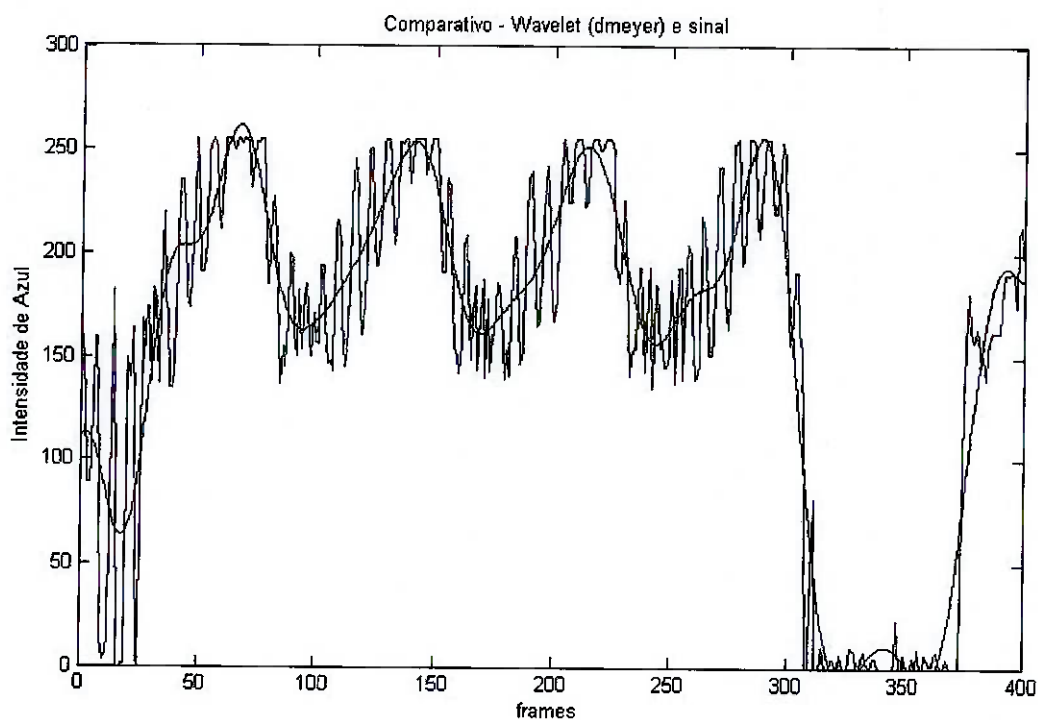
E após o processamento, de maneira exatamente homologa a anterior, com o sinal da respiração.

Fig. 4.9 – Pixel filtrado



E ainda, para propiciar uma melhor observação, é conveniente plotar o sinal filtrado em sobreposição ao sinal original.

Fig. 4.10 – Pixel filtrado



Repare que o sinal filtrado não apresenta defasagem temporal (característica supracitada da Wavelet) e que a aproximação Wavelet trabalha em cima da aproximação média da região.

Comparativamente, teremos as FFT dos sinais antes da filtragem e após o procedimento:

Fig. 4.11 – FFT do sinal adquirido

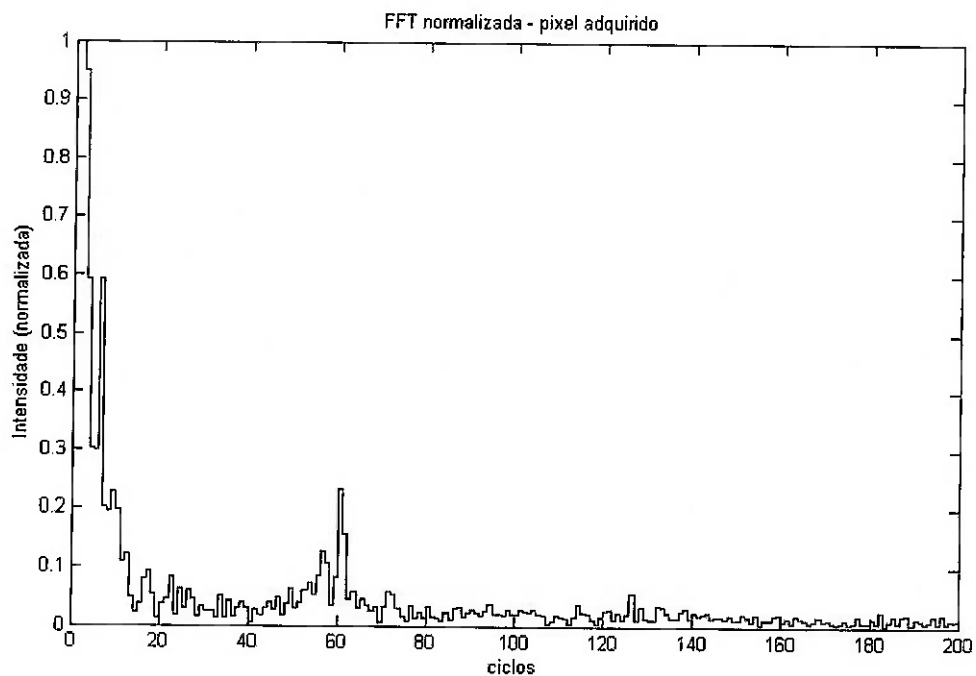
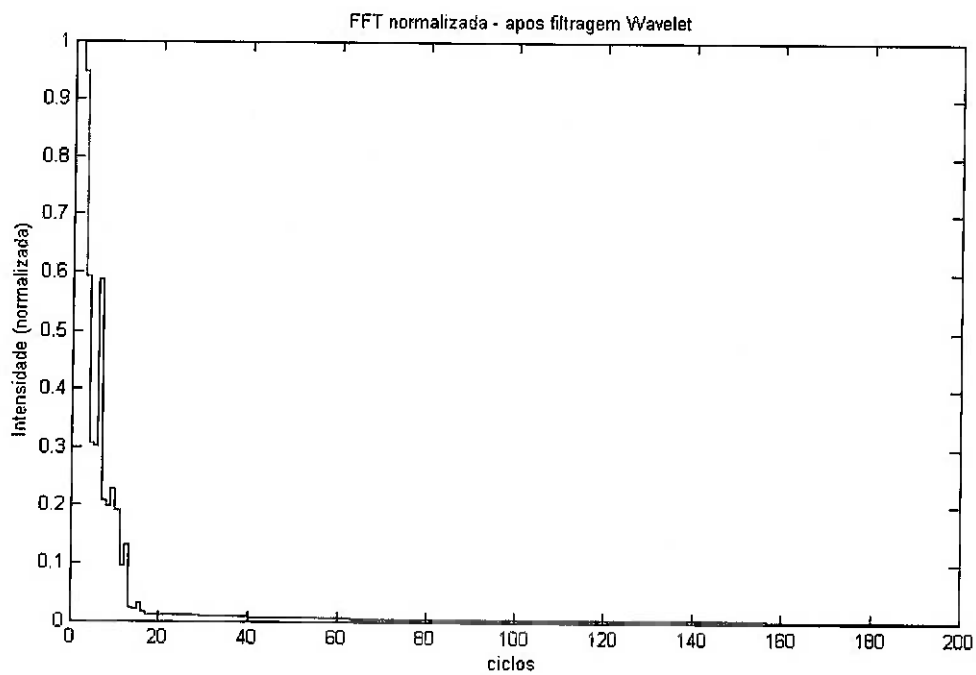


Fig. 4.12 – FFT do sinal filtrado



Repare que o segundo pico, que no primeiro gráfico encontra-se próximo a taxa de 60 ciclos. A unidade de ciclos refere-se ao comprimento total do sinal.

#### 4.4. Diferentes filtragens Wavelet para o pixel:

Aqui serão expostas diversas filtragens via Wavelet para o pixel avaliado, sempre na mesma decomposição, no caso em questão a quarta passagem do filtro.

Fig. 4.13 – Filtrado por Coiflet2

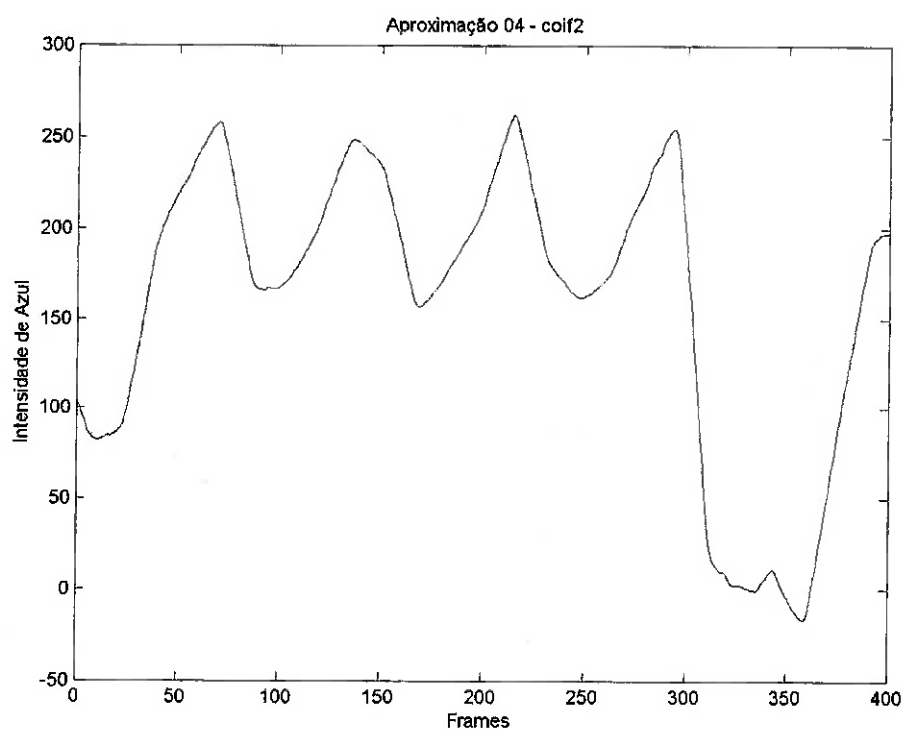


Fig. 4.14 – Filtrado por Daubechies3

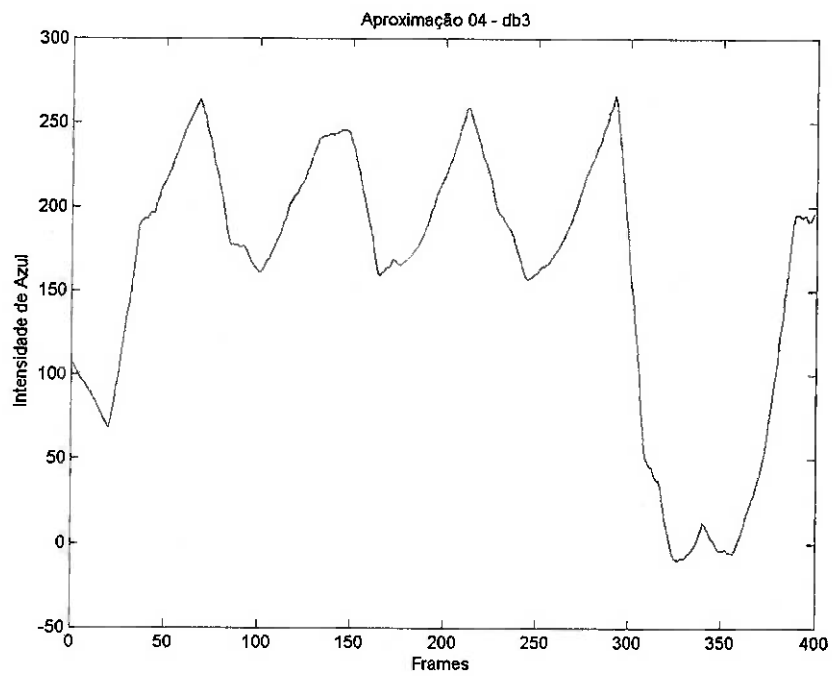


Fig. 4.15 – Filtrado por Haar

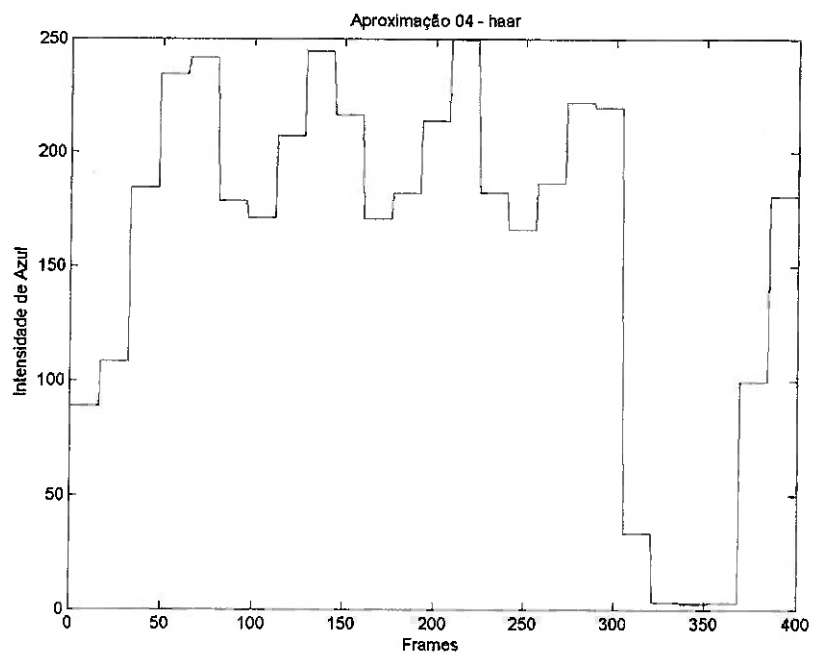


Fig. 4.16 – Filtrado por Symlet7

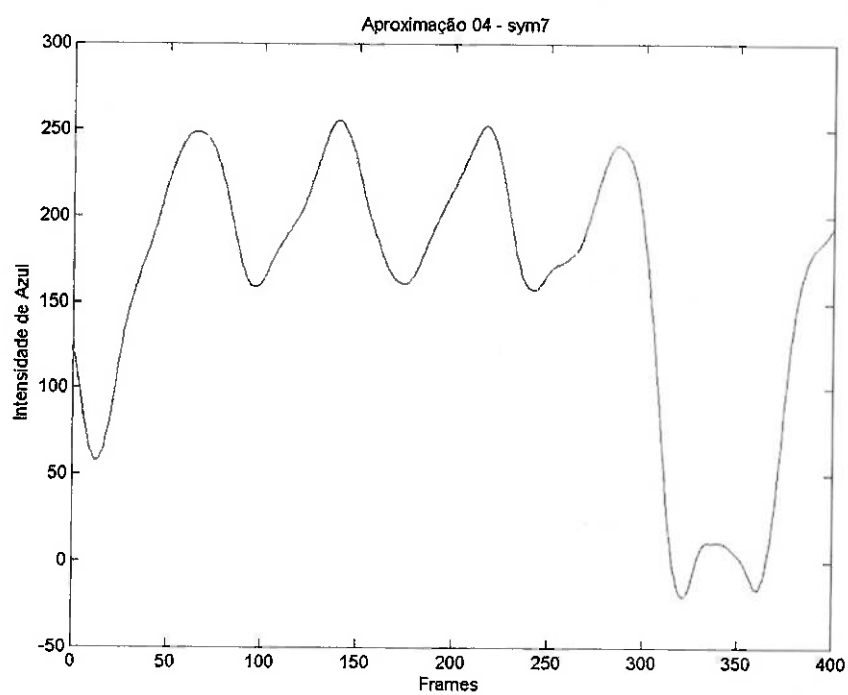


Fig. 4.17 – Filtrado por Daubechies2

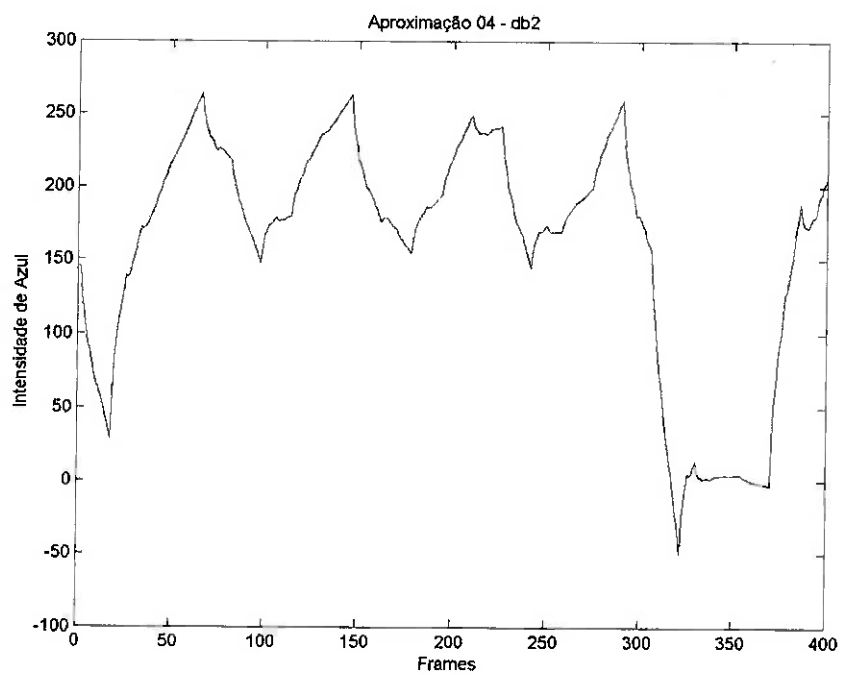
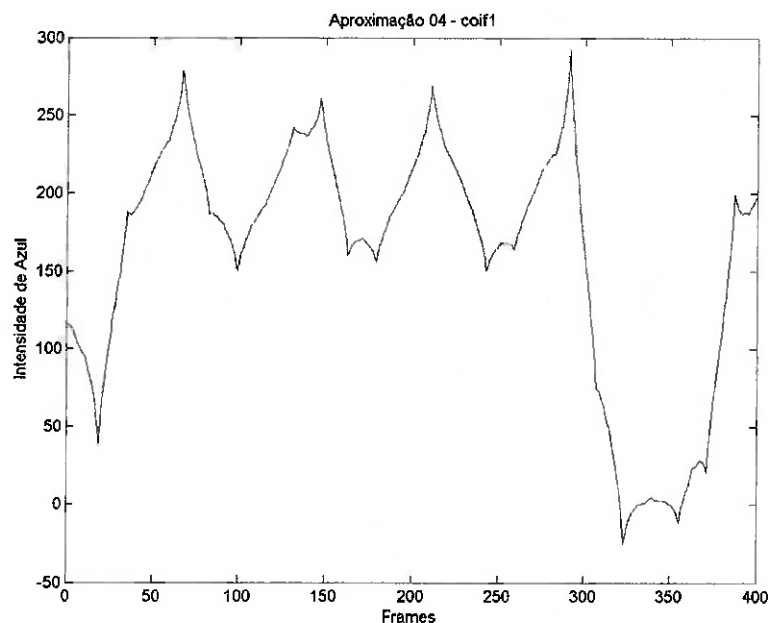




Fig 4.18 – Filtrado por Coiflet1



Nesta seção é importante reparar que: uma filtragem de mesma ordem com famílias diferentes gera resultados bem diferentes. Deste fato remete a importância de escolher uma boa família Wavelet. Lembrando ainda que muitas famílias diferentes podem resultar em boas aproximações, e que nem sempre é simples optar entre uma ou outra família.

#### 4.5. Passagens da decomposição Wavelet do pixel:

Para obter o sinal filtrado conforme a intenção do trabalho, foram executadas quatro decomposições sucessivas, aqui, serão apresentadas a aproximações e o detalhe de cada uma das passagens, inclusive passagens posteriores à quarta decomposição, apenas para que se evidencie a escolha da quarta decomposição.

Fig. 4.19 – Primeira aproximação e detalhe

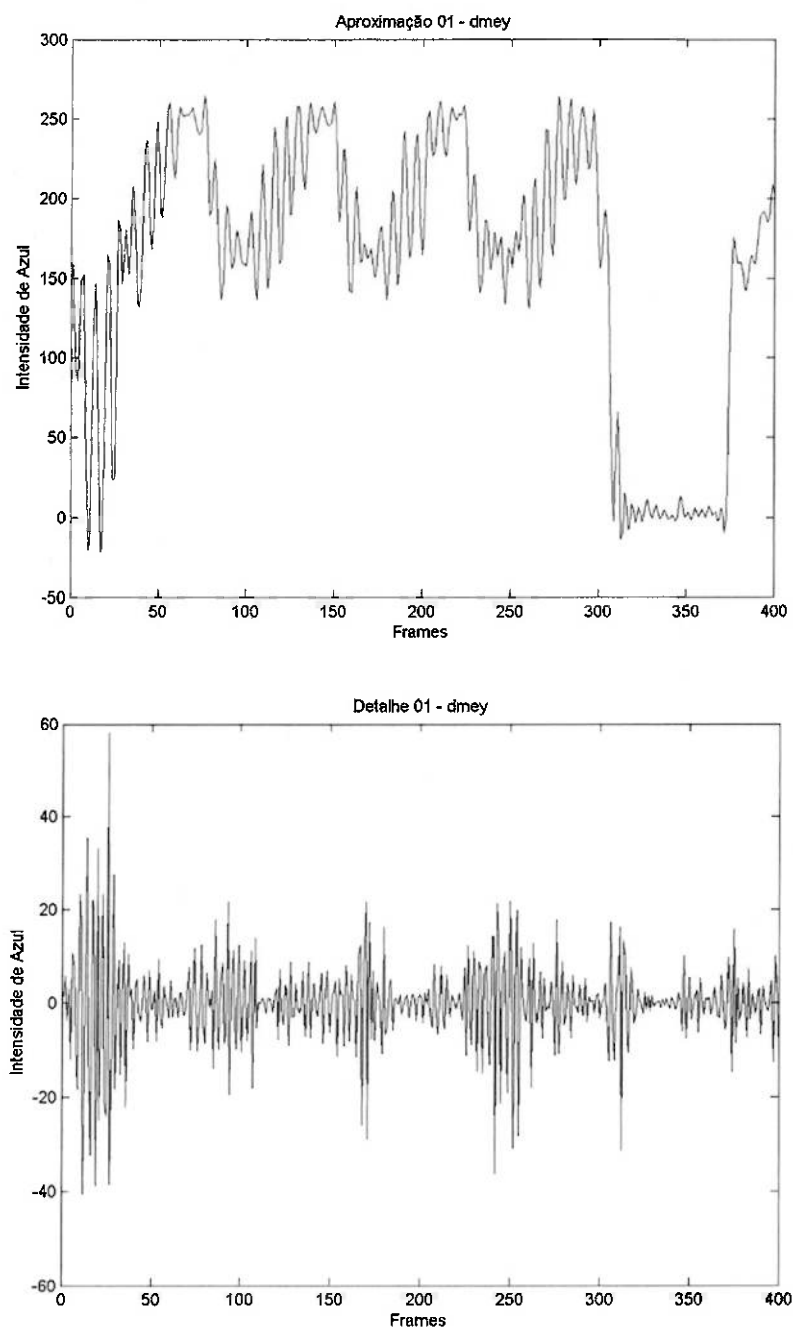


Fig. 4.20 – Segunda aproximação e detalhe

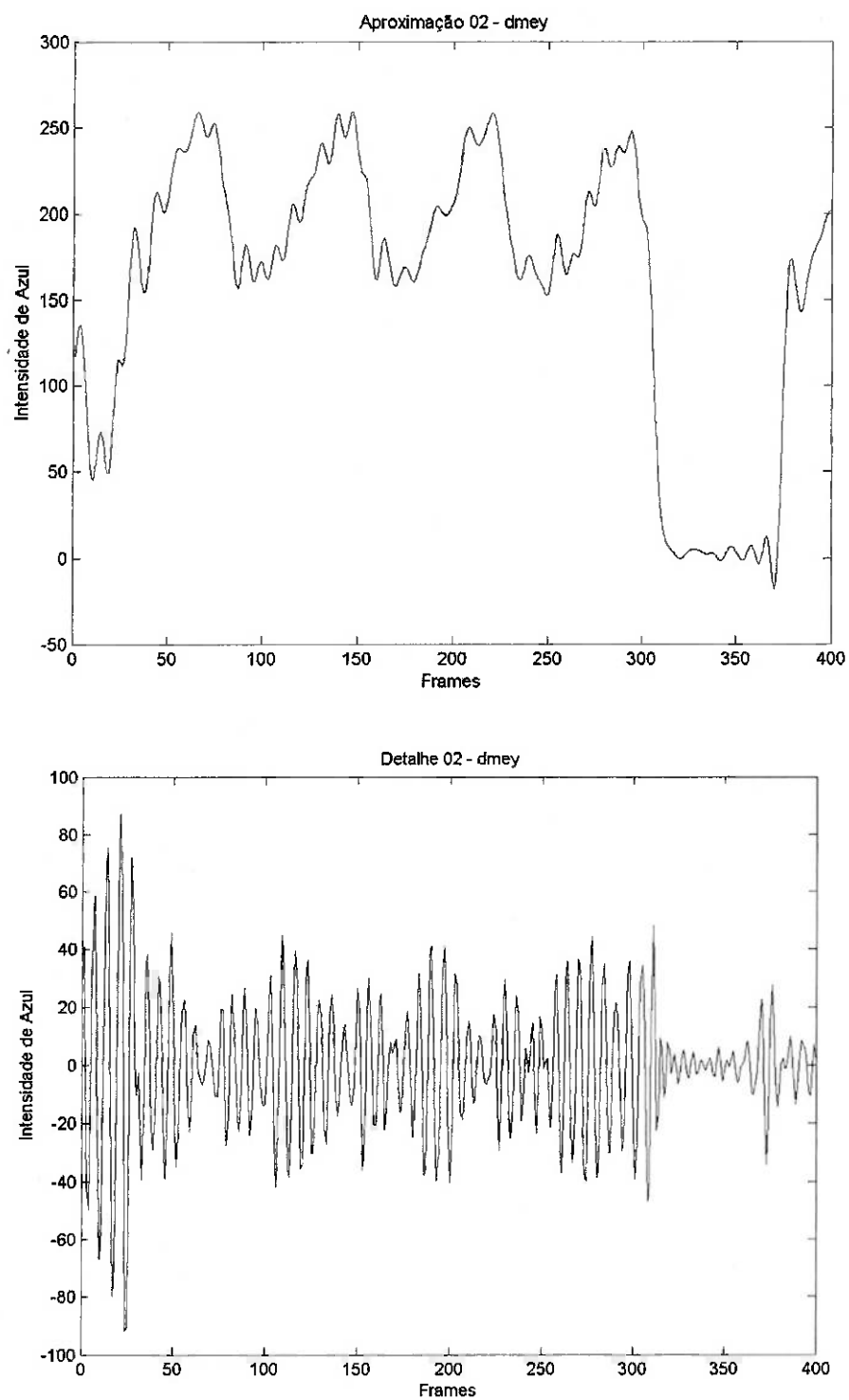


Fig 4.21 – Terceira aproximação e detalhe

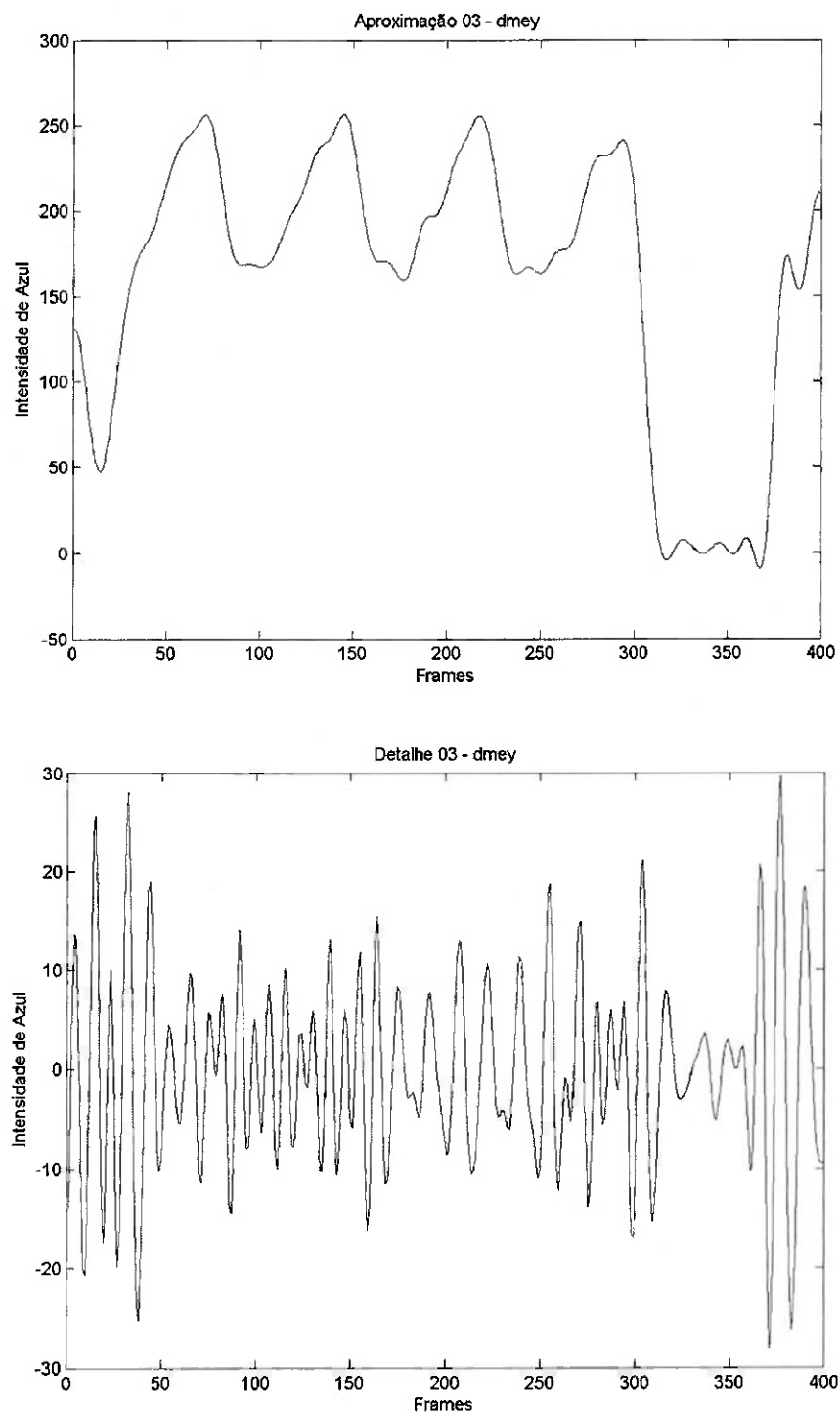


Fig. 4.22 – Quarta aproximação e detalhe

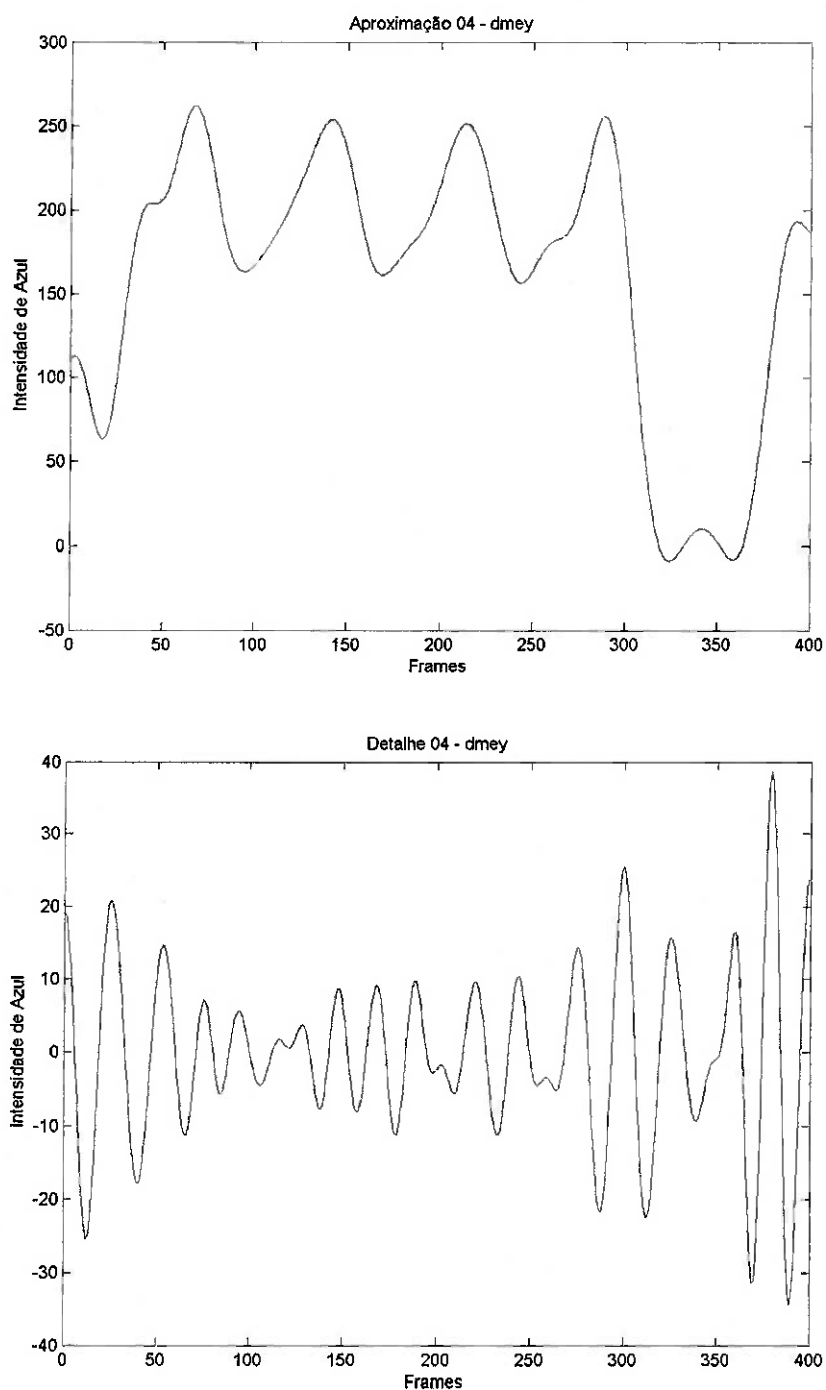


Fig. 4.23 – Quinta aproximação e detalhe

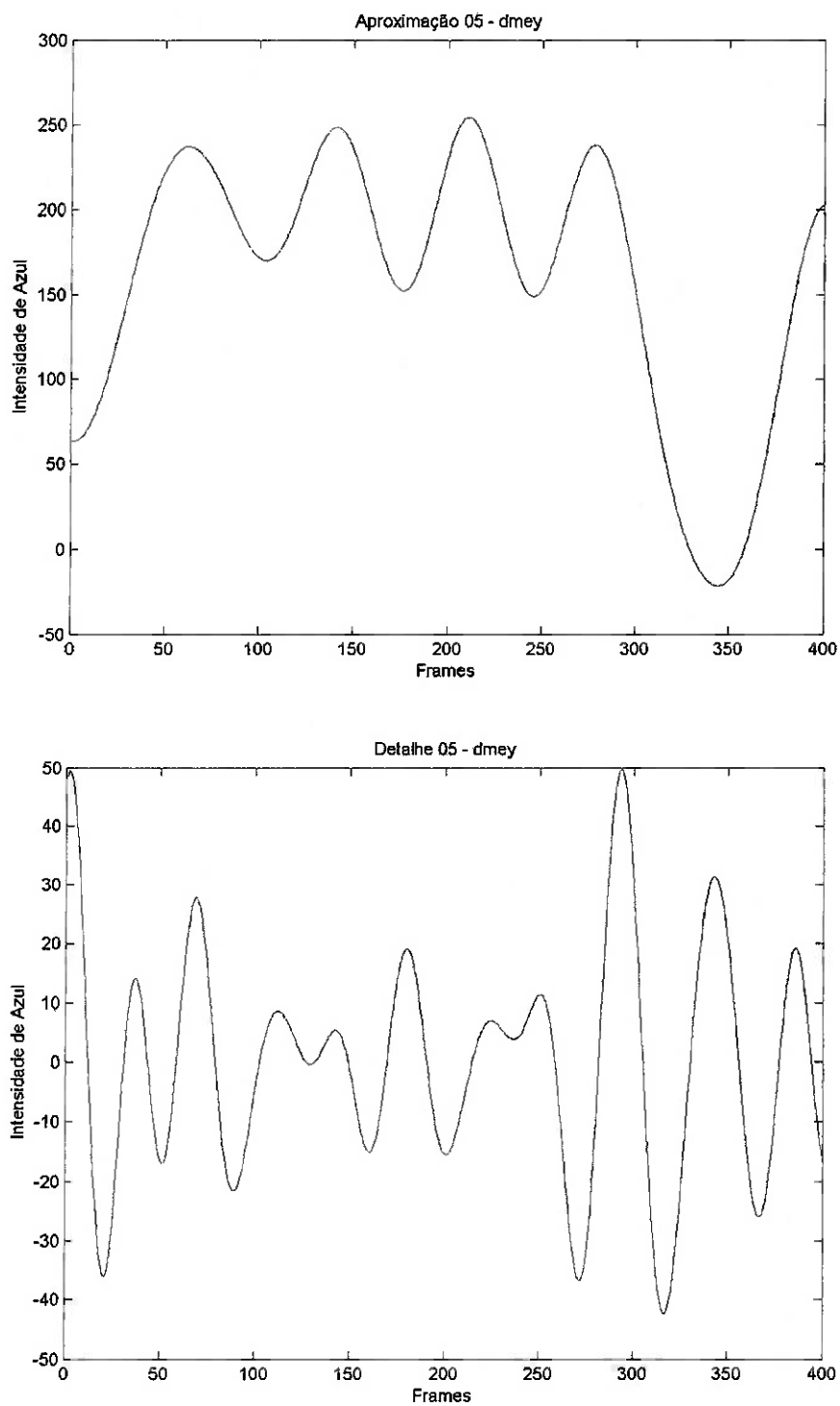


Fig. 4.24 – Sexta aproximação e detalhe

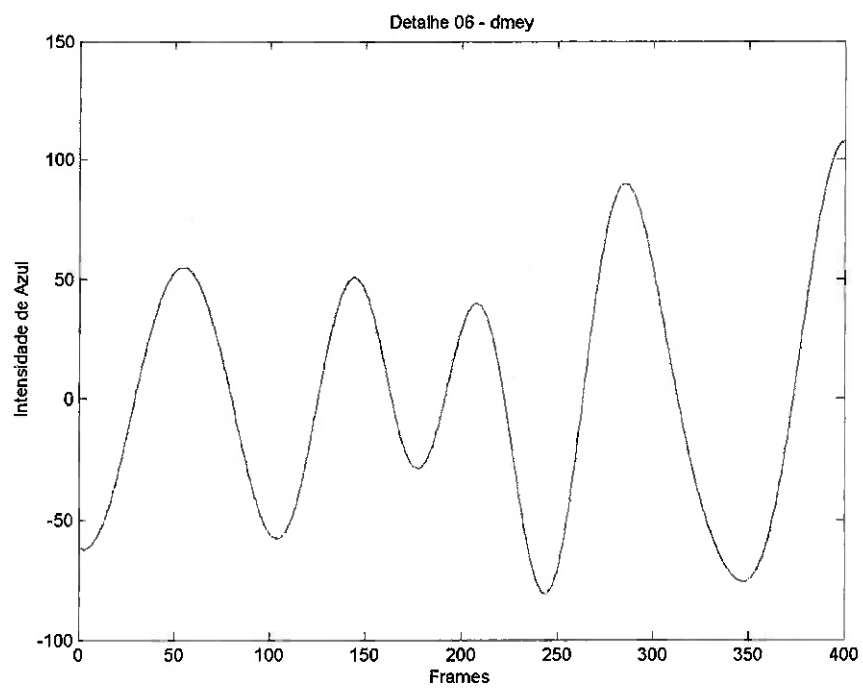
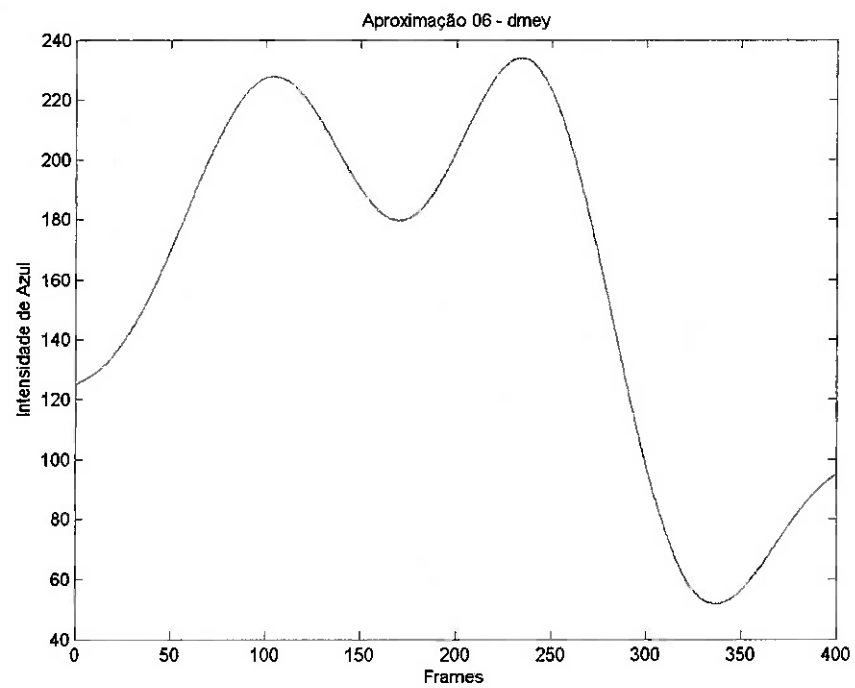


Fig. 4.25 – Sétima aproximação e detalhe

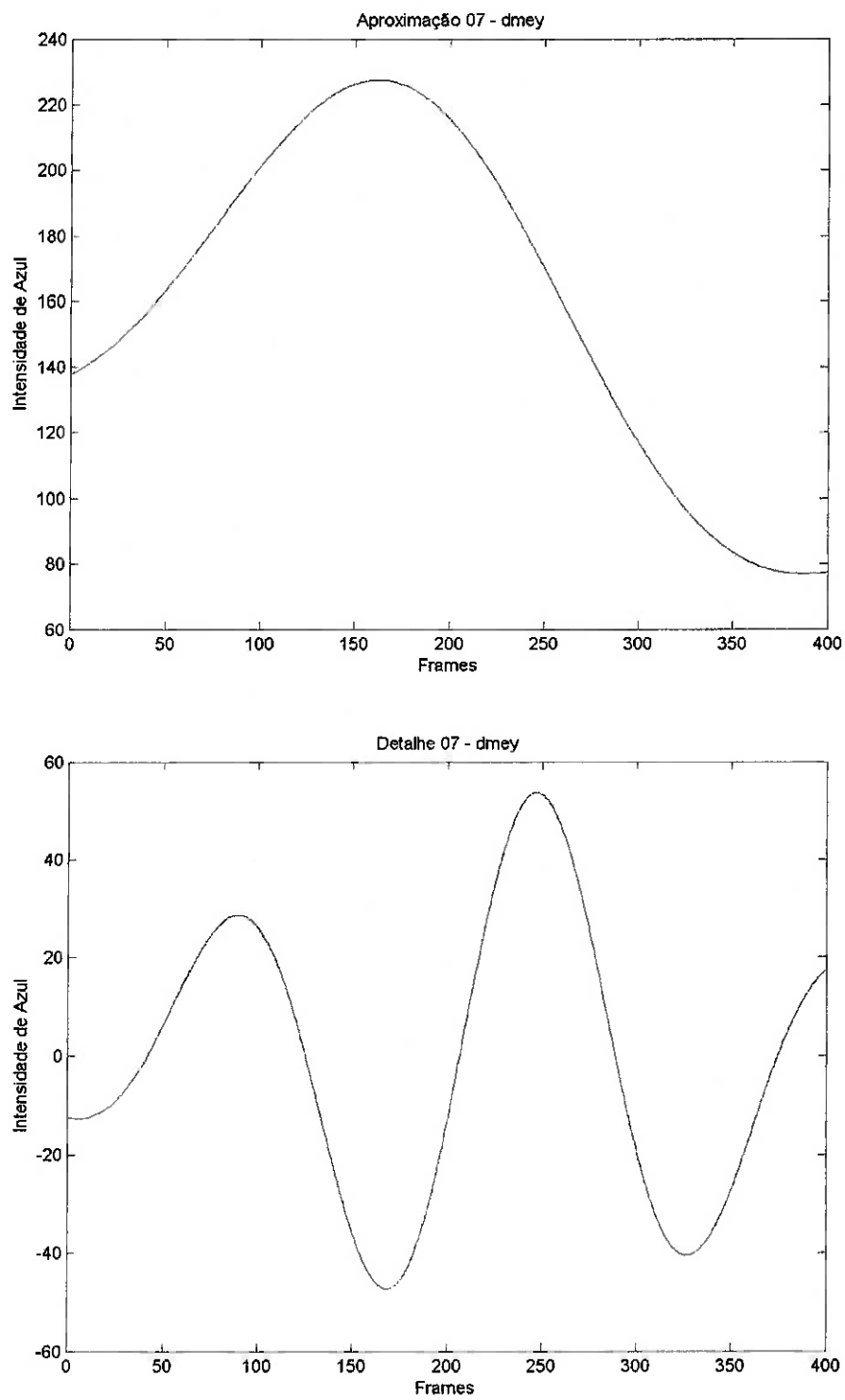
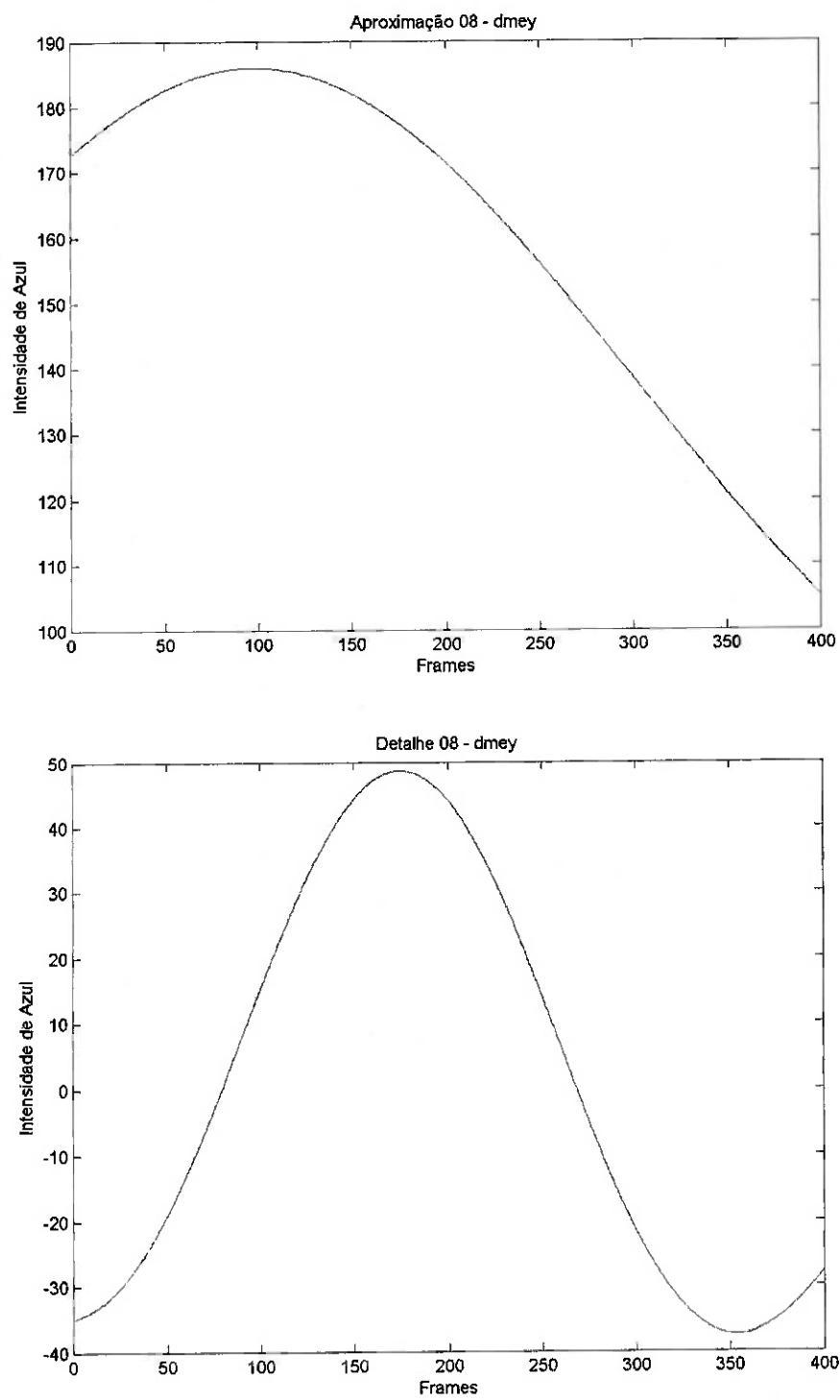




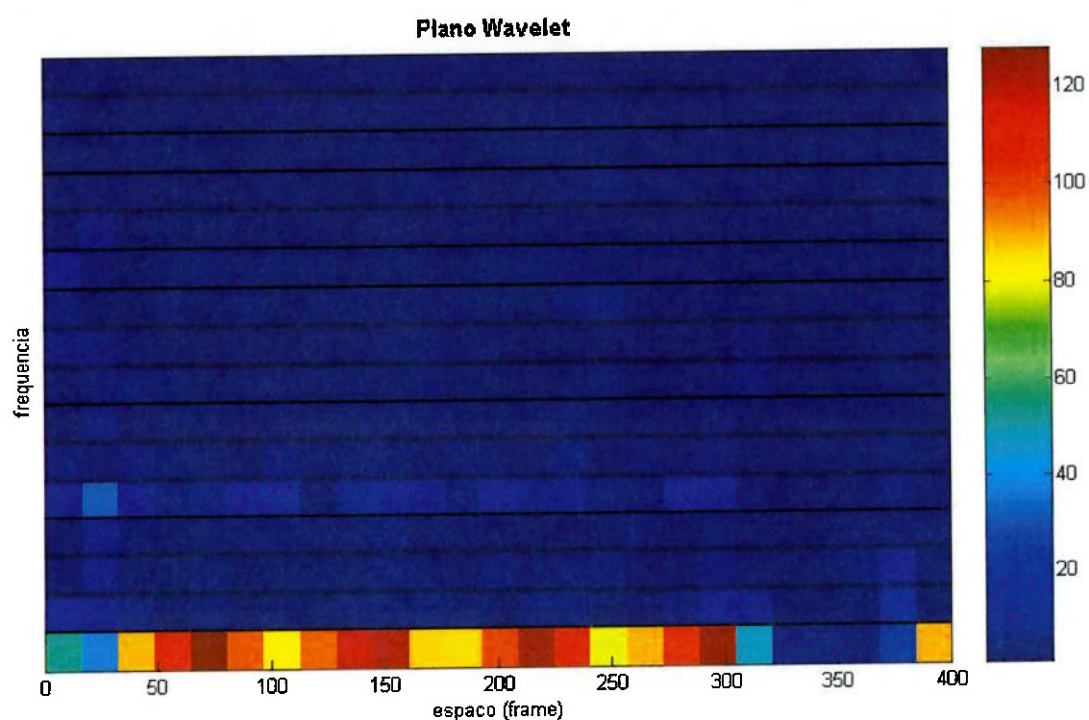
Fig. 4.26 – Oitava aproximação e detalhe



#### 4.6. Plano da filtragem Wavelet:

Uma forma de observar toda a composição da árvore da decomposição Wavelet é plotar no plano do tempo (no caso em frames) pela frequência, que na realidade remete a todas as divisões de meia banda. A intensidade do sinal representada em um código de cores conveniente.

Fig. 4.27 – Plano Wavelet



Perceba que as cores mais quentes se concentram na região inferior, isto porque, como visto na FFT, a faixa de maior energia é a de baixas frequências, a informação adicional obtida nesse tipo de análise está relacionada a distribuição espacial (no caso em unidades de frame) na abscissa, em uma análise convencional em termos do tempo.

#### **4.7. Resultados da filtragem do vídeo com FIR:**

Foram obtidos resultados positivos para um pixel, e então, partiu-se para a rotina de filtragem de cada pixel através da utilização de laços lógicos em ambiente Matlab, a fim de filtrar toda a matriz de imagens no tempo (frames). As simulações dos 392.700 sinais das 3 cores (totalizando mais de 1 milhão de sinais) consumiram ao todo cerca de 10 horas de processamento em uma máquina com 640MB de memória RAM e processador de 1,53GHz.

Os resultados foram satisfatórios, embora a defasagem gerada em cada pixel seja um pouco diferente, o que cria uma pequena desconexão entre os pontos filtrados, por vezes é possível notar resquícios de alta frequência no vídeo filtrado, não por o pixel não estar devidamente filtrado, mas pelo fato de pixels vizinhos estarem em pequena defasagem. O filtro FIR apresenta uma boa vantagem, de permitir a filtragem contínua.

#### **4.8. Resultados da filtragem do vídeo com Wavelet:**

A exemplo do executado para o filtro FIR, após a execução bem sucedida das rotinas com um pixel, iniciou-se o procedimento de análise para o vídeo completo. Ainda em ambiente Matlab, no caso da filtragem via Wavelet, as simulações dos 392.700 sinais de cada cor, somados consumiram cerca de 1,5 horas de processamento na mesma máquina, com 640MB de memória RAM e processador de 1,53GHz.

A principal característica observada diz respeito à manutenção do espaço por todos os pixels, o que gera um excelente acoplamento entre os pixels. Outro fato interessante do filtro aplicado é a necessidade da filtragem por janelas, a esse fato podem estar ligados alguns pesares nessa filtragem, que não pode ser feita instantaneamente, além de contar com possíveis erros de truncamento. Se bem utilizada produz excelentes resultados.

## CONCLUSÕES

Primeiramente, cabe ressaltar que aqui foram apresentados alguns (poucos) conceitos do tratamento matemático de sinal via transformada Wavelet, esta tecnologia encontra-se atualmente em estágio de desenvolvimento, e ainda apresenta um infindável leque de possibilidades.

Quanto às filtragens do sinal e do vídeo, ambas mostraram-se bem sucedida e cada uma apresentou alguns pontos favoráveis e outros não tão favoráveis.

Foi comprovado o atraso de fase imposto pelo filtro FIR e a condição de infidelidade ao sinal nos instantes iniciais, dado que os coeficientes do filtro ainda estão sendo determinados, o que não chega a ser um problema, mas vale ser ressaltado.

O atraso de fase gerou um efeito de pulsação relativo entre os pixels, o que pode induzir a algum erro.

O filtro FIR mostra-se consistente para diversas possibilidades, porem, suponhamos que a faixa de frequências fugisse ao controle (uma respiração ofegante por exemplo), pode-se imaginar que a filtragem não ocorreria devidamente. É evidente que essa abordagem diz respeito apenas ao caso de sinais biológicos, e não coloca em questão o funcionamento do mesmo.

Para a decomposição Wavelet, os resultados se mostram muito fiéis e a questão da manutenção temporal é talvez, o ponto alto da mesma. Um problema diz respeito à escolha da família, pois, além da necessidade de estudar cada uma delas, é altamente recomendável a execução de testes, a fim de comprovar uma boa escolha.

O ponto negativo da filtragem Wavelet, sem duvida diz respeito à necessidade da análise por janelas, o que pode ser corrigido através da imposição de atrasos na execução dos resultados, eliminando nesse caso toda a vantagem de processamento obtida pelo método. É necessário observar também que no caso de alterações da frequência do sinal, a filtragem pode não se mostrar adequada.

A filtragem de um pixel mostrou-se tão efetiva quanto a filtragem do sinal do eletrodo, embora existisse um agravante para esse aspecto, pelo fato de termos uma baixa definição do sinal (8 bits).

A filtragem do vídeo foi uma excelente forma de analisar as limitações do sistema de pós-filtragem por meio do filtro FIR para cada pixel individualmente, que embora tenha se mostrado eficiente, exigiu bastante dos recursos computacionais, através dos meios pelos quais foi executado. E quanto ao Wavelet, embora existam ganhos incomparáveis de execução, ainda desempenhou processamento lento. É claro que deve ser levado em consideração que o processamento foi feito por um software pesado, com interface gráfica e tantos outros agravantes em termos computacionais.

Em termos computacionais, o filtro FIR consumiu cerca de 7 vezes mais recursos do que a análise Wavelet. Qualitativamente, a transformada por ondulas apresenta melhor separação entre imagens de respiração e perfusão, utilizando como critério o que se espera destes sinais do ponto de vista fisiológico.

Conforme era de se esperar, na execução FIR, o conjunto de pixels pós-filtragem ainda apresentou alguma característica da frequência mais alta, isso porque cada pixel foi filtrado localmente, tendo sido deixado de lado o efeito global do vídeo, onde parte do efeito pulsátil permanece aparente. Esse efeito mostra-se bastante sutil, mas não pode ser relevado.

Evidentemente, os resultados poderiam ser melhores caso a filtragem ocorresse antes da estimação dos parâmetros, onde os sinais são em menor numero do que as centenas de milhares de pixels de um vídeo. O efeito de pulsação lateral seria eliminado na estimação dos parâmetros.

Uma outra possibilidade, para eliminar o efeito global no filtro FIR, seria a filtragem da matriz de pixels no sentido de cada frame (linhas horizontais e verticais) porem ainda assim, existem os pixels com vizinhança diagonal, e isso demandaria alto custo computacional.

O tratamento, no entanto, mostrou-se eficiente na eliminação das frequências indesejadas, o que demonstra uma rotina robusta.

À análise Wavelet deve ser destacado o fato de não deixar os chamados efeitos globais, o que demonstra excelente acoplamento entre os pixels, isso ocorre principalmente pela preservação da informação do tempo na filtragem.

Uma comparação entre os dois procedimentos faz com que sejam avaliados cada um dos problemas supracitados e a viabilidade de aceitá-los, ou resolvê-los por outros meios. Para esse estudo, os resultados Wavelet são melhores, engrandeceria o trabalho a filtragem continuada do sinal, a fim de analisar o comportamento da Wavelet.

O tempo exigido na execução dessas filtrações poderia ser bastante diminuído se as rotinas fossem executadas por exemplo em linguagem C, através de um programa compilado, porém em termos de rastreabilidade de problemas a experiência sairia perdendo, visto que a utilização de ferramentas como as do Matlab auxiliam notoriamente a execução do estudo.

O resgate bibliográfico através do tema possibilitou um bom entendimento a respeito do assunto e dos fenômenos que o cercam.

A proximidade do trabalho com a área médica torna-o ainda mais interessante, por vislumbrar a possibilidade de agregar algum tipo de valor tecnológico ao desenvolvimento da Bioengenharia, por meio do tratamento técnico de sinais adquiridos a partir de funções vitais.

A utilização da transformada por ondulas para separar a perfusão da respiração em imagens obtidas por tomografia por impedância elétrica é tecnicamente viável e, nestes resultados preliminares, apresenta vantagens qualitativas nas imagens filtradas resultantes, os sinais filtrados aproximam-se dos fisiológicos.

### **BIBLIOGRAFIA RECOMENDADA:**

- [1] Lu, S., Iyer, K., Hu, J. Functional Characterization of Surface Roughness Generated by Plateau Honing - Process Using Wavelet Analysis, 2004, Detroit, SAE World Congress
- [2] Murray, J. E., Vanryper, W.. Encyclopedia of Graphics File Formats, 1994, O'Reilly & Associates, Inc.
- [3] Nelson, Mark. The Data Compression Book, 1991, M&T Publishing
- [4] Ortolan, R. L.. Estudo e Avaliação de Técnicas de Processamento do Sinal Mielétrico para o Controle de Sistemas de Reabilitação, 2002, São Carlos, EESCUSP
- [5] Proakis, G. J.; Manolakis, G. D.. Digital Signal Processing – Principles, Algorithms, and Applications, 1988, 2 ed., New York, Macmillan
- [6] Runza, F. B.. Análise de Sinais Biológicos Utilizando Wavelets, 2001, São Paulo, EPUSP
- [7] Scholl, D., Yang, B.. Wavelet-Based Visualization, Separation, and Synthesis Tools for Sound Quality of Impulsive Noises, 2003, Traverse City – Michigan, Noise & Vibration Conference and Exhibition - SAE TECHNICAL PAPER SERIES
- [8] Trigo, F. C.. Filtro Estendido de Kalman Aplicado à Tomografia por Impedância Elétrica, 2001, São Paulo, EPUSP

## ANEXO I – Código Fonte Scilab – Rotina FIR

```

clear;
v=fscanfMat('C:\...');
v=v*1000; // efeito de escala
i=0:455;
t=i*42/456;
plot2d(t,v);
T=6*7; // tempo total
l=max(size(v));
fs=l/T;
frq=i*fs/456;
// taxa de amostragem - características do sinal
M=21; ,
// tamanho do filtro fir tem que ser numero impar
passa=0;
//0 se passa baixa, 1 se passa alta
fc=0.5; //frequencia de corte
for i=1:((M+1)/2);
wc(i)=fs*(i-1)/M;
end
//wc= fc*2*pi/fs;
// define o vetor de frequências
for i=1:((M+1)/2);
w(i)=2*pi*(i-1)/M;
//w(i)=2*pi*(i-1)/M;
// metade do espectro
end

controle=wc/fc;
controle=controle-1;
absoluto=abs(controle)
minimo=min(absoluto)

if passa==0 then
for j=1:((M+1)/2)
if controle(j)<minimo
H(j)=1;
end
if abs(controle(j))==minimo
H(j) = 0.4;
end
if controle(j)>minimo
H(j) = 0;
end
end
end

if passa==1 then
for j=1:((M+1)/2)
if controle(j)<minimo
H(j)=0;
end
if abs(controle(j))==minimo

```



```

H(j) = 0.4;
end
if controle(j)>minimo
H(j) = 1;
end
end
end

// define a matriz de coeficientes para realizar a IFFT com restricoes
for n=1:10;
for k=1:11;
a(k,n)=2*cos(w(k)*(M-1)/2-(n-1));
end;
end;
for k=1:11;
a(k,11)= 1;
end;
// determina os coeficientes do filtro FIR, h=fft(H,1)
h=inv(a)*H;
// condicao de simetria nos coeficientes
for i=1:11;
ht(i)=h(i);
end;
for i=1:10;
ht(21-(i-1))=h(i);
end;
// calcula o sinal de voltagem filtrado
vf(1)=v(1)*h(1);
vf(2)=v(2)*h(1)+v(1)*h(2);
vf(3)=v(3)*h(1)+v(2)*h(2)+v(1)*h(3);
for i=1:21
aux=0.0;
for j=1:i
aux=aux+v(i-(j-1))*ht(j);
end;
vf(i)=aux;
end;
for i=22:456;
aux=0.0;
for j=1:21;
aux=aux+v(i-(j-1))*ht(j);
end;
vf(i)=aux;
end;
xset("window",1)
plot2d(t,vf);

```

**ANEXO II – Código Fonte Matlab – Rotina para separar as matrizes RGB**

```
clear
for i=1:400
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
if (i<10)
n=['0',n];
end
file=['arquivo',n,'.bmp'];
IMG=imread(file);
R=IMG(:,:,1);
G=IMG(:,:,2);
B=IMG(:,:,3);
file=['arquivo',n];
save(file,'R','G','B');
end
```

### ANEXO III – Código Fonte Matlab – Redução em quadrantes da área filtrada

```

clear
lim=1062;
inf=45;
sup=385;
ld=450;
le=65;
for i=1:lim
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
if (i<10)
n=['0',n];
end
file=['salvando',n];
load(file);
RED(:,:,i)=R(inf:sup,le:ld);
end
file='RED';
save(file,'RED');
clear RED;
clear R;
for i=1:lim
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
if (i<10)
n=['0',n];
end
end
file=['salvando',n];
load(file);
GREEN(:,:,i)=G(inf:sup,le:ld);
end
file='GREEN';
save(file,'GREEN');
clear GREEN;
clear G;
for i=1:lim
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
end

```

```
if (i<10)
n=['0',n];
end
file=['salvando',n];
load(file);
BLUE(:,:,i)=B(inf:sup,le:ld);
end
file='BLUE';
save(file,'BLUE');
clear BLUE;
clear B;
```

#### ANEXO IV – Código Fonte Matlab – Filtro FIR para todos os pixels

```

clear
clear all;
for color=1:3
    color
    if color==1
        load RED
        cor=RED(:,:,1:400);
        clear RED
    end
    if color==2
        clear cor
        load GREEN
        cor=GREEN(:,:,1:400);
        clear GREEN
    end
    if color==3
        clear cor
        load BLUE
        cor=BLUE(:,:,1:400);
        clear BLUE
    end
    for lin=1:341
        for col=1:386
            for i=1:400
                v(i)=cor(lin,col,i);
            end
            i=1:400;
            t=i*42/400;
            %v=sin(2*pi*i/30)+0.5*sin(2*pi*i/5);
            %plot(t,v);
            T=6*7; % tempo total
            l=max(size(v));
            fs=1/T;
            % taxa de amostragem - características do sinal
            M=21;
            % tamanho do filtro fir tem que ser numero impar
            passa=0;
            %0 se passa baixa, 1 se passa alta
            fc=1; %frequencia de corte
            for i=1:(M+1)/2
                wc(i)=fs*(i-1)/M;
            end
            %wc= fc*2*pi/fs;
            % define o vetor de frequencias
            for i=1:(M+1)/2
                w(i)=2*pi*(i-1)/M;
                %w(i)=2*pi*(i-1)/M;
            end
            % metade do espectro
            controle=wc/fc;
            controle=controle-1;
        end
    end
end

```

```

absoluto=abs(controle);
minimo=min(absoluto);

if passa==0
for j=1:((M+1)/2)
if controle(j)<minimo
H(j)=1;
end
if abs(controle(j))==minimo
H(j) = 0.4;
end
if controle(j)>minimo
H(j) = 0;
end
end
end

if passa==1
for j=1:((M+1)/2)
if controle(j)<minimo
H(j)=0;
end
if abs(controle(j))==minimo
H(j) = 0.4;
end
if controle(j)>minimo
H(j) = 1;
end
end
end

% define a matriz de coeficientes para realizar a IFFT com restricoes
% a matriz a(,) esta explicada na secao 8.1.2 do Proakis
for n=1:10
for k=1:11
a(k,n)=2*cos(w(k)*((M-1)/2-(n-1)));
end
end
for k=1:11
a(k,11)= 1;
end
% determina os coeficientes do filtro FIR, h=fft(H,1)
h=inv(a)*H';
%h=inv(a)*H;
% condicao de simetria nos coeficientes
for i=1:11
ht(i)=h(i);
end;
for i=1:10
ht(21-(i-1))=h(i);
end
% calcula o sinal de voltagem filtrado
vf(1)=v(1)*h(1);
vf(2)=v(2)*h(1)+v(1)*h(2);

```

```
vf(3)=v(3)*h(1)+v(2)*h(2)+v(1)*h(3);
for i=1:21
    aux=0.0;
    for j=1:i
        aux=aux+v(i-(j-1))*ht(j);
    end;
    vf(i)=aux;
end;
for i=22:400;
    aux=0.0;
    for j=1:21;
        aux=aux+v(i-(j-1))*ht(j);
    end;
    vf(i)=aux;
end;
%plot(t,vf);
cor(lin,col,:)=vf;
end
end
file='COLOR';
RGB=num2str(color);
file=[file,RGB];
save(file,'cor');
end
```

**ANEXO V – Código Fonte Matlab – Recomposição das imagens**

```
clear
load LOWFIR
for i=1:400
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
if (i<10)
n=['0',n];
end
file=['lowfir',n,'.bmp'];
IM(:,:,1)=FLRED(:,:,i);
IM(:,:,2)=FLGREEN(:,:,i);
IM(:,:,3)=FLBLUE(:,:,i);
imwrite(IM,file);
clear IM;
end
```



**ANEXO VI– Código Fonte Matlab – Recomposição do vídeo em AVI**

```
clear
AVIOBJ=avifile('movie.avi','fps',24)
for i=1:400
n=num2str(i);
if (i<1000)
n=['0',n];
end
if (i<100)
n=['0',n];
end
if (i<10)
n=['0',n];
end
file=['lowfir',n,'.bmp'];
IM=imread(file);
AVIOBJ=addframe(AVIOBJ,IM);
clear IM;
end
AVIOBJ=close(AVIOBJ);
```

**ANEXO VII – Código Fonte Matlab – Filtragem Wavelet para todos os pixels**

```
clear all;
for color=1:3
    color
    if color==1
        load RED
        cor=RED(:,:,1:400);
        clear RED
    end
    if color==2
        clear cor
        load GREEN
        cor=GREEN(:,:,1:400);
        clear GREEN
    end
    if color==3
        clear cor
        load BLUE
        cor=BLUE(:,:,1:400);
        clear BLUE
    end
    for lin=1:341
        for col=1:386
            for i=1:400
                v(i)=cor(lin,col,i);
            end
            s=double(v);
            [c,l]=wavedec(s,4,'dmey');
            a4 = wrcoef('a',c,l,'dmey',4);
            cor(lin,col,:)=a4;
        end
    end
    file='COLOR';
    RGB=num2str(color);
    file=[file,RGB];
    save(file,'cor');
end
```