

Transformada de **WAVELET**

2016

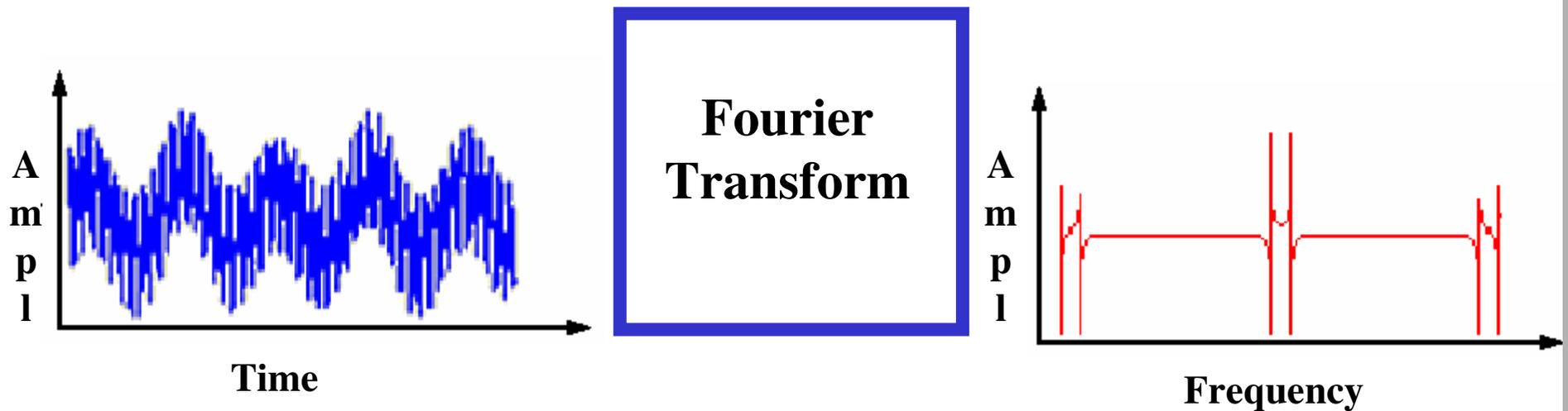
Curso PISB:

Processamento de Imagens e Sinais Biológicos

Cap. 5 : K. Najarian and R. Splinter, **Biomedical Signal and Image**

Processing CRC Press - Taylor & Francis Group, 2006

A Transformada de Fourier (TF) descreve as diferentes frequências contidas em uma imagem, mas **não a localização espacial** destas frequências



Fourier (1807)

Fourier é ótimo para sinais estacionários

- Sinais cujo **conteúdo não mudam no tempo** são chamados de sinais estacionários.
- Em outras palavras, o conteúdo do sinal não muda com o tempo.
- Neste caso, **não é preciso saber “quando” um determinado componentes de frequência existe**, já que todos os componentes de frequência existem todo o tempo!

$$z(t) = \cos(2\pi \cdot 10t) + \cos(2\pi \cdot 25t) + \cos(2\pi \cdot 50t) + \cos(2\pi \cdot 100t)$$

$$-\infty < t < \infty$$

Fourier não distingue **sinais não estacionários**

Sinais cuja frequência muda - > sinais não estacionários.

Por exemplo tem a mesma TF os conjuntos de sinais $x(t)$ e $y(t)$:

$$x(t) = \cos(2\pi * 10t), \quad \text{para} \quad 0 < t < 200 \text{ mili segundos}$$

$$x(t) = \cos(2\pi * 25t), \quad \text{para} \quad 200 < t < 400 \text{ mili segundos}$$

$$x(t) = \cos(2\pi * 50t), \quad \text{para} \quad 400 < t < 800 \text{ mili segundos}$$

$$x(t) = \cos(2\pi * 100t), \quad \text{para} \quad 800 < t < 1000 \text{ mili segundos}$$

$$y(t) = \cos(2\pi * 25t), \quad \text{para} \quad 0 < t < 200 \text{ mili segundos}$$

$$y(t) = \cos(2\pi * 100t), \quad \text{para} \quad 200 < t < 400 \text{ mili segundos}$$

$$y(t) = \cos(2\pi * 10t), \quad \text{para} \quad 400 < t < 800 \text{ mili segundos}$$

$$y(t) = \cos(2\pi * 50t), \quad \text{para} \quad 800 < t < 1000 \text{ mili segundos}$$

A TF dos 3 sinais $x(t)$, $y(t)$ e $z(t)$, tem quatro picos, correspondendo a quatro frequências: 10, 25, 50 e 100 Hz.

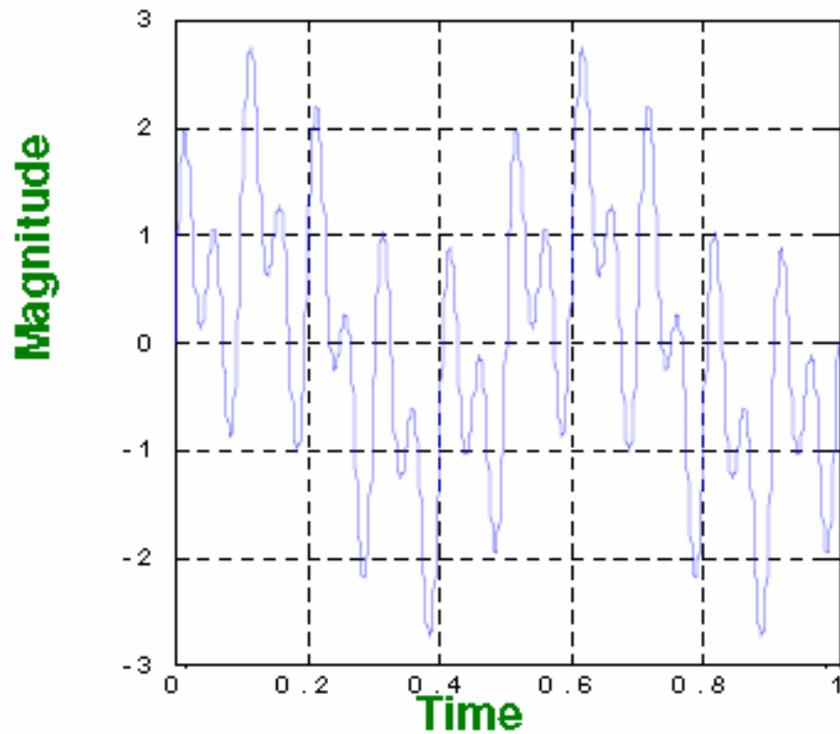
- FT não é uma técnica adequada para o sinal não-estacionário.
- FT pode ser usado para sinais não-estacionários, se estamos interessados apenas em **quais componentes de frequência existem no sinal**, mas não em que tempo estes ocorrem.
- No entanto, se for necessário saber, **em que tempo um componente (que momento), a transformada de Fourier não é a mais adequada para se usar.**

Outro exemplo:

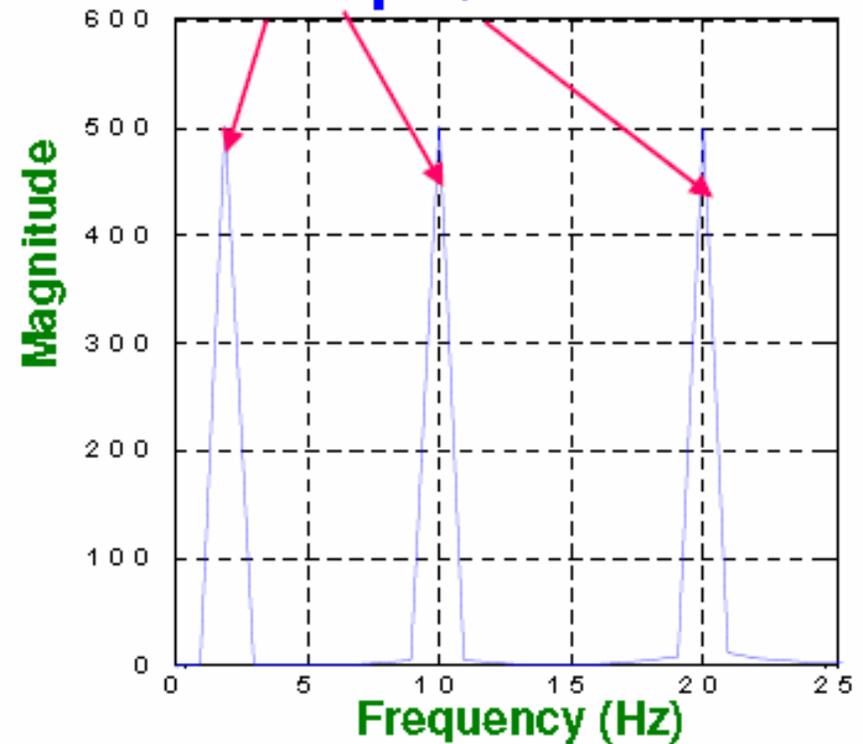
- estacionário

2 Hz + 10 Hz + 20Hz

Estacionário

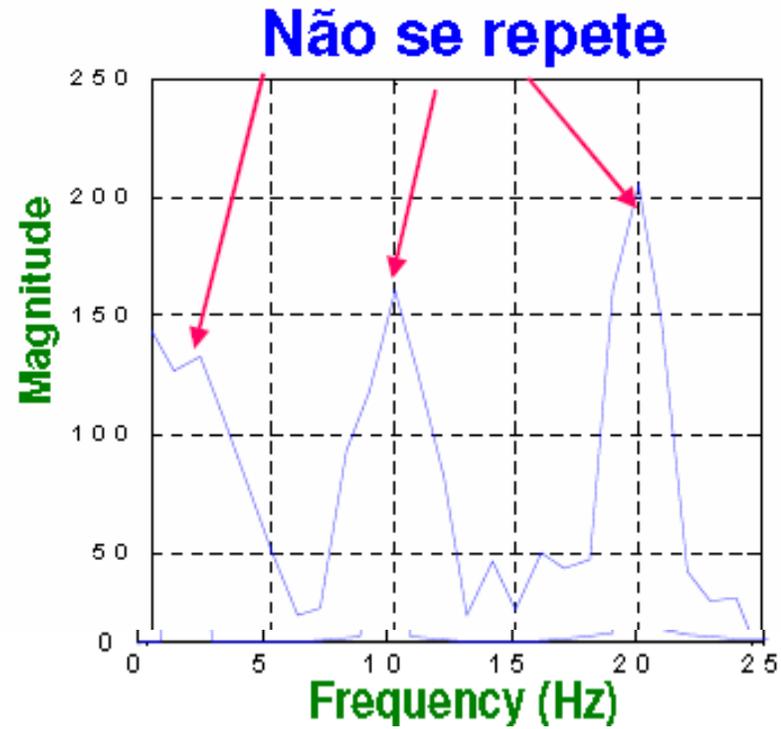
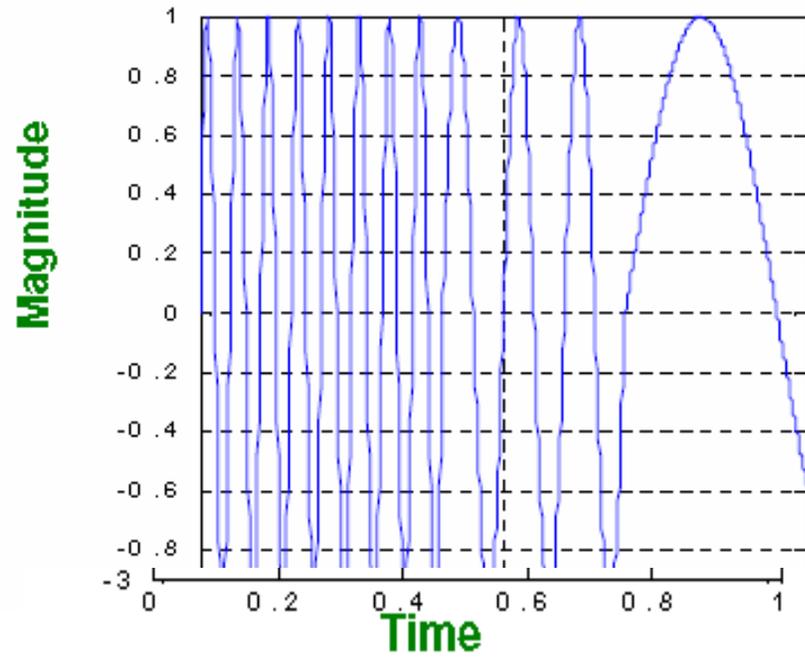


Repete-se...

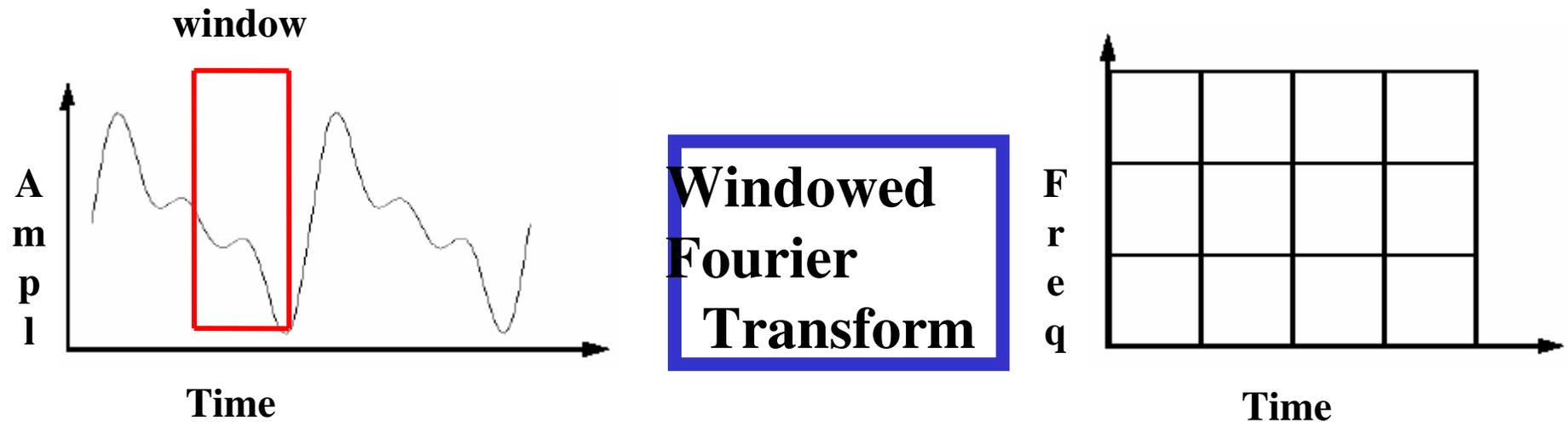


0.0-0.4: 2 Hz +
0.4-0.7: 10 Hz +
0.7-1.0: 20Hz

Não-
Estacionário



Para contornar isso algumas propostas foram surgindo:
técnicas que aplicam particionamentos e multiresolução visando incluir informações temporais (ou espaciais no caso de imagens) junto com as de frequência, uma destas foi a **Windowed Fourier Transform** ou Short Time Fourier Transform (STFT) .



Gabor (1946)

sugestão de leitura: <http://users.rowan.edu/~polikar/WAVELETS/WTpart3.html>

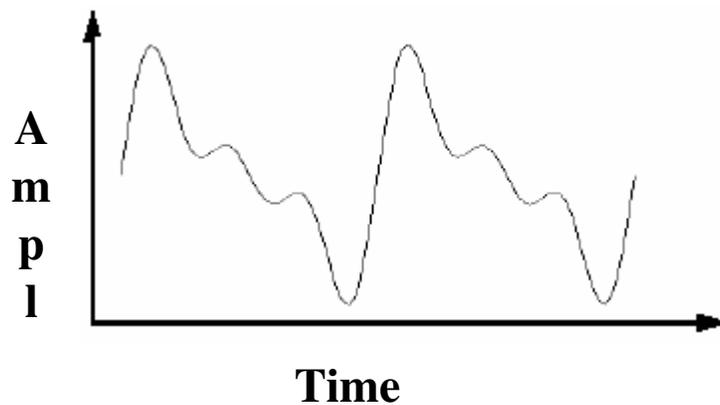
- Transformada de Fourier para Tempo Curto
 - *Short Time Fourier Transform* (STFT)
- Considera uma janela que vai se deslocando pelo sinal e avaliando cada parte dele em separado
- Cada janela considera que a parte do sinal dentro dela é aproximadamente estacionário

Short Time Fourier Transform (STFT)

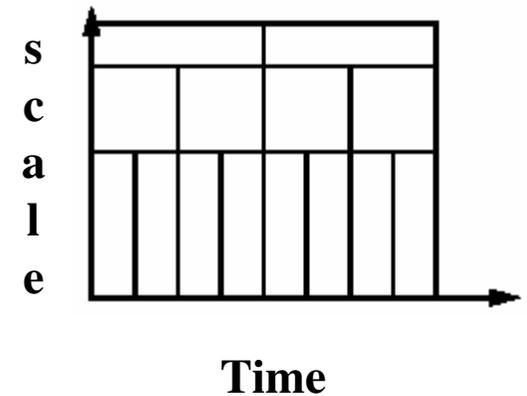
- Problema: Janela de tamanho invariante
 - Além disso, como definir o tamanho da janela?
 - Janela pequena:
 - Pouca informação sobre o sinal
 - Muito processamento
 - Janela grande:
 - Aumenta o erro na consideração de estacionaridade do sinal

Mas a mais interessante foi a idéia inicialmente proposta por Mallat em 1989 , que hoje chamamos:

Wavelet Transform

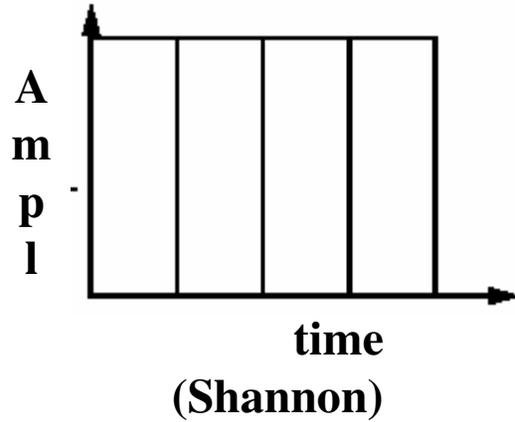


Wavelet Transform

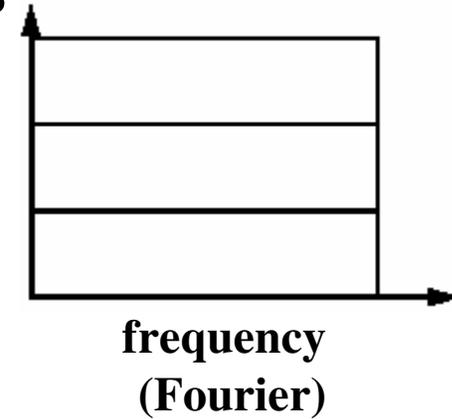


<http://www.cmap.polytechnique.fr/~mallat/papiers/MallatTheory89.pdf>

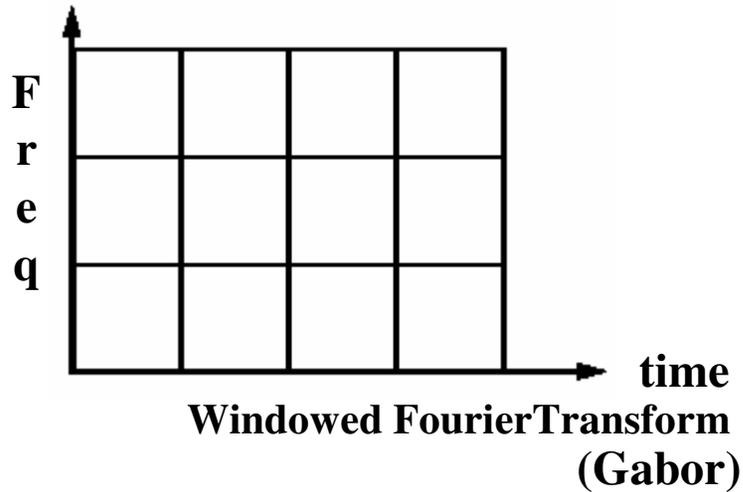
Evolução das Transformadas



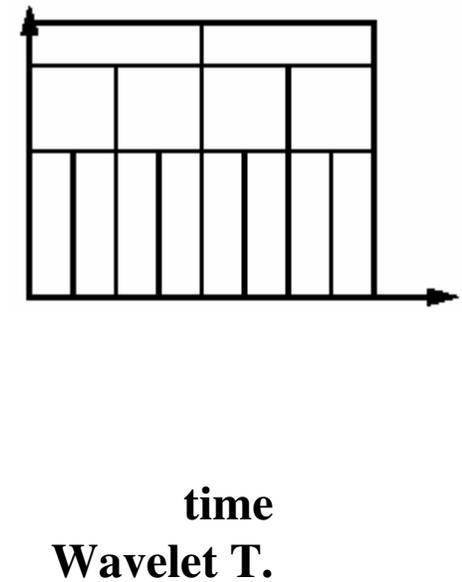
amplitude



Todos os pontos no plano tempo-frequência, que estão dentro de uma caixa são representada por um valor da WT.



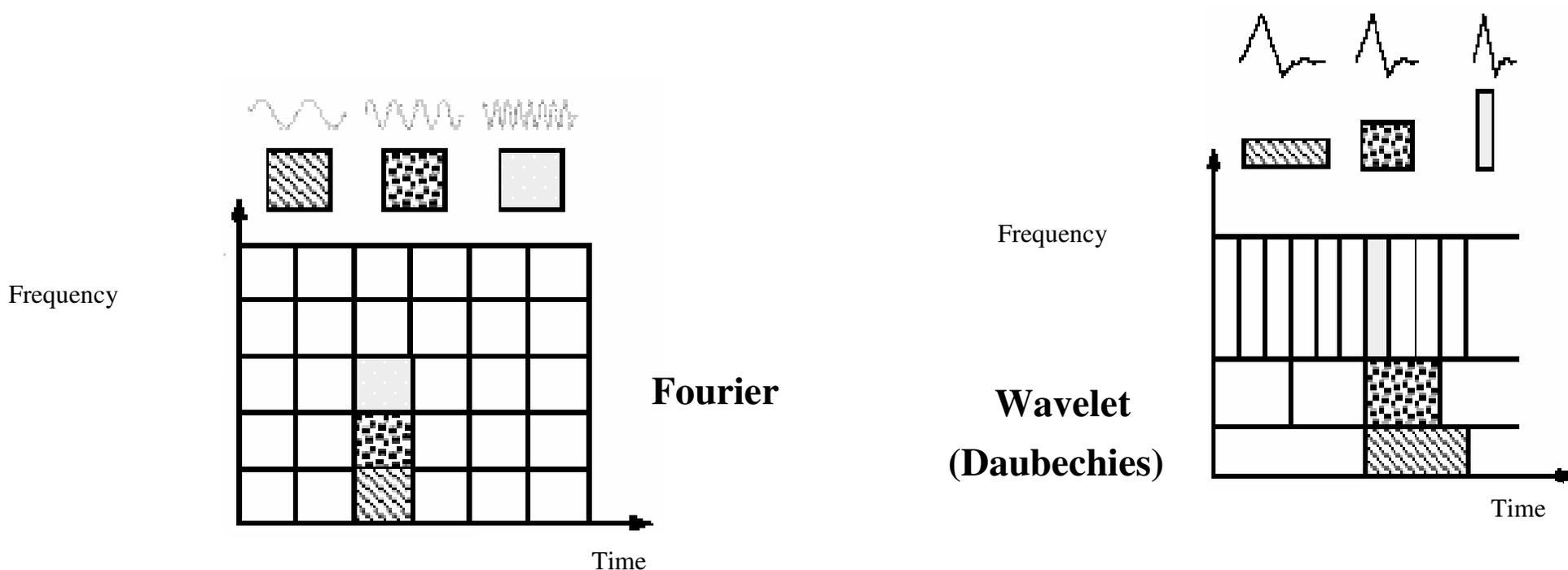
s c a l a



A **resolução** ou detalhamento (da análise) **no domínio da frequência** (Fourier) **diminui enquanto a resolução no tempo aumenta.**

É **impossível** aumentar o detalhamento em um dos domínios sem diminuí-lo no outro (chama-se esta **relação inversa entre os domínios da frequência e do tempo** de)
: *princípio da incerteza.*

Usando as wavelet, é possível escolher a **melhor combinação dos detalhamentos** para um objetivo estabelecido.



Wavelet x Fourier

TW

- Transformada wavelet é capaz de fornecer a **informação do tempo e da frequência em simultâneo**, por conseguinte, dando uma representação de tempo - frequência do sinal ao mesmo tempo
- A WT passa o sinal no **domínio do tempo** por vários filtros passa **alta e passa-baixa**, que filtram as altas e baixa frequência do sinal.
- Este procedimento é **repetido**, cada vez que uma parte do sinal que corresponde a algumas frequências seja removidos do sinal.

Próximo passo na evolução....

- Uma transformada em janelas, mas com janelas de tamanho variável:
 - Transformada Wavelet
 - Intervalos maiores quando queremos informações mais precisas sobre baixas freqüências
 - Intervalos menores quando queremos informações mais precisas sobre altas freqüências

Análise Multi Resolução – MRA)

- Embora o problema de analisar em detalhes no tempo e frequência ao mesmo tempo seja devido a fenômeno físico (o **princípio da incerteza** de Heisenberg) e existir independentemente do fenômeno em análise, é possível analisar qualquer sinal, utilizando uma **abordagem alternativa denominada análise multi resolução** (MRA).
- MRA, como o nome sugere, analisa **o sinal em frequências diferentes, com diferentes resoluções**. Cada componente espectral não é resolvido igualmente como foi o caso no STFT (TF ou TF janelada) .
- MRA é projetada para uma resolução de tempo boa e resolução de frequência pobre em altas frequências e resolução boa em frequência e em tempo ruim em baixas frequências.
- Essa abordagem faz sentido, especialmente quando o sinal tem componentes de **alta frequência por períodos curtos** e **componentes de baixa frequência para durações longas**.
- Felizmente, os sinais e imagens que são encontradas em aplicações práticas são deste tipo.
- Mas os das imagens térmicas seria? Sim, fizemos essa experiência anos atrás!

A transformada de wavelet

- Foi desenvolvida como uma abordagem alternativa para a STFT no sentido de resolver o problema de resolução tempo/frequência.
- A análise wavelet é feita de uma maneira similar à análise STFT, no sentido em que o **sinal é multiplicado com uma função**, (a wavelet), semelhante à função janela no STFT e a transformação é calculada em separado para diferentes segmentos do sinal no domínio de tempo.
- No entanto, existem duas diferenças principais entre a STFT e a WT:
 1. A transformação de Fourier das janelas **não são** calculadas; e
 2. A **largura da janela** é alterada quando a transformação é calculada para cada componente, o que é provavelmente a característica mais significativa da transformada wavelet.

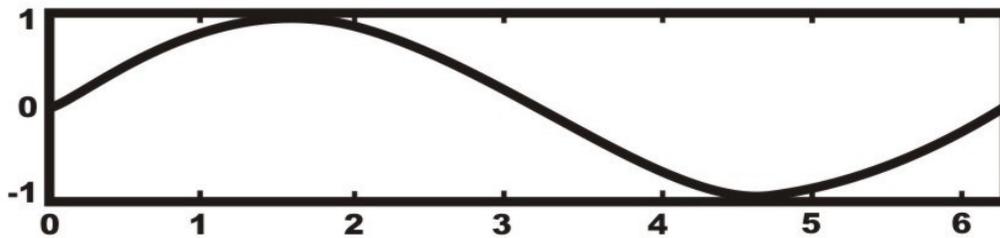
O que é Wavelet (significa pequena onda , ou em português: onduleta ou ondaleta)

(*ondeletes*, em frances) ?

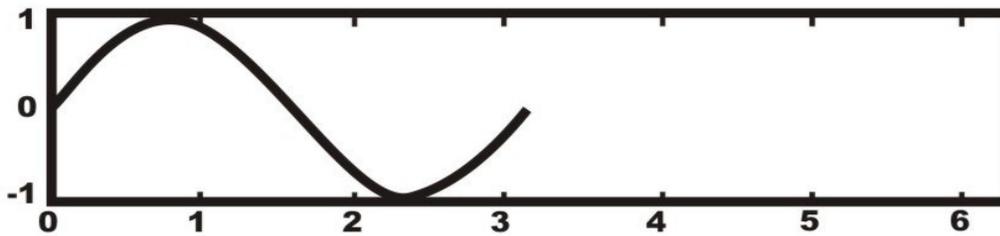
- Wavelets são uma classe de funções usadas **para localizar** uma determinada imagem ou sinal **ao mesmo tempo em posição e escala**.
- Uma família de Wavelets pode ser construído a partir de uma função, chamada “wavelet mãe” ou “base”.
- As "*Wavelets filhas*" são, então, formadas por translação e contração da “wavelet mãe”.

**Se a define a escala e b a translação ,
uma wavelet base ou “wavelet mãe”
pode ser escrita como:**

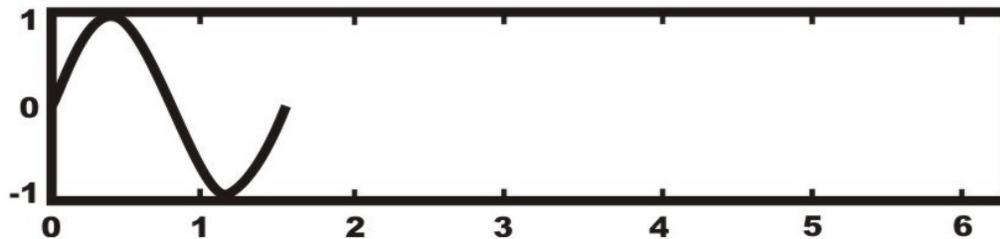
$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right), \quad a \neq 0, \quad b \in \mathfrak{R}$$



$$f(t) = \sin ; a = 1$$



$$f(t) = \sin(2t) ; a = \frac{1}{2}$$



$$f(t) = \sin(4t) ; a = \frac{1}{4}$$

Exemplificando o que seria o fator de escala de uma função

Parâmetro de Escala

O fator escala a representa uma contração ou dilatação no sinal.

Para $a > 1$ a função sofre uma **dilatação**,

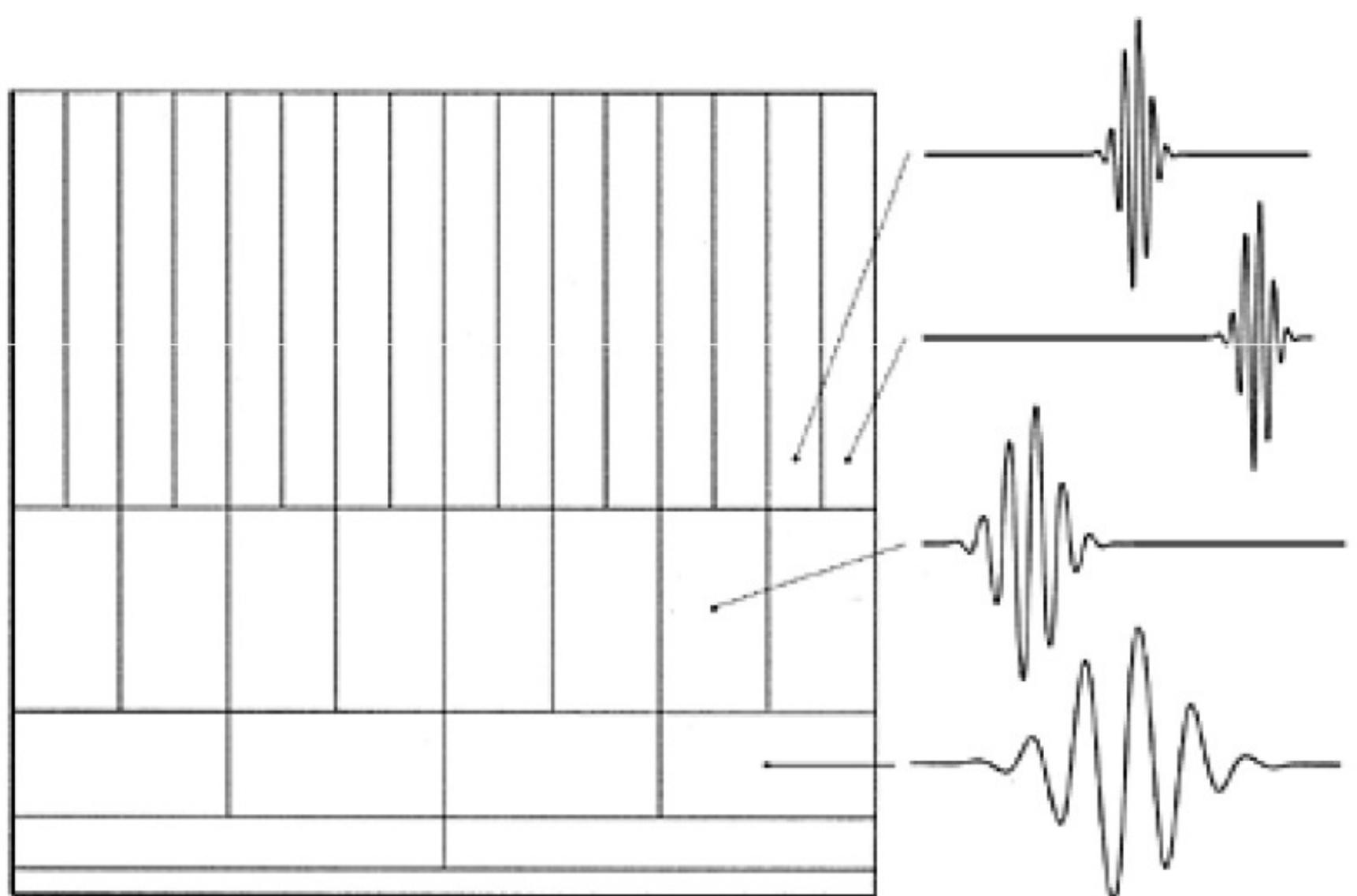
para $a < 1$ obtém uma **contração** do sinal.

As **escalas maiores** correspondem a uma vista não detalhada (**global**) e as escalas **menores** correspondem a uma vista detalhada (**local**).

De modo semelhante, em frequência, as **baixas frequências** correspondem a uma **informação global** (que geralmente se estende por todo o sinal ou imagem), enquanto as **frequências altas** (escalas reduzidas) correspondem a uma informação **detalhada local** (que geralmente dura um período de tempo relativamente curto).

Transformada wavelet

- A decomposição de uma função com o uso de wavelets é conhecida como **transformada wavelet** e pode ser **continua ou discreta**.
- Graças a capacidade de decompor as funções **tanto no domínio da frequência quanto no domínio do tempo** (ou espaço) => as wavelet são *tools* **poderosas** no processamento de sinais e imagens.

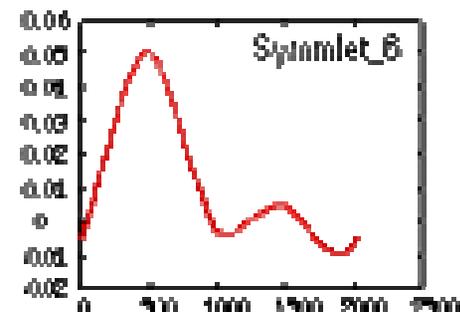
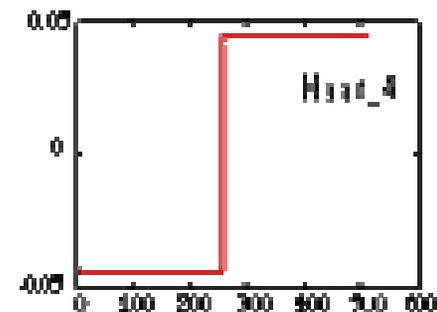
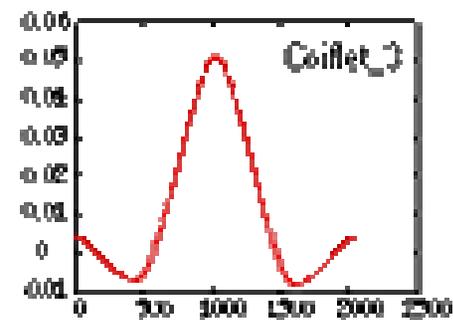
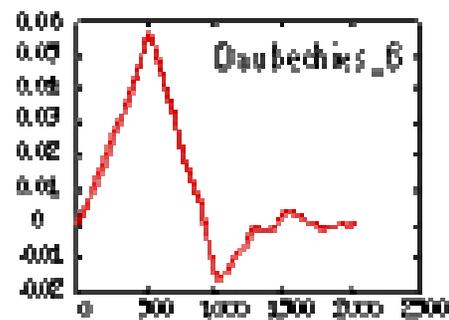


Transformada de *Wavelet* Contínua

Transformada de *Wavelet* Contínua é a **integral ao longo do tempo de um sinal multiplicado por uma escala, e deslocado** por uma função *wavelet* Ψ (*Psi*), também chamada *wavelet* mãe:

$$C(\text{escala}, \text{posição}) = \int_{-\infty}^{\infty} f(t) \Psi(\text{escala}, \text{posição}, t) dt$$

O número ao lado do nome da *wavelet* representa o número de momentos nulos



Repare que mais de uma forma (função base ou mãe) pode ser usada para **gerar uma família**

Para ser considerada uma wavelet, uma função inversível tem que:

Ter uma área total NULA sob a curva da função (ou integral nula) ; e

$$\int \Psi (t) dt = 0$$

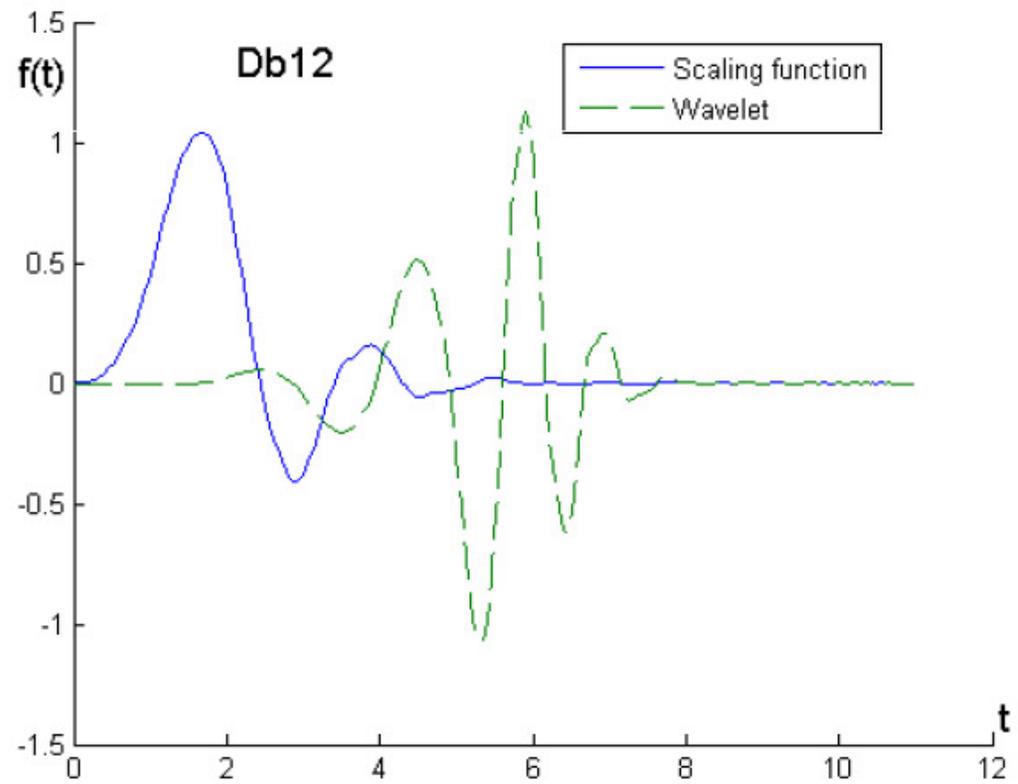
Ter energia (ou integral do quadrado da função) finita,

$$C_{\Psi} = 2 \pi \int |u|^{-1} \left| \hat{\Psi}(u) \right|^2 du < \infty$$

Famílias de wavelets

- Definem a forma da wavelet mãe
 - Haar
 - Daubechies
 - Biortogonal
 - Coiflets
 - Symlets
 - Shannon

Família Daubechies



Para que uma f seja uma Ψ

Área zero

$$\int_{-\infty}^{\infty} \psi(t) dt = 0$$

Energia finita:

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt$$

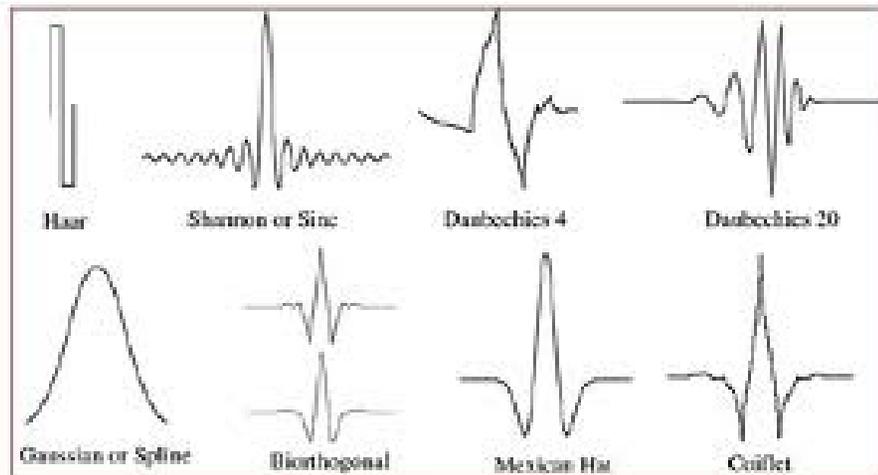


Figure 8

Examples of types of wavelets

Tem que ter um **suporte compacto**

- >

o que significa que ela deve **desaparecer** fora de um intervalo finito

Wavelet x Fourier

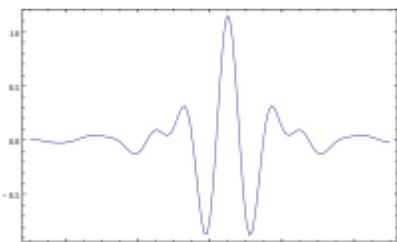
- Essa característica de **energia concentrada em uma região finita** é que diferencia as wavelets da Fourier
- Lembrem que a TF usa as funções de **seno e cosseno** que são **periódicas e infinitas**.
- a Teoria Wavelet foi estruturada **na década de 80**, embora as **origens da teoria Wavelet remontam aos anos 30**.
- Teoricamente há **infinitas possibilidades** de se projetar wavelets !!
- Pode-se projetar **wavelets otimizadas para realizar análises especiais**, onde as wavelets tenham características semelhantes aos sinais sob análise. Assim, wavelets que são utilizadas para **compressão de dados**, podem revelar-se péssimas para aplicações de análises de **sinais biológicos**, ou síntese de música.
- Obs. JPEG 2000 usa wavelets bi ortogonais.

Resumindo: a Transformada de *Wavelets* contínua em $F(a,b)$ é uma função de dois parâmetros reais, a e b :

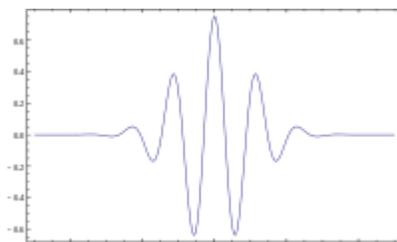
$$F(a,b) = \int f(t) \Psi_{a,b}(t) dt$$

A função $\Psi_{a,b}(t)$ é denominada **base wavelet** e definida como:

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right), \quad a \neq 0, \quad b \in \mathfrak{R}$$

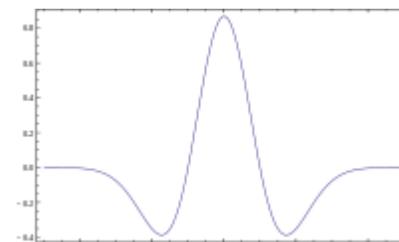


Meyer



Morlet

Chapeu Mexicano



A transformada de wavelet decompõe uma função definida no **domínio do tempo** em outra função, definida no **domínio do tempo e no domínio da frequência**.

Nestas transformações duas operações
são importantes:

the operators D (scales) and T (shift) defined by

$$y = Dx \quad \text{with} \quad y(t) = [\sqrt{2}x(2t)]$$

and

$$y = Tx \quad \text{with} \quad y(t) = x(t - 1)$$

A well-known wavelet associated with the above operators is the so-called *Haar wavelet* or *Daubechie 1*

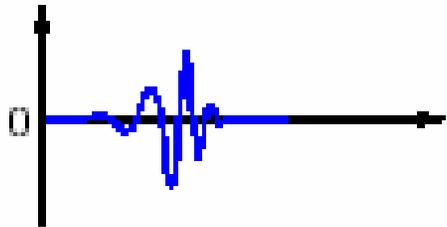
$$\psi(t) = \begin{cases} 1; & 0 < t \leq 1/2, \\ -1; & 1/2 < t \leq 1, \\ 0; & t \in R \setminus (0,1]. \end{cases}$$

A base Haar é simples, e útil para fins de explicação:

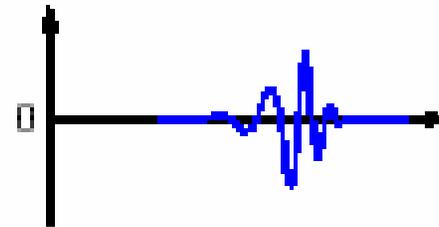
- Dilatações e translações da "função de mãe" definem uma base ortogonal (a base wavelet) é usada para descrever o sinal ou a imagem por combinações lineares desta.
- Como termo escala é usado na frequência e no denominador, deve se tomar cuidado ele não deve ser zero!

Wavelet Transform

Positional Parameter

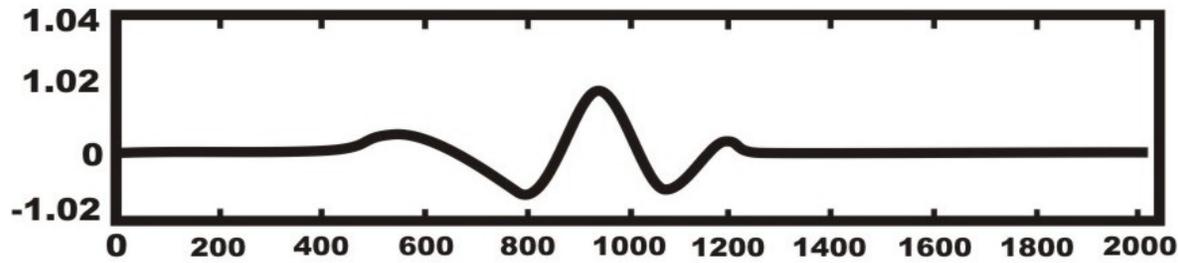


Wavelet



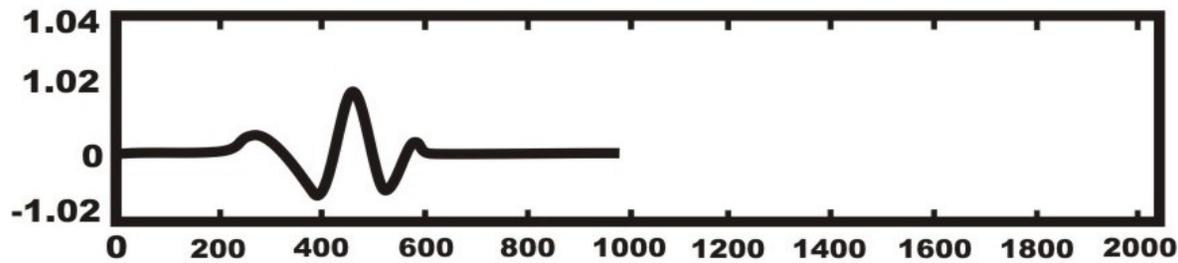
Same Function:
new location

A translação é usado no mesmo sentido, utilizado no STFT: está relacionado com a localização da janela

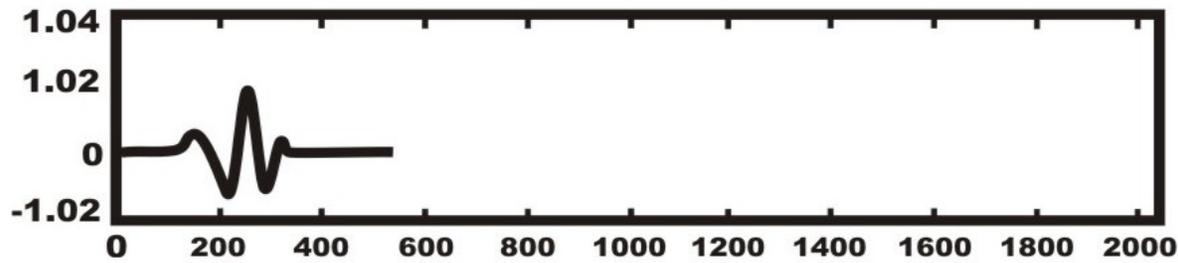


$$f(t) = \Psi(t) ; a = 1$$

mother wavelet
 $a=1, b=0$



$$f(t) = \Psi(2t) ; a = \frac{1}{2}$$

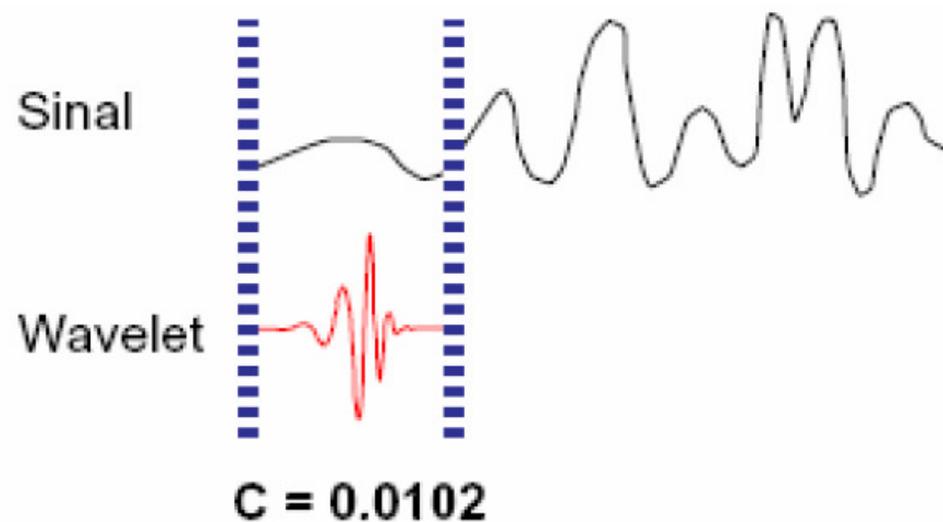


$$f(t) = \Psi(4t) ; a = \frac{1}{4}$$

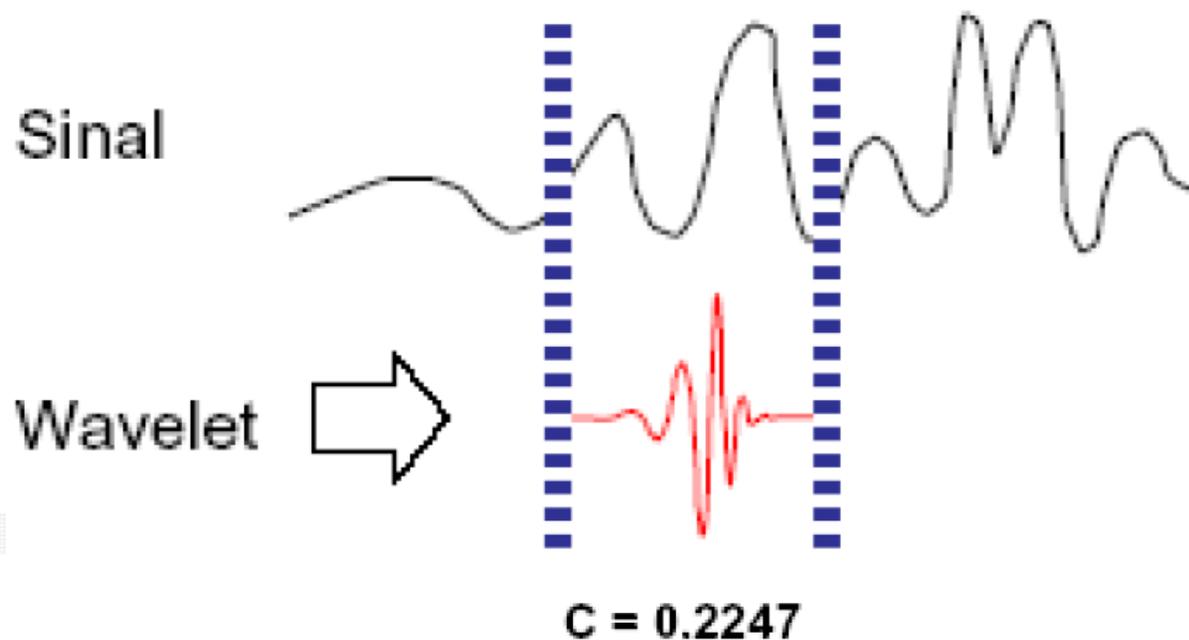
Exemplo de escala agindo em uma função *wavelet*

Análise de Wavelets (no contínuo)

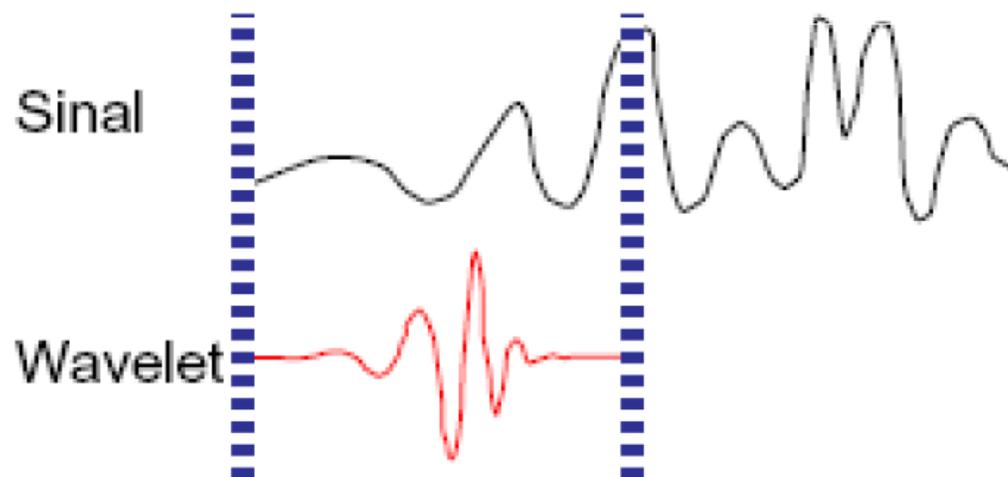
- 1- Tome uma função wavelet e compare-a com a região no início do sinal original
- 2- Calcule a correlação C entre a função e a região do sinal original



3- Desloque a wavelet para a direita e repita os passos 1 e 2 até que todo o sinal tenha sido coberto



4 – Aplique uma escala na wavelet e repita os passos 1 a 3

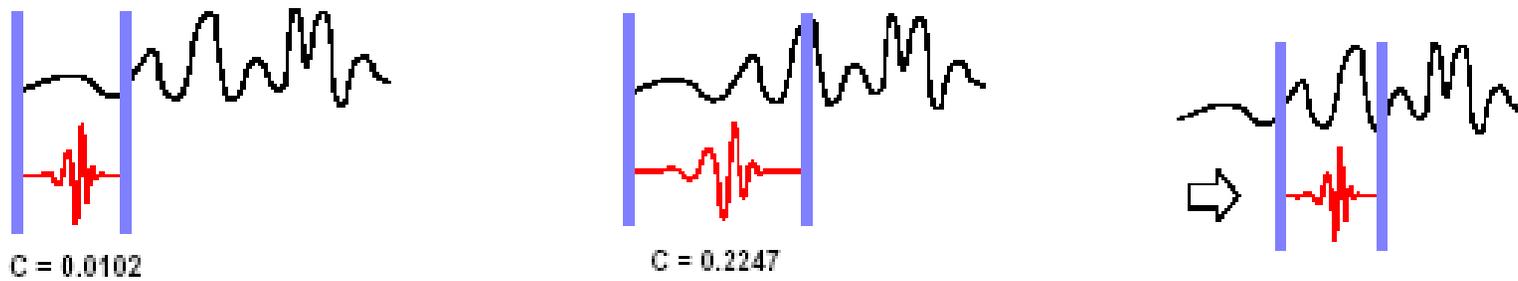


$$C = 0.2247$$

- 5 – Repita os passos 1 a 4 para todas as escalas
- Quando for concluído, você terá todos os coeficientes produzidos em diferentes escalas e diferentes seções do sinal
- Agora, o que fazer com essas informações?
- Um gráfico é plotado onde x representa a posição ao longo do sinal (tempo), y representa a escala e a cor em cada ponto (x,y) representa a magnitude do coeficiente da wavelet (C)

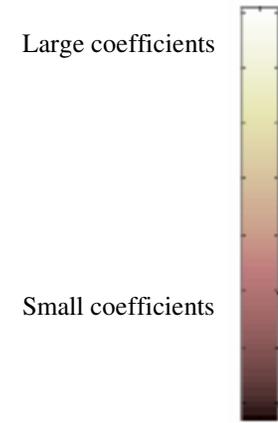
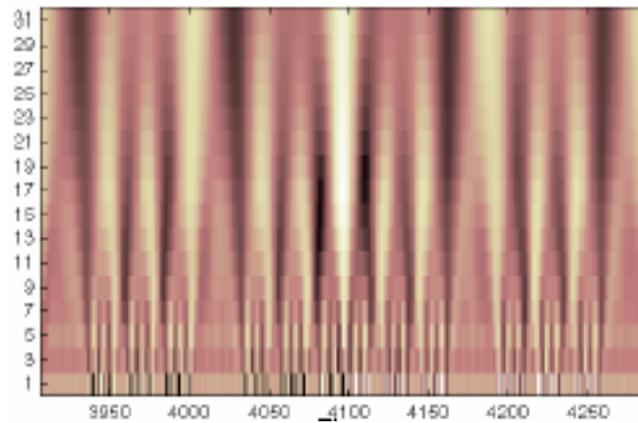
Wavelet Transform

location, scale, position and time:



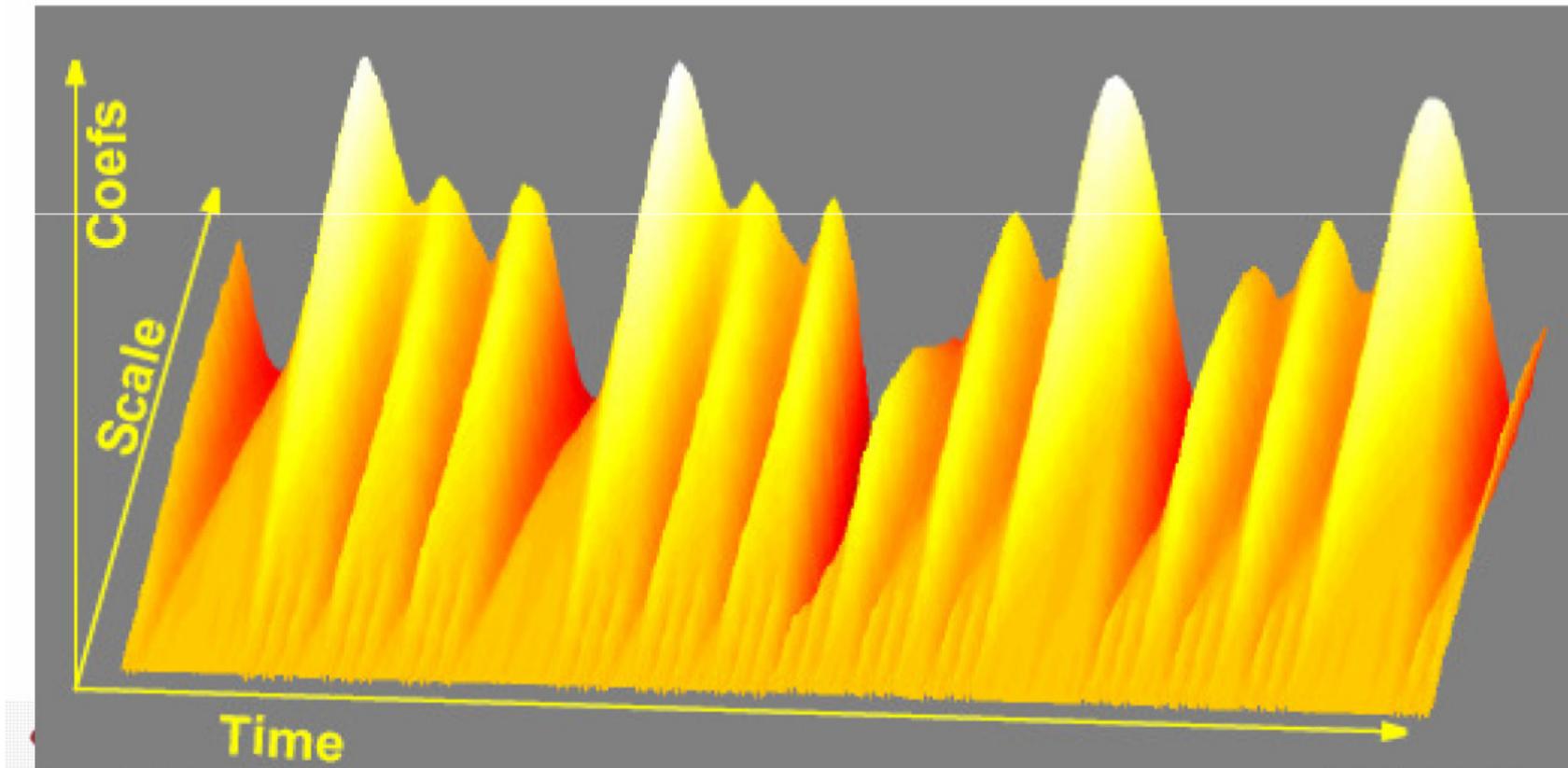
mother wavelet $a=1, b=0$

scale

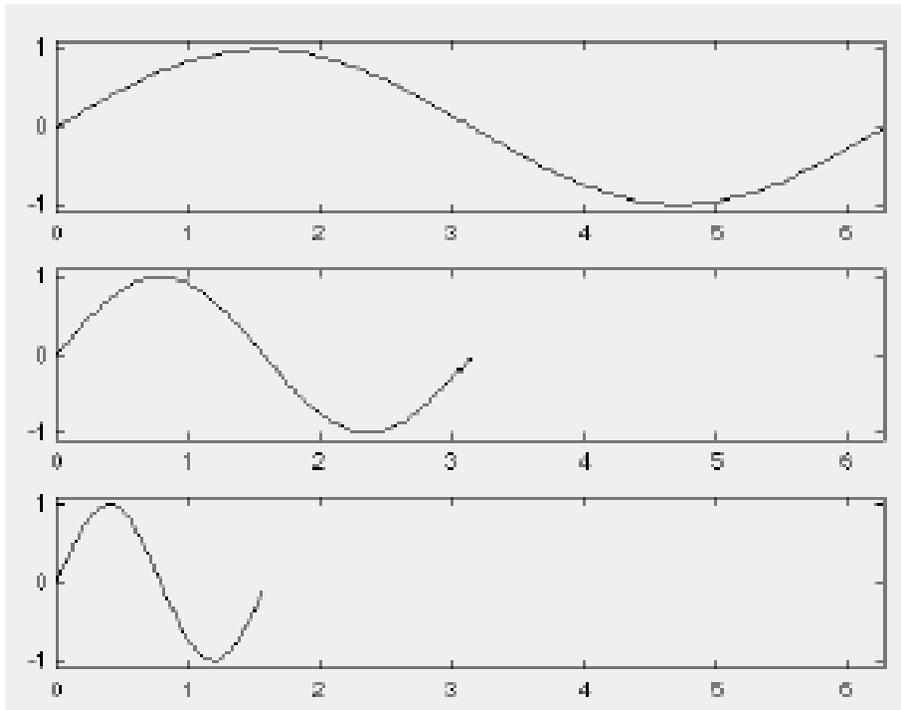


time

O mesmo gráfico anterior mas agora em altitude



Combinando escala em uma onda senoidal



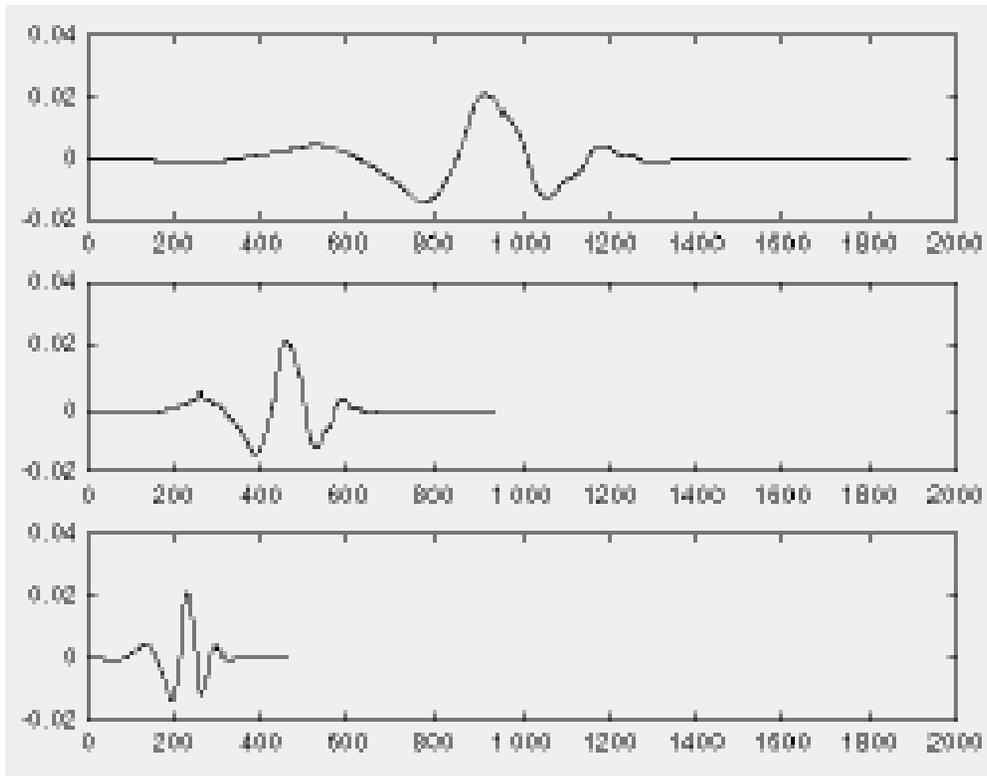
$f(t) = \text{sen}(t); a=1$
mother wavelet
 $a=1, b=0$

$f(t) = \text{sen}(2t); a=1/2$

$f(t) = \text{sen}(4t); a=1/4$

Wavelet Transform

Combinando escala e
+ a translação = uma família de Wavelets



$$f(t) = \Psi(t); a=1$$

$$f(t) = \Psi(2t); a=1/2$$

$$f(t) = \Psi(4t); a=1/4$$

escala

Wavelets

$$F(a, b) = \int f(t) \Psi_{a,b}(t) dt$$

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right), \quad a \neq 0, \quad b \in \mathfrak{R}$$

condição de admissibilidade

$$C_{\Psi} = 2\pi \int |u|^{-1} |\hat{\Psi}(u)|^2 du < \infty$$

$$\hat{\Psi}(0) = 0$$

A Transformada de *Wavelets* contínua em $F(a,b)$ é:

$$F(a,b) = \int f(t) \Psi_{a,b}(t) dt$$

A função $\Psi_{a,b}(t)$ é denominada *wavelet*:

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{|a|}} \Psi\left(\frac{t-b}{a}\right), \quad a \neq 0, \quad b \in \mathfrak{R}$$

As funções *wavelets* devem ter área zero e energia finita:

$$C_{\Psi} = 2\pi \int |u|^{-1} |\hat{\Psi}(u)|^2 du < \infty$$

condição de admissibilidade

Análise de *Wavelet* (AW)

A Análise de *Wavelet* (AW) é feita pela aplicação sucessiva da transformada de wavelet com diversos valores para a (resolução) e b ,

AW é uma ferramenta matemática para decomposição em nível hierárquico em um conjunto de aproximações e detalhes.

O nível hierárquico corresponde à escala diática (formado por potência de 2).

AW Permite a descrição de uma “função” em termos globais, mais termos que variam de detalhes globais até detalhes finos.

A “função” em questão pode ser uma imagem, uma curva, um exame médico, um objeto ou uma superfície.

Discrete Wavelet Transform

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right), \quad a = 2^j, b = k 2^j, \quad (j,k) \in \mathbb{Z}^2$$

mother wavelet $a=1, b=0 \Rightarrow j=0$ e $k=0$

- A escolha de escalas e posições é baseada em potências de 2
 - Escalas e posições diádicas (*dyadics*)
- Mallat (1988)
 - Implementa a análise de wavelets usando filtros
 - Conhecido como *two-channel subband coder*

Função mãe discreta: Haar

- Função mãe

$$\psi(x) \equiv \begin{cases} 1 & 0 \leq x < \frac{1}{2} \\ -1 & \frac{1}{2} < x \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Transformada de Haar

» Considerando j um inteiro positivo e

$$0 \leq k \leq 2^j - 1$$

$$\psi_{jk}(x) \equiv \psi(2^j x - k)$$

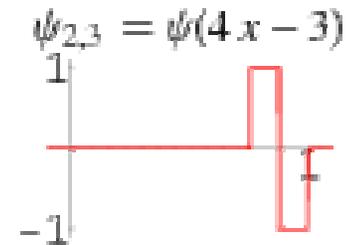
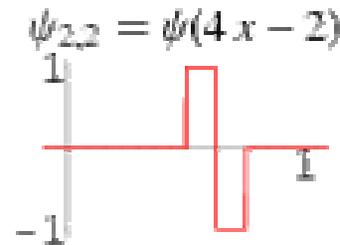
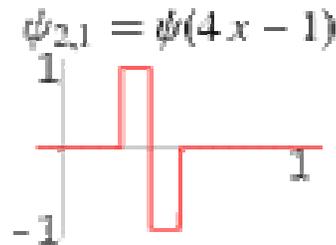
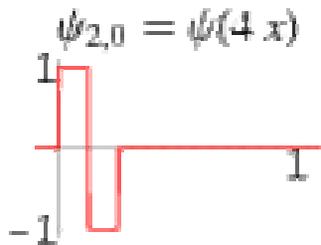
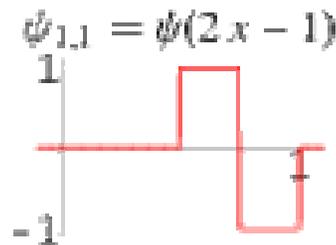
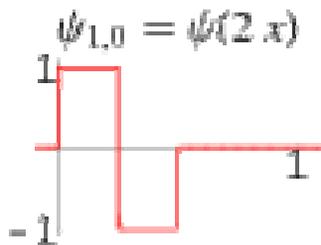
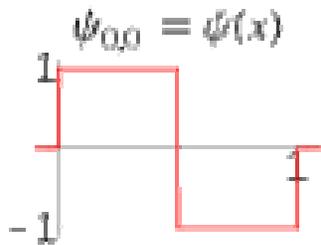
mother wavelet $j=k=0$

A família de Haar discreta:

Proposta pelo matemático Alfred Haar (húngaro) em 1909.

A transformada de Haar é um caso particular das de Ingrid Daubechies, onde é usado um pulso quadrado definido por:

$$\psi(x) \equiv \begin{cases} 1 & 0 \leq x < \frac{1}{2} \\ -1 & \frac{1}{2} < x \leq 1 \\ 0 & \text{otherwise} \end{cases}$$



- Criada por Ingrid Daubechies, a família das wavelets Daubechies (ou db) é a única família de wavelets que tem suporte compacto e decaimento suave
- O suporte compacto impede que a wavelet se espalhe por todo o espectro
- O decaimento suave impede que a wavelet introduza artefatos de altas frequências



Wavelet (db10)



Era uma vez...

- Depois de Haar em 1985, Stephane Mallat deu as wavelets mais um impulso em um trabalho em processamento digital de imagens.
- Y. Meyer construiu a primeira wavelet suave e continuamente diferenciáveis (mas sem suportes compactos, contidas em regiões finitas).
- Ingrid Daubechies (usando os trabalhos de Mallat) construiu um conjunto de bases ortonormais de wavelets suaves, com suportes compactos.
- Os trabalhos de Daubechies são as bases das aplicações atuais.
- Em 1989, Coifman sugeriu a Daubechies uma base ortormal de wavelets que foram denominadas de coifets
- <http://www.mat.ufmg.br/~lima/artigos/rmu33.pdf>

A base de um espaço vetorial V

conjunto de vetores linearmente independentes, que fazem com que qualquer vetor v em V pode ser escrito como uma combinação linear dos vetores da base.

A base de um espaço vetorial V

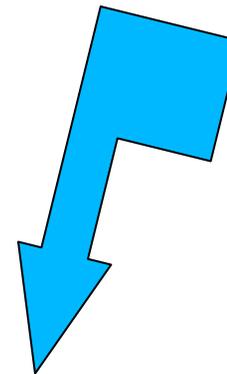
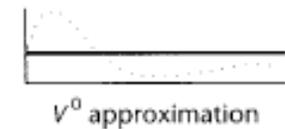
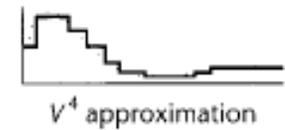
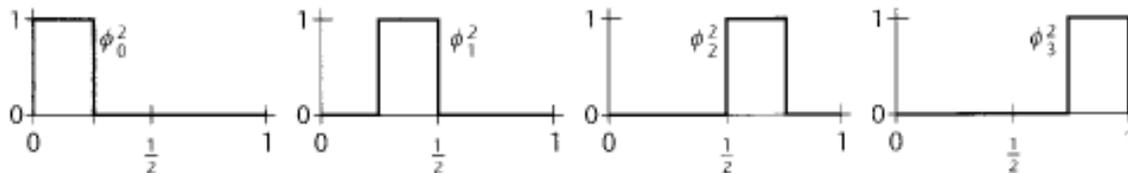
- é um conjunto de vetores linearmente independentes, que fazem com que qualquer vetor v em V pode ser escrito como uma **combinação linear dos vetores da base**.
- Pode haver **mais do que uma base** para um espaço vetorial.
- No entanto, todas têm o mesmo número de vetores, e este número é conhecido como a **dimensão do espaço vetorial**.
- Por exemplo, no espaço bidimensional, a base terá dois vetores. Esses podem ser $(1,0)$ ou $(0,1)$ para o \mathbb{R}^2 por exemplo.
- Os senos e cossenos são as funções de base para a TF.

Uma função $f(x)$ pode ser escrita como combinação linear de uma base.
 A wavelet de Haar esta associada a uma **base de ondas quadradas** em diversas resoluções.

Considerando diversos coeficientes c

$$f(x) = c_0 + \sum_{j=0}^{\infty} \sum_{k=0}^{2^j-1} c_{jk} \psi_{jk}(x).$$

<http://mathworld.wolfram.com/HaarFunction.html>



- Aproximações

- Componentes de alta escala e baixa frequência do sinal

- Detalhes

- Componentes de pequena escala e alta frequência do sinal

- Para a maior parte dos sinais, a informação de baixas frequências são as mais importantes
- Voz humana:
 - Se removermos as altas frequências, a fala ainda é compreendida embora a voz soe diferente

EXEMPLO, UM **SINAL COM 4 AMOSTRAS** e seus coeficientes de Haar:

$$f = \begin{pmatrix} 9 \\ 1 \\ 2 \\ 0 \end{pmatrix} = 3 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + 2 \begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \end{pmatrix} + 4 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + 1 \begin{pmatrix} 0 \\ 0 \\ 1 \\ -1 \end{pmatrix},$$

$$\begin{pmatrix} 3 \\ 2 \\ 4 \\ 1 \end{pmatrix}$$

A primeira amostra **do sinal a ser codificado** contém o **coeficiente que descreve o componente DC (MÉDIA GERAL)**.

Esse coeficiente que descreve o sinal como uma única wavelet Haar.

Em seguida, o coeficiente para duas wavelets Haar (com comprimento metade da primeira)

Assim por diante cada uma com metade da wavelet anterior.

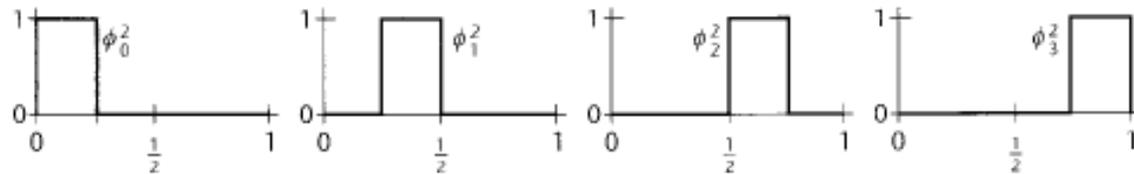
médias				R e s. 4	Deta- lhes	
9	1	2	0			
5		1		2	4	1
3				1	2	

Para chegar aos coeficientes calculamos em diferentes resoluções esses valores de médias e detalhes para reconstruir o sinal

- A **resolução** corresponde ao número de divisões do intervalo $[0,1]$ que consideramos.
- Os valores de média correspondem, na representação codificada do sinal, a sua descrição por uma **onda quadrada** (no caso de Haar) .
- Os detalhes são relacionados a forma da base da **família Haar** usada (ondas de **área zero** de Haar)

Outro exemplo: Qual seria a representação do sinal:
9 7 3 5 em TW de Haar?

- Repare que esse sinal pode ser entendido como decomposto no mesmo nível de resolução das bases:

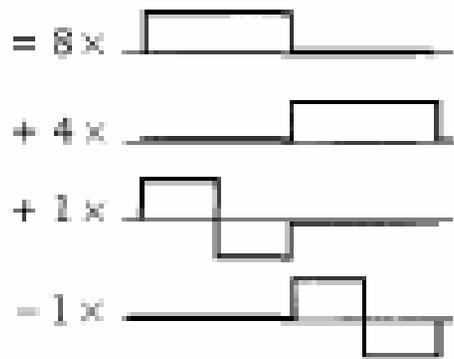


Ou seja ficaria como
 (mas ai todas as ondas estão com
 a **mesma resolução**)

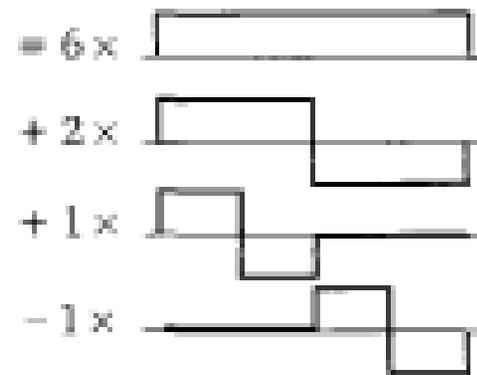
$$f(x) = 9 \times \begin{array}{c} \text{[rectangle from } x=0 \text{ to } x=1/2 \text{]} \\ \text{---} \end{array} \\
 + 7 \times \begin{array}{c} \text{[rectangle from } x=1/4 \text{ to } x=3/4 \text{]} \\ \text{---} \end{array} \\
 + 3 \times \begin{array}{c} \text{[rectangle from } x=3/8 \text{ to } x=5/8 \text{]} \\ \text{---} \end{array} \\
 + 5 \times \begin{array}{c} \text{[rectangle from } x=7/8 \text{ to } x=1 \text{]} \\ \text{---} \end{array}$$

Mas podemos também re-escrê-lo considerando **médias** e **detalhes** como:

$$I(x) = c_0^1 \phi_0^1(x) + c_1^1 \phi_1^1(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x)$$



$$I(x) = c_0^0 \phi_0^0(x) + d_0^0 \psi_0^0(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x)$$

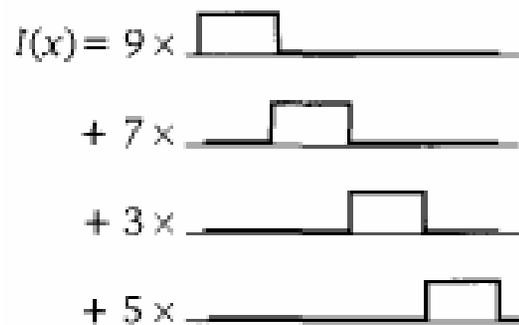


Resolution	Averages	Detail Coefficients
4	9 7 3 5	
2	8 4	1 -1
1	6	2

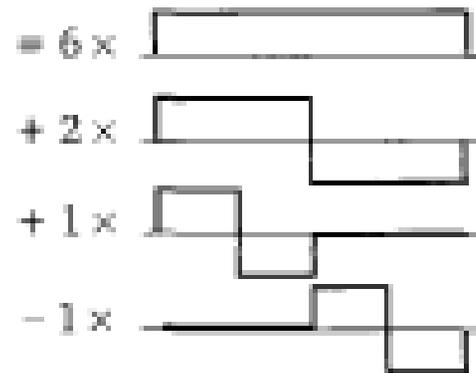
(agora estamos mesmo fazendo análise em multi resolução)

Aplicação em compressão sem perdas

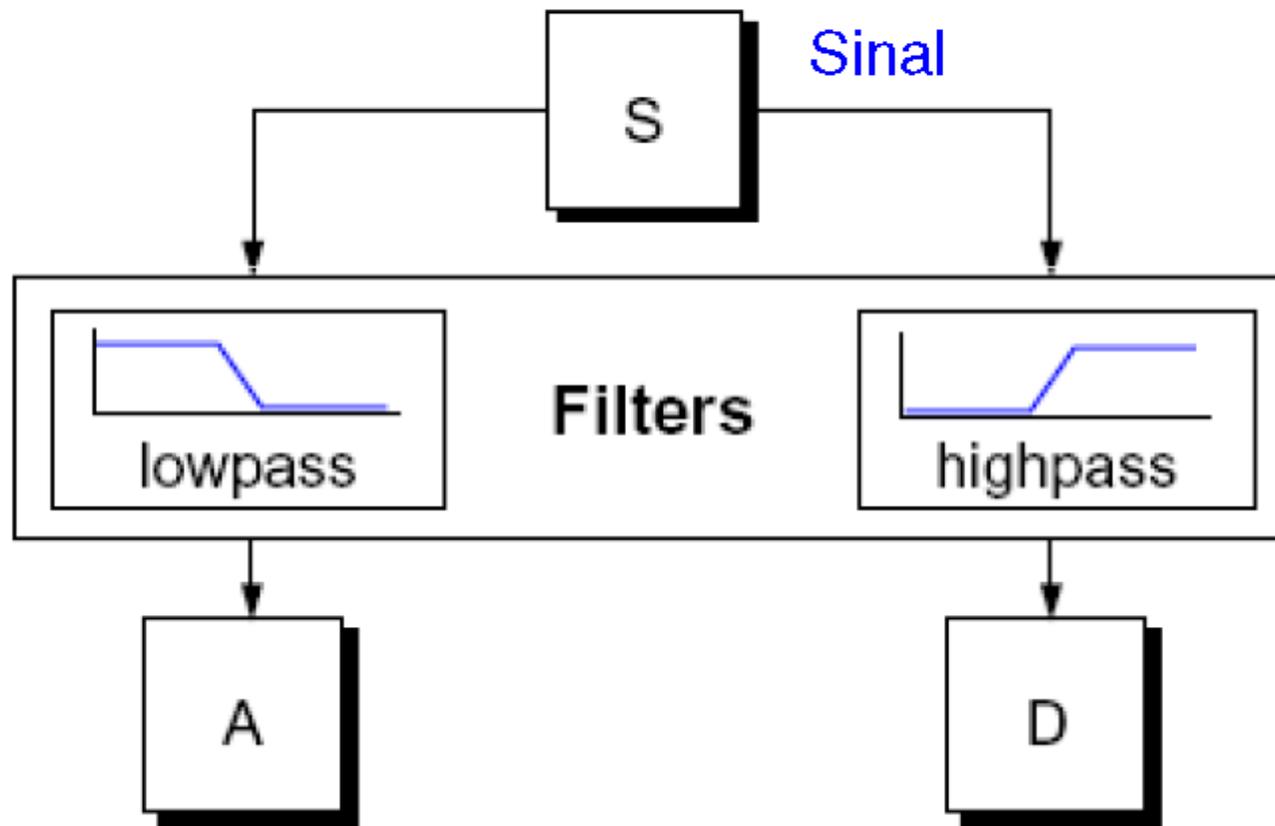
- Repare que só em escrever esse sinal na última base já temos menos espaço de armazenamento do mesmo (11 bits x 7 bits)



$$I(x) = c_0^0 \phi_0^0(x) + d_0^0 \psi_0^0(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x)$$



Processo de filtragem



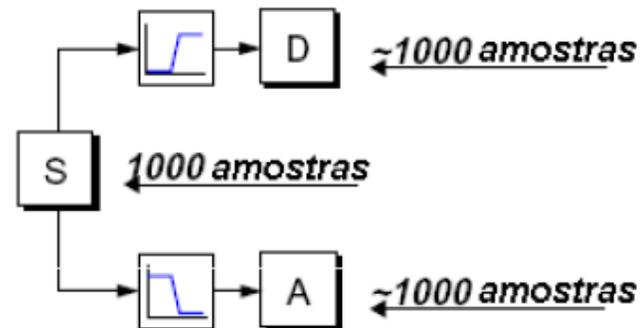
Aproximações

Detalhes

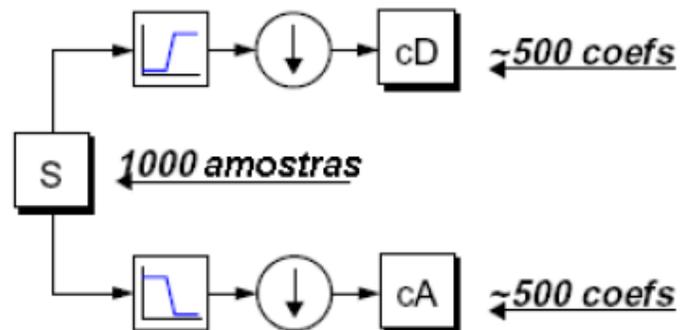
Processo de filtragem

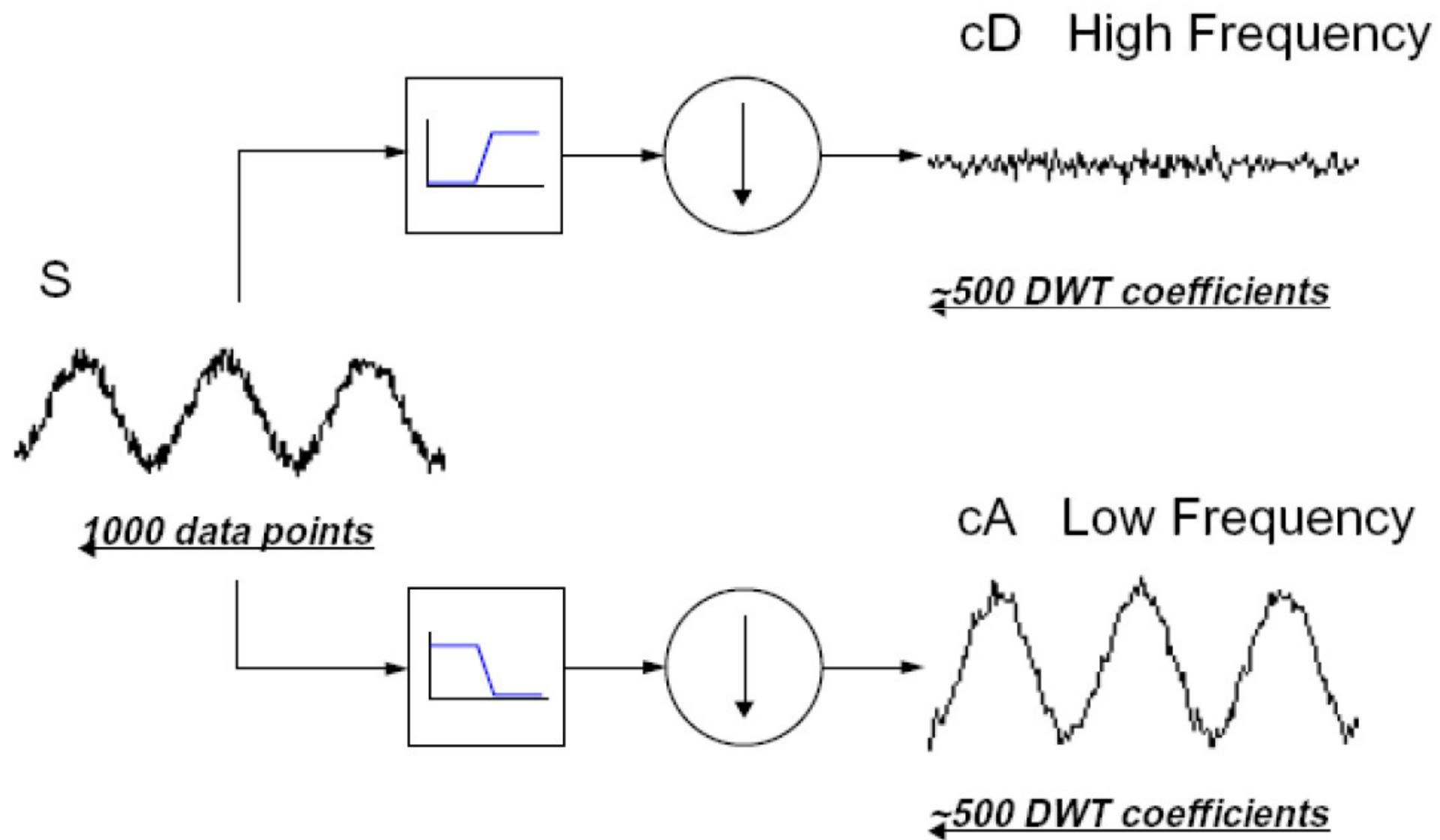
- Problema: Suponha que o sinal original tem 1000 amostras...
 - Com esse processo de filtragem, passamos a ter dois sinais cada um com 1000 amostras
 - Esse aumento na quantidade de dados não é interessante
 - Podemos pegar apenas uma em cada duas amostras dos sinais filtrados
 - Assim, ficaríamos apenas com as 1000 amostras
 - Essa é a idéia de *Downsampling*

Ou seja, ao invés de:

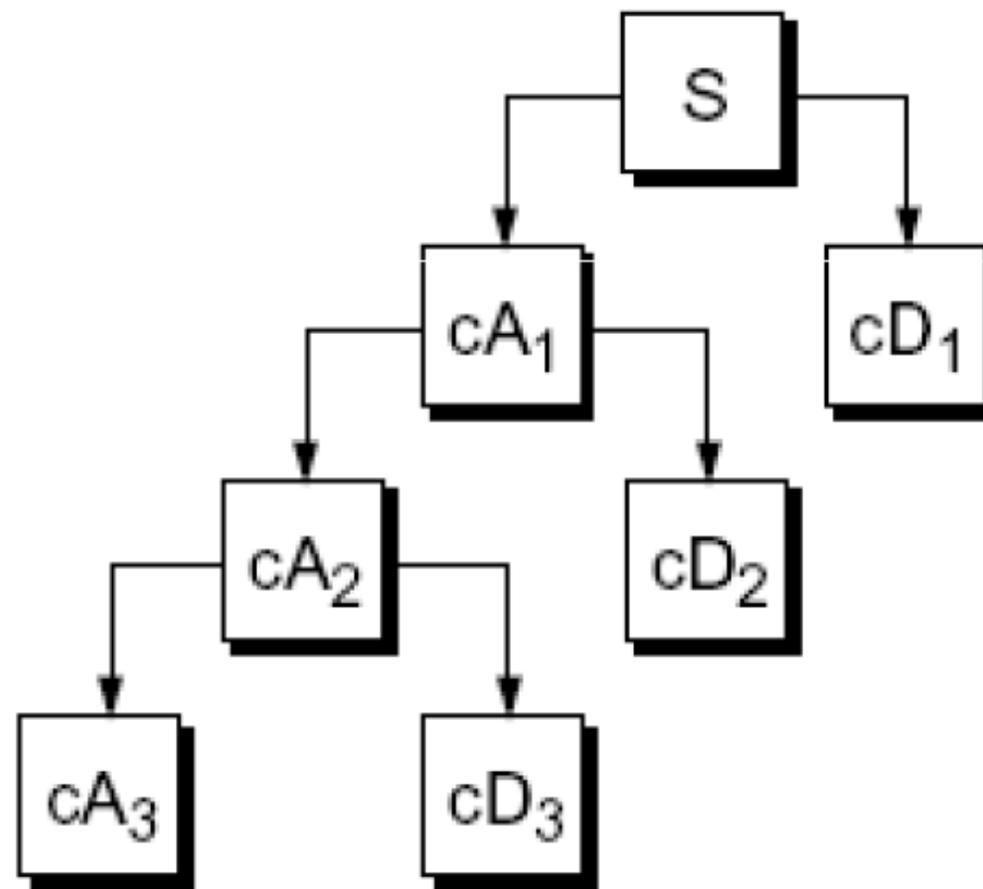


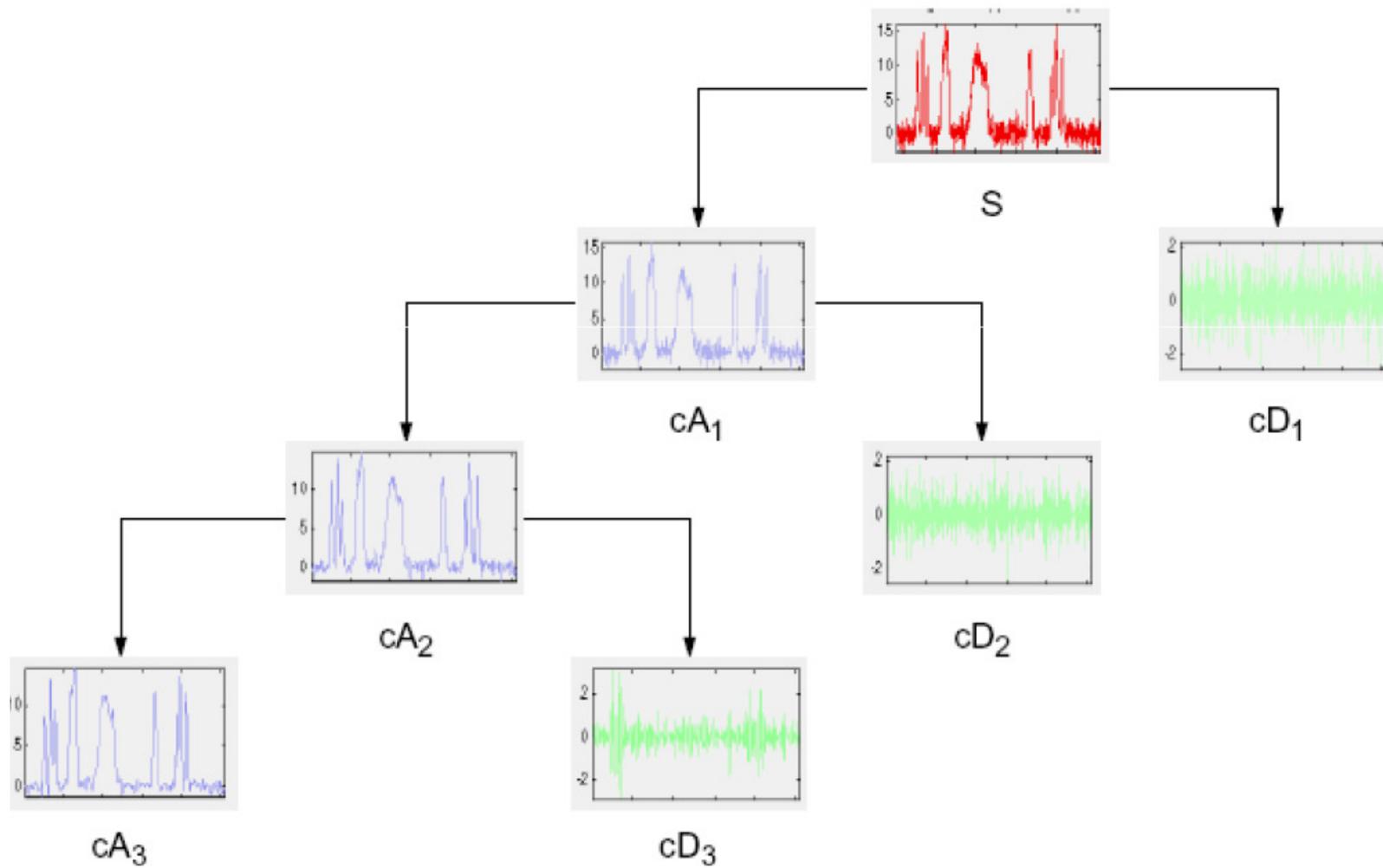
Temos agora:





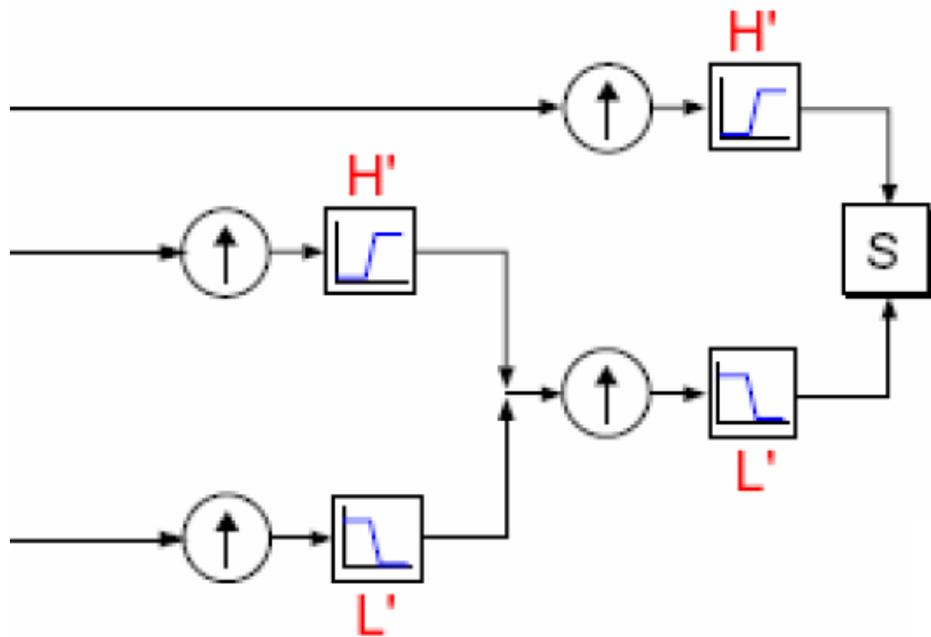
Decomposição em Múltiplos Níveis





O que a análise de wavelets faz é decompor um sinal

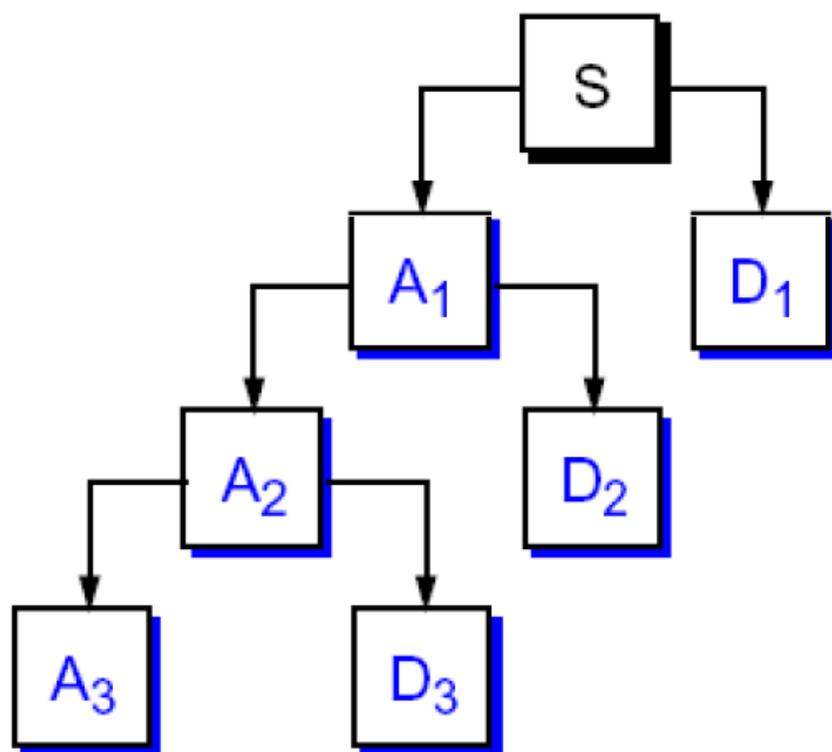
O processo inverso é a reconstrução ou síntese do sinal



Upsampling ao invés de Downsampling

O processo inverso é a reconstrução ou síntese do sinal

- Componentes de um sinal reconstruído



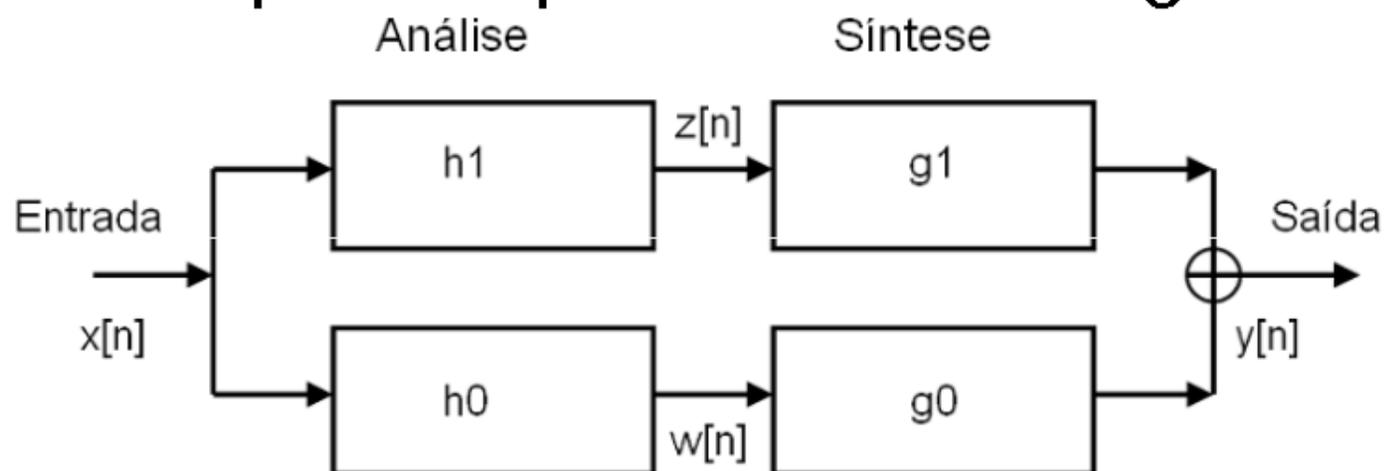
$$S \approx A_1 + D_1$$

$$\approx A_2 + D_2 + D_1$$

$$\approx A_3 + D_3 + D_2 + D_1$$

Coeficientes das Wavelets

Para demonstrar uma transformada geral, vamos usar o esquema apresentado na Fig. abaixo:



Nela, um sinal de entrada alimenta dois canais, cada qual com um par de filtros FIR. Tal estrutura é chamada de *two-channel filter banks*.

Coeficientes das Wavelets

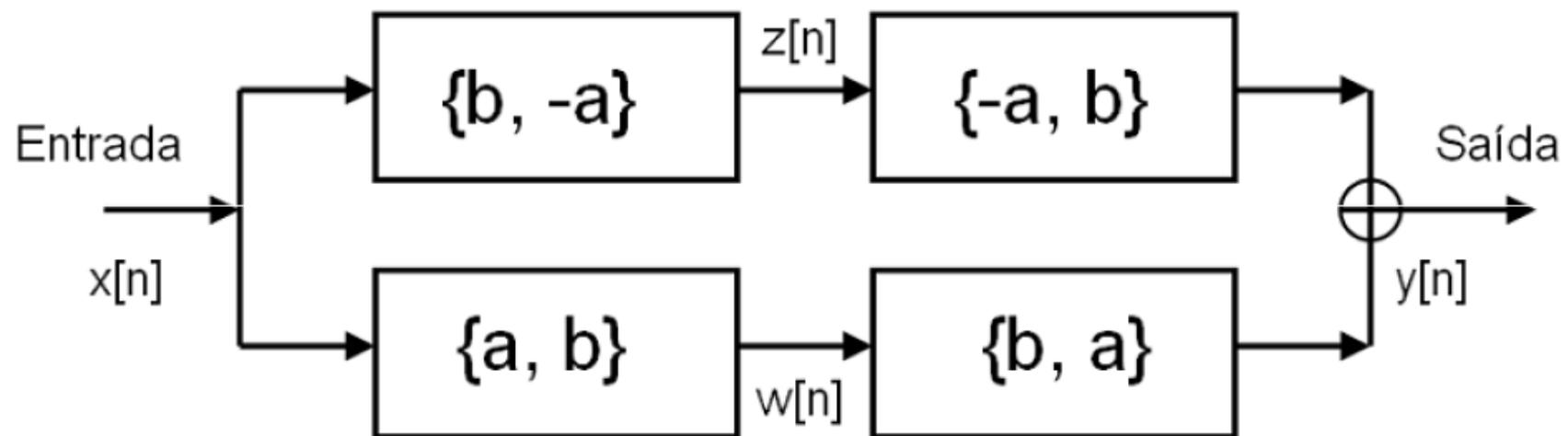
- A metade da esquerda da figura (filtros h_0 e h_1) corresponde à transformada direta
 - Análise
- A metade da direita é a transformada inversa
 - Síntese
- É esperado que a síntese gere um sinal de saída igual ao sinal de entrada
- Os filtros complementares de banco de filtros (h_0 para h_1 e g_0 para g_1) dividem o sinal em sub-sinais de baixa e alta frequência
- Isso é chamado de *subband coding*

Coeficientes das Wavelets

- Para a transformada inversa, as saídas dos filtros da parte de análise ($z[n]$ e $w[n]$) passam por outros filtros FIR e, em seguida, são combinadas para gerar a saída $y[n]$
- A idéia é que $w[n]$ e $z[n]$ sejam versões transformadas de $x[n]$ e que $y[n]$ é o sinal após a transformada inversa
- Como dito anteriormente, espera-se que $y[n]$ seja igual a $x[n]$

Coeficientes das Wavelets

- Vamos considerar o *filter bank* da figura abaixo e vamos ver como se comportam $w[n]$ e $z[n]$:



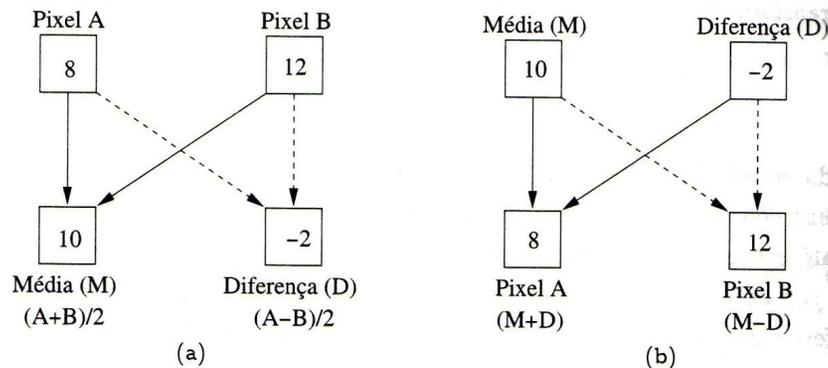
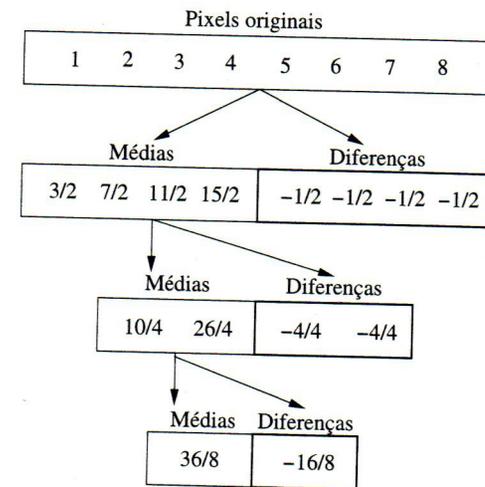
- Na parte da análise, $w[n]$ e $z[n]$ são definidos por:
 - $w[n] = a.x[n] + b.x[n - 1]$
 - $z[n] = b.x[n] - a.x[n - 1]$

Coeficientes das Wavelets

- Precisamos saber também que são $w[n - 1]$ e $z[n - 1]$:
 - $w[n - 1] = a.x[n - 1] + b.x[n - 2]$
 - $z[n - 1] = b.x[n - 1] - a.x[n - 2]$
- Assim, $y[n]$ será:
 - $y[n] = -a.z[n] + b.z[n - 1] + b.w[n] + a.w[n - 1]$
- E, em relação ao sinal original:
 - $y[n] = -a.(b.x[n] - a.x[n - 1]) + b.(b.x[n - 1] - a.x[n - 2]) + b.(a.x[n] + b.x[n - 1]) + a.(a.x[n - 1] + b.x[n - 2])$
 - $y[n] = -ab.x[n] + aa.x[n - 1] + b.b.x[n - 1] - ba.x[n - 2] + b.a.x[n] + b.b.x[n - 1] + a.a.x[n - 1] + ab.x[n - 2]$
 - $y[n] = aa.x[n - 1] + bb.x[n - 1] + bb.x[n - 1] + aa.x[n - 1]$
 - $y[n] = (2aa + 2bb).x[n - 1]$

Como automatizar o processo de codificação e de reconstrução?

- Cada conjunto : resolução, base, função de escala é representada por um conjunto de filtros de médias e detalhes aplicado até um determinado nível.



Alfréd Haar, 1909



Filtros normalizados de Haar

- Os filtros normalizados de Haar passam a ser:

$$\phi_j^l(x) := 2^{j/2} \phi(2^j x - i)$$

$$\psi_j^l(x) := 2^{j/2} \psi(2^j x - i)$$

- De modo que os coeficientes do exemplo anterior mudam de $[1 \ 2 \ 1 \ -1]$ para :

$$[\frac{1}{\sqrt{2}} \ \frac{-1}{\sqrt{2}}]$$

- Ou seja os filtros ficam:

$$l = \left[\frac{1}{\sqrt{2}} \ \frac{1}{\sqrt{2}} \right] \quad h = \left[\frac{1}{\sqrt{2}} \ -\frac{1}{\sqrt{2}} \right]$$

Filtros normalizados de Daubechies

- Os filtros normalizados de Daubechies :

$$l = \left[\frac{1+\sqrt{3}}{4\sqrt{2}} \quad \frac{3+\sqrt{3}}{4\sqrt{2}} \quad \frac{3-\sqrt{3}}{4\sqrt{2}} \quad \frac{1-\sqrt{3}}{4\sqrt{2}} \right]$$

$$h = \left[\frac{1-\sqrt{3}}{4\sqrt{2}} \quad -\frac{3-\sqrt{3}}{4\sqrt{2}} \quad \frac{3+\sqrt{3}}{4\sqrt{2}} \quad -\frac{1+\sqrt{3}}{4\sqrt{2}} \right]$$

Uma função $f(x)$ pode ser escrita como

Considerando diversos coeficientes c

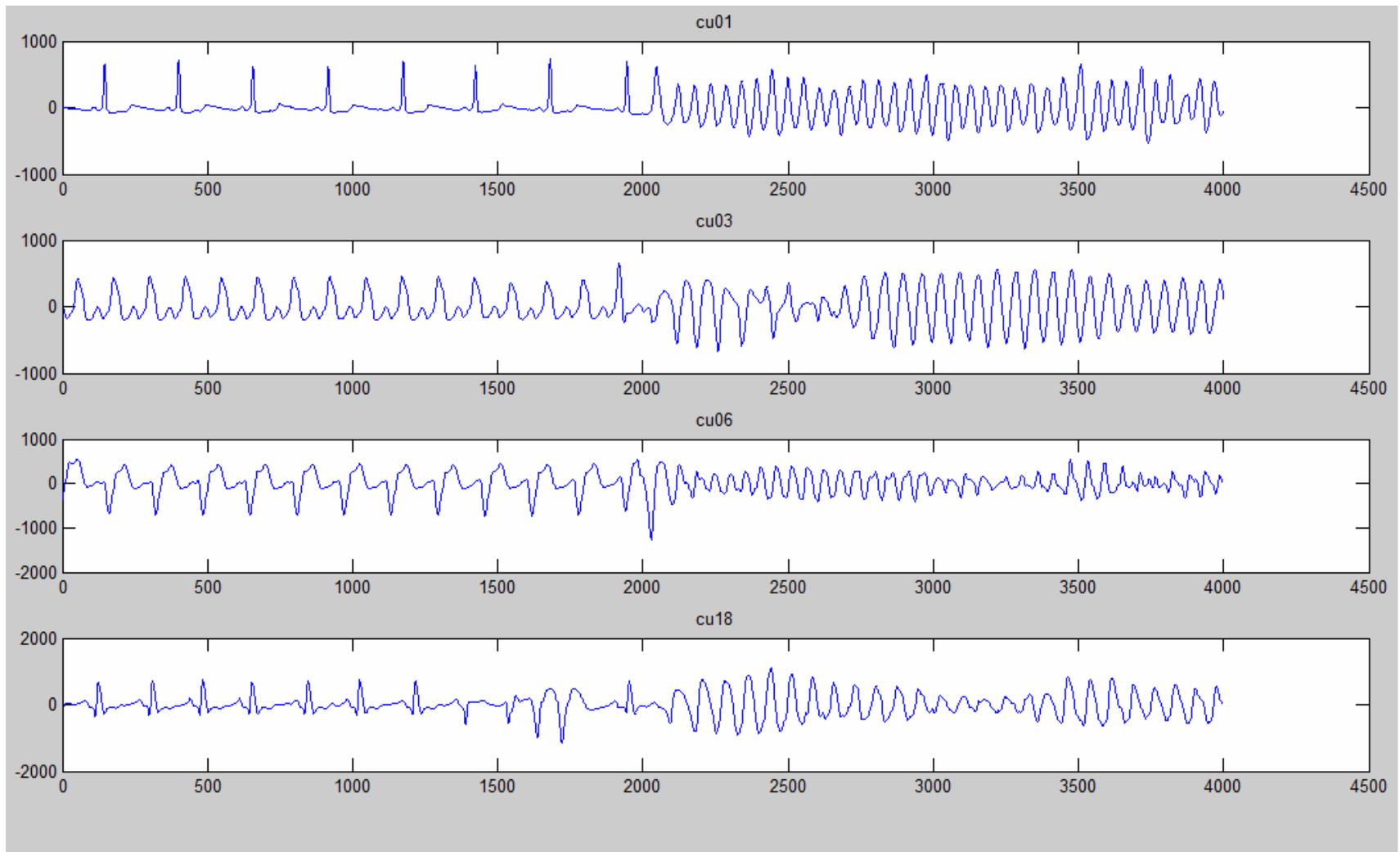
$$f(x) = c_0 + \sum_{j=0}^{\infty} \sum_{k=0}^{2^j-1} c_{jk} \psi_{jk}(x).$$

<http://mathworld.wolfram.com/HaarFunction.html>

Experimentos com ECG em 2016

Usando os coeficientes de wavelets poderíamos detectar a Ventricular fibrillation?

Qual e melhor base para detectar a Ventricular fibrillation ?



Intuitivamente

- nós sabemos que a frequência é algo a ver com a taxa de alteração de alguma coisa.
- Se algo (uma variável) muda rapidamente, dizemos que tem alta frequência, e
- se esta variável mudar lentamente, dizemos que tem baixa frequência.
- Quando a variável não muda nunca, dizemos que ela tem frequência zero, ou não tem nenhuma frequência.
- A frequência é medida em ciclos por segundo, ou em unidades de "Hertz".

Quadrature mirror filter pair

- É útil pensar os coeficientes como um filtro.
- Os filtro ou coeficientes são colocados em uma matriz de transformação, o qual é aplicado a um vetor de dados.
- Os coeficientes são ordenados usando dois padrões dominantes: um que funciona como um filtro de suavização (média), e outro que trabalha para obter os dados da "detalhe" da informação.
- Essas duas ordenações dos coeficientes são chamados de um par de espelhados de quadratura (quadrature mirror filter pair) .

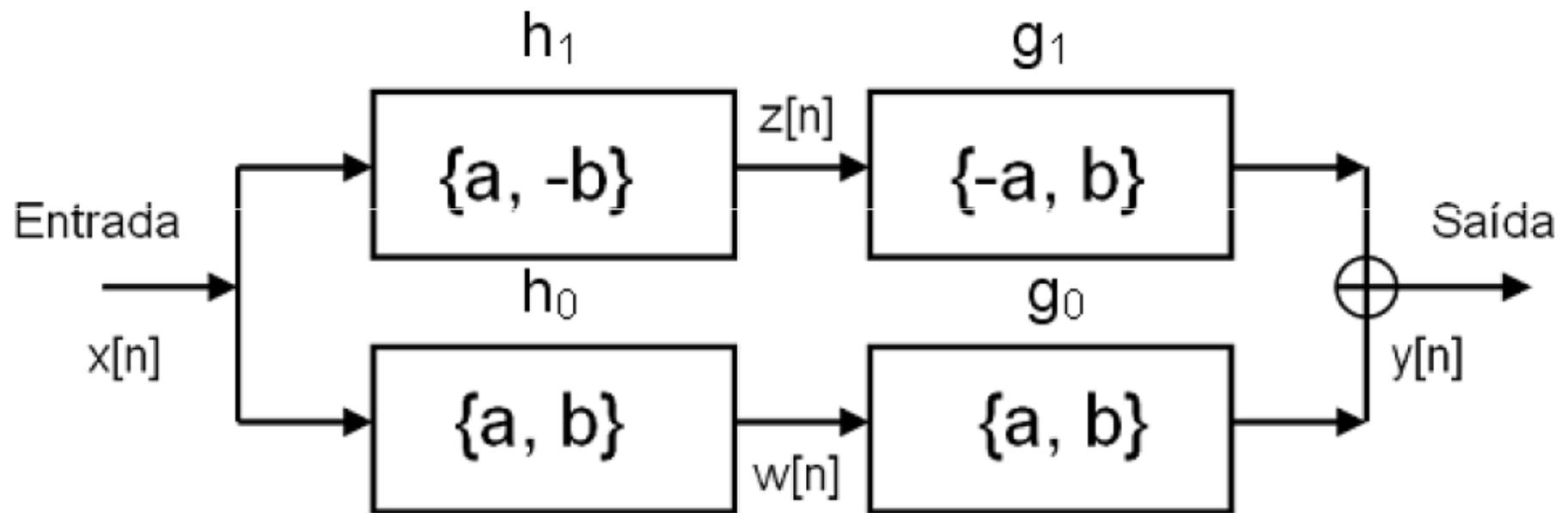
Algumas vezes, um *two channel filter bank* é chamado de *quadrature mirror filter* (QMF)

Um QMF é um *filter banks* com condições especiais nos coeficientes dos filtros para evitar aliasing e conseguir uma reconstrução perfeita

Ou seja, sua única diferença para *filter banks* é a escolha dos coeficientes dos filtros

- *Quadrature mirror filter (QMF):*
 - Seja um dos filtros da análise h_0
 - Os outros filtros (h_1 , g_0 e g_1) são gerados a partir de h_0
 - Especificamente, h_1 usa os mesmos coeficientes de h_0 , mas negativa os coeficientes de índice par
 - O filtro de reconstrução g_0 é igual a h_0 e $g_1 = -h_1$

1 *Quadrature mirror filter (QMF):*



Numa **imagem típica**

- o que se vê são regiões enormes onde os valores dos pixels **são muito próximos**, o que significa que os coeficientes de wavelets associadas ou **são nulos ou desprezíveis**.
- Somente em regiões de transições, próximas aos **contornos** onde os valores dos “pixels” variam muito, teremos uma mudança significativa nos valores dos “pixels”, portanto, haverá **coeficientes de wavelets apreciáveis**.
- Será que o mesmo ocorre com **imagens térmicas**?

- DWT emprega dois conjuntos de funções, chamadas funções de escala e funções wavelet, que estão associados com filtros passa-baixa e passa-alta, respectivamente.
- A decomposição do sinal em bandas de frequência diferentes são simplesmente obtidas por sucessivas filtragem do sinal no domínio do tempo por esses filtros.
- O sinal original $x[n]$ é primeiro analisado através de um filtro passa-alta halfband $g[n]$, e um filtro passa-baixo $h[n]$.
- Após a filtragem, a metade das amostras pode ser eliminada de acordo com a regra de Nyquist, uma vez que o sinal tem agora uma frequência alterada.
- O sinal pode, portanto, ser sub-amostrado, simplesmente, descartando uma parte da amostra.
- Isto constitui um nível de decomposição

Em 2D

- 1º Passo: Expansão da imagem para que tenha número de colunas e número de linhas múltiplo de 2^n
 - Não amplia a imagem, apenas a expande
 - Cria efeito de bordas
 - Necessário para o *Downsampling*
- 2º Passo: Define os coeficientes do filtro (wavelet mãe)
 - Chamaremos de *Wave*

- 3º Passo: Cálculo dos filtros (vetores):
 - Lo_R // filtro de reconstrução passa-baixa
 - $Lo_R[i] = \sqrt{2} * Wave[i] / \sum Wave[i]$
 - Lo_D // filtro de decomposição passa-baixa
 - Lo_D = O inverso de Lo_R
 - Hi_R // filtro de reconstrução passa-alta
 - $Hi_R = qmf(Lo_R)$
 - Inverte o vetor e muda o sinal de todos os elementos que estão em posição de índice par
 - Hi_D // filtro de decomposição passa-alta
 - Hi_D = O inverso de Hi_R

- 4º Passo: Gera as matrizes de Aproximação e Detalhes
 - $a = Lc(Lr(\text{imagem}))$
 - Matriz de coeficientes para a aproximação de um nível
 - $h = Hc(Lr(\text{imagem}))$
 - Matriz de coeficientes para detalhes horizontais
 - $v = Lc(Hr(\text{imagem}))$
 - Matriz de coeficientes para detalhes verticais
 - $d = Hc(Hr(\text{imagem}))$
 - Matriz de coeficientes para detalhes diagonais

- 4º Passo: Continuação (em termos de algoritmo)

- $Y = \text{Imagem} *_L \text{Lo_D}$

- $a = Y *_C \text{Lo_D}$

- $h = Y *_C \text{Hi_D}$

- $Y = \text{Imagem} *_L \text{Hi_D}$

- $v = Y *_C \text{Lo_D}$

- $d = Y *_C \text{Hi_D}$

- $*_L = \text{Convolução em linha}$

- $*_C = \text{Convolução em coluna}$

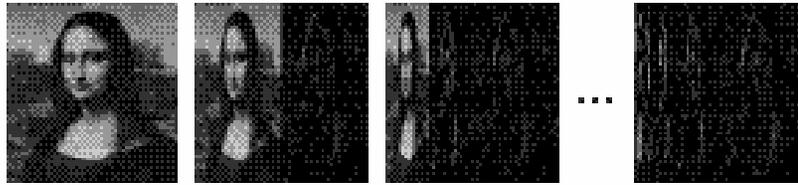
- 5º Passo: Retorna a matriz de Aproximações
 - Saída da função
- 6º Passo: Volta a matriz às dimensões originais da imagem caso tenha havido a expansão

Em uma imagem digital ou sinal (bi-dimensional),

- Calcula-se seus coeficientes de wavelets tratando suas linhas e suas colunas como se fossem sinais ou “imagens unidimensionais”.
- Imagine que tenhamos uma imagem com $2^k \times 2^k$ pixels, a qual pode ser armazenada numa matriz quadrada, $A[i; j]$, $i; j = 0; \dots 2^k - 1$.
- Cada linha ou coluna é considerada como se fosse uma imagem unidimensional, aplicando-se o processo de obtenção dos coeficientes de wavelets, separadamente.
- Existem dois tipos de decomposições de imagens digitais: a **padrão** (vai-se até o fim por linhas de pois por colunas) e a **não padrão** ou **piramidal** (decompomos cada linha aplicando-se apenas um passo no processo e depois, tratamos cada coluna resultante).
- De modo que uma imagem unidimensional (ou sinal) com 2^k pixels é uma a seqüência de números onde cada um é a intensidade correspondente ao valor do pixels

Bidirectionally

Transform lines →

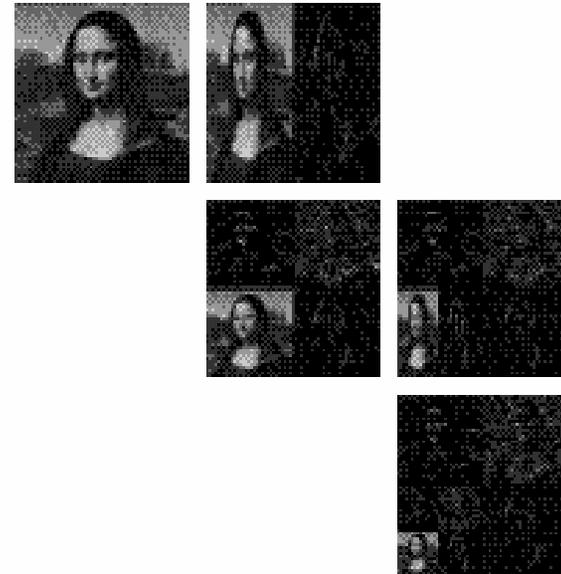


Transform columns ↓

(a)

Standard Decomposition

Transform lines →

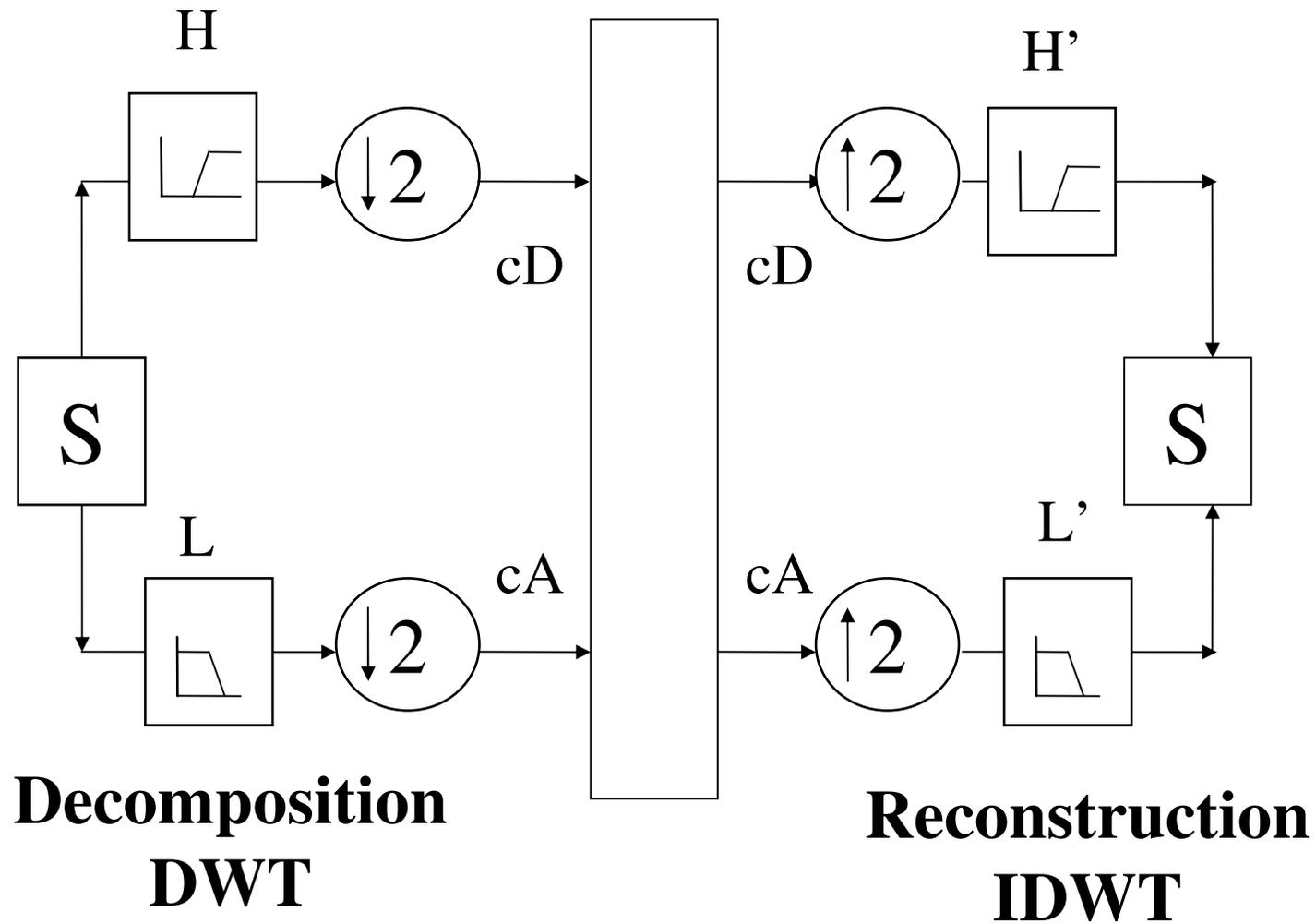


Transform columns ↓

(b)

Pyramidal Decomposition

Low and High frequency components



Onde ocorre a compressão dos dados?

- Exemplo de reconstrução de uma imagem usando-se diferentes percentagens de seus coeficientes de wavelets.
- Note que mesmo usando apenas cerca de 5% dos coeficientes de wavelets a reconstrução é perfeita
- Só se inicia a ver defeitos com menos de 2% dos coeficientes.



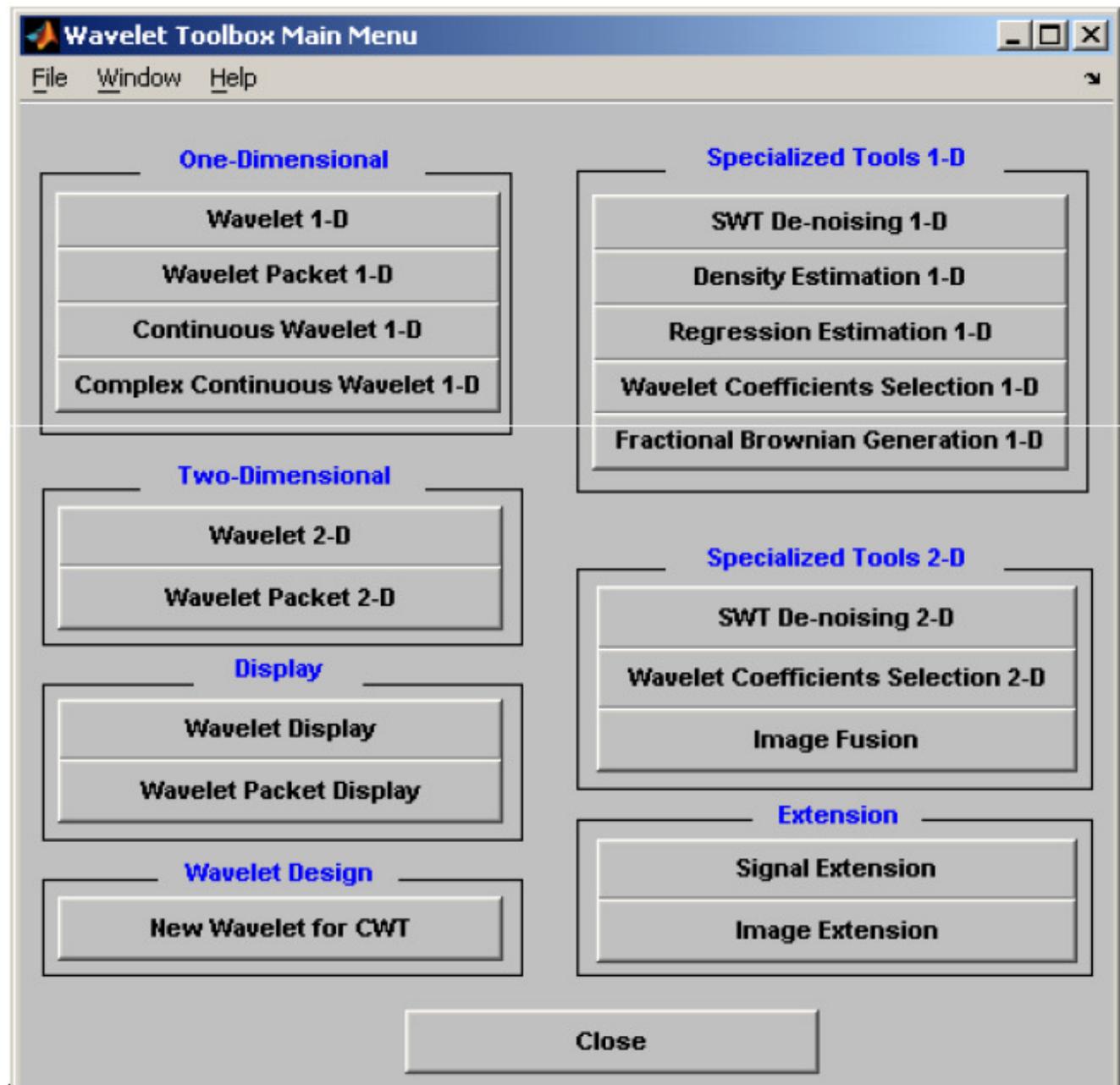
Reconstrução com a wavelet **Daub4**

WA => codificação esparça

- Uma vez que o sinal original ou a função pode ser representado em termos de expansão de wavelet (utilizando coeficientes de uma combinação linear das funções wavelet)
- Operações pode ser executada utilizando apenas os coeficientes wavelet correspondentes.
- Pode-se ainda escolher as melhores formas adaptadas aos dados, e truncar os coeficientes abaixo de um limite (dados pouco representativos).
- Isto faz com que a codificação wavelets seja uma excelente forma de compressão de dados.

matlab

- wave menu



Compressão com Perdas

The screenshot displays the Wavelet 2-D software interface, which is used for image processing and compression. The main workspace is divided into several panels:

- Original Image:** Shows the input image of a woman's face. The axes are labeled from 0 to 250.
- Synthesized Image:** Shows the reconstructed image after compression. The axes are also labeled from 0 to 250.
- Decomposition at level 2:** A 2x2 grid of sub-images representing the wavelet decomposition. A green box labeled "Image Selection" highlights the top-left sub-image, which is a high-frequency detail component.
- Annotations:** A yellow arrow labeled "dwt" points from the Original Image to the Decomposition grid. A yellow arrow labeled "idwt" points from the Decomposition grid to the Synthesized Image.

The right-hand side of the interface contains control panels:

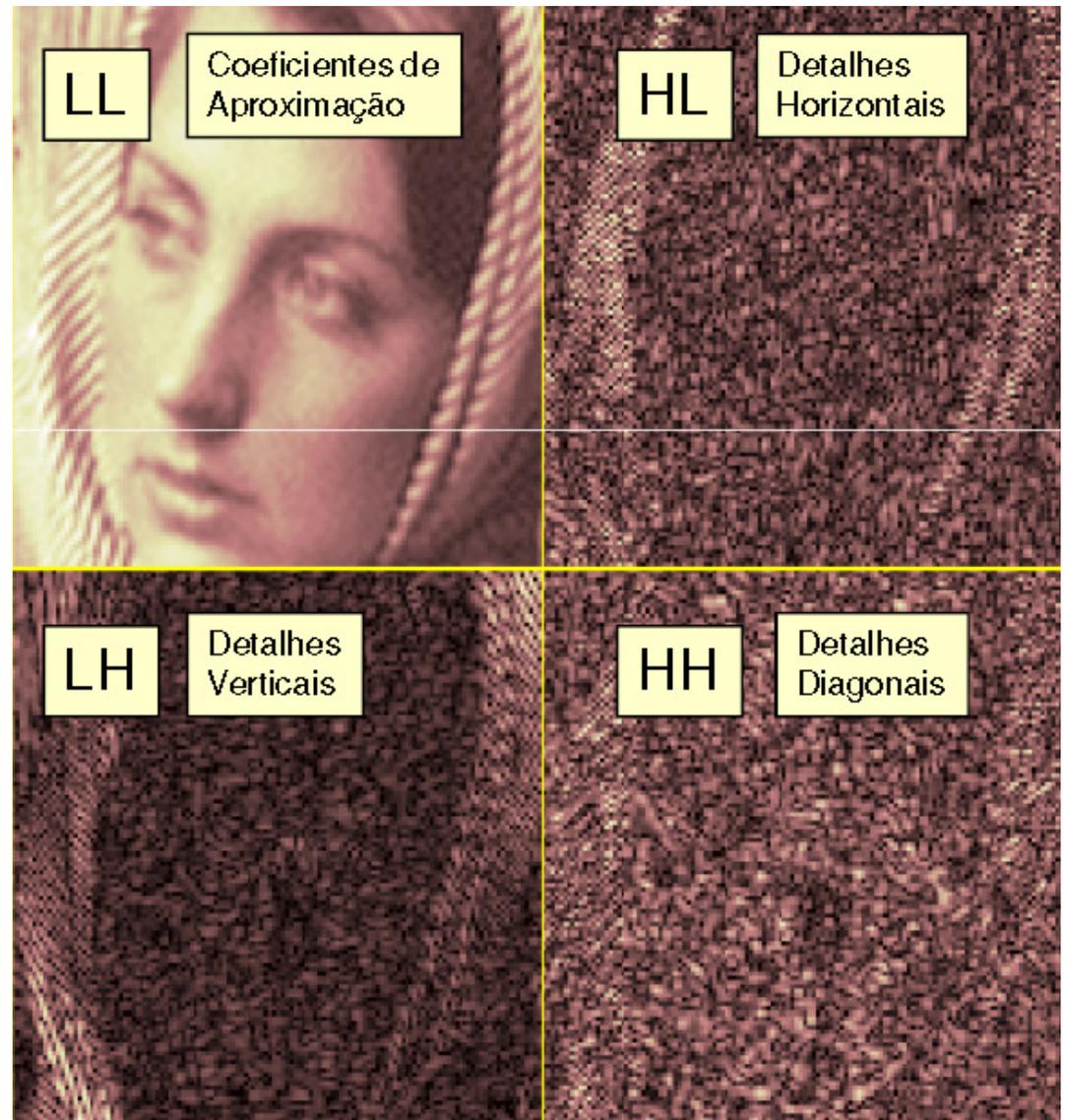
- Data (Size):** woman (256x256)
- Wavelet:** db (dropdown), 4 (dropdown)
- Level:** 2 (dropdown)
- Analyze:** Button
- Statistics:** Button
- Compress:** Button
- Histograms:** Button
- De-noise:** Button
- Decomposition at level:** 2 (dropdown)
- View mode:** Square (dropdown)
- Full Size:** Grid with buttons 1, 2, 3, 4
- Operations on selected image:** Visualize, Full Size, Reconstruct buttons
- Colormap:** pink (dropdown)
- Nb. Colors:** 255 (input field)
- Brightness:** - and + buttons
- Close:** Button

The bottom of the interface features a toolbar with navigation and display controls:

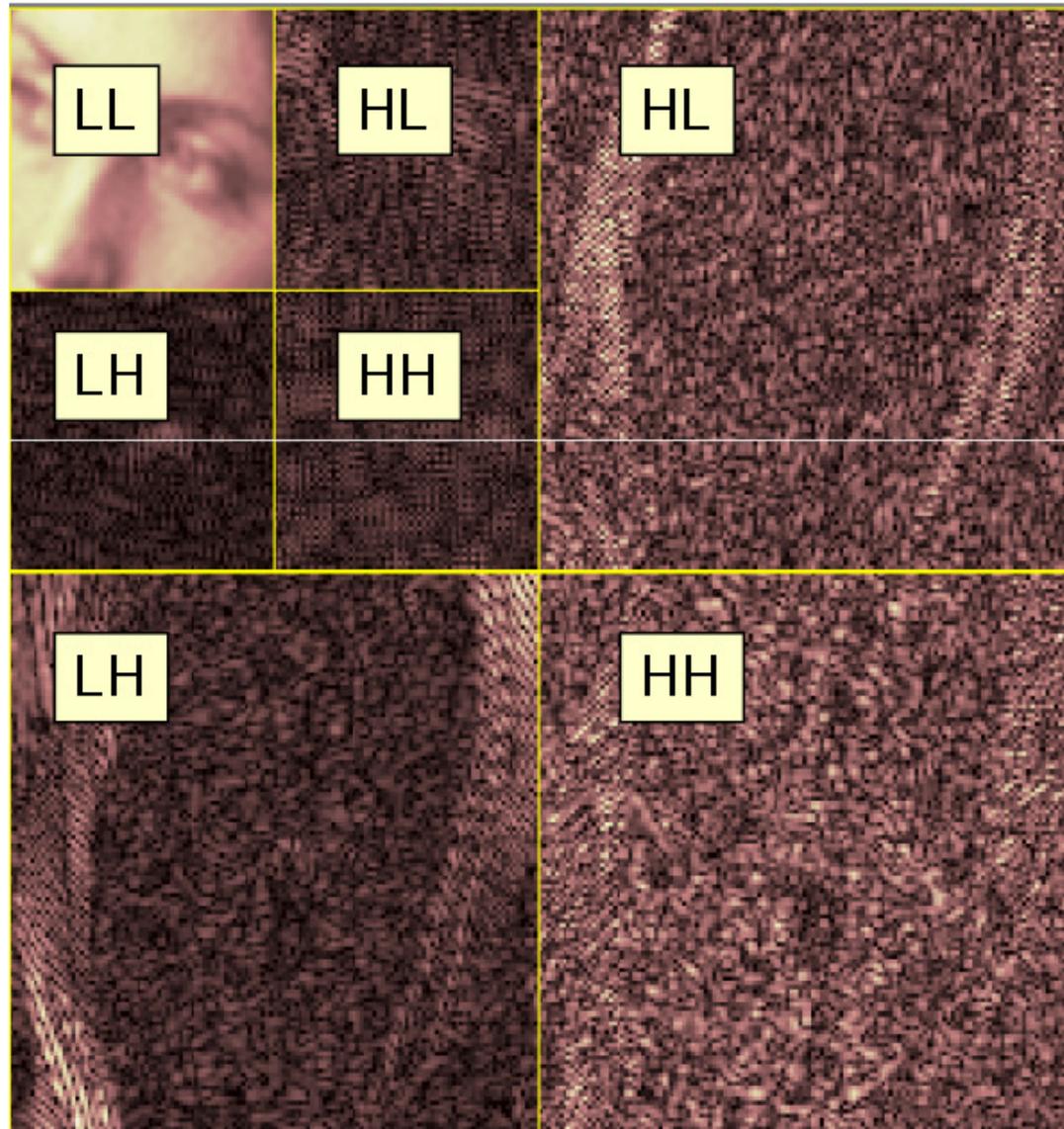
- X+, V+, XY+, X-, Y-, XY-:** Navigation buttons for zooming and panning.
- Center On:** Button with X and Y input fields.
- Info:** Panel showing X = and Y = coordinates.
- History:** Panel with left and right arrow buttons.
- View Axes:** Button.

Decomposição em níveis:

- Nível 1



- Nivel 2



Wavelet Packets

- A transformada wavelet é um subconjunto de uma transformação muito mais versátil, chamada Wavelet packets (pacote de wavelet) .
- Wavelet packets são combinações lineares de Wavelet.
- Eles formam bases que mantêm muitas das propriedades das wavelets-mãe como ortogonalidade, suavidade e localização.
- Os coeficientes das combinações lineares calculados por um algoritmo recursivo calculado por uma estrutura de árvore.

Wavelets pode ajudar a resolver o problema de ruído?

- Em diversos campos do uso de sinais, os cientistas se deparam com o problema de recuperação de dados ruidosos.
- AW , através de uma técnica chamada de *wavelet shrinkage* ou *thresholding* (encolhimento ou limiarização) , proposta por David Donoho pode resolver sim!!

Decomposition & Denoising

The image noise is a stochastic Gaussian Distributed function.



LL	HL	HL
LH	HH	
LH		HH

A técnica funciona da seguinte maneira.

Quando se decompõem dados usando wavelets, usa-se filtros que agem como filtros de média e detalhes.

Assim alguns dos coeficientes wavelet resultantes correspondem aos detalhes do conjunto de dados.

Se os detalhes são pequenos, eles podem ser omitidos sem afetar substancialmente as principais características do conjunto de dados.

A idéia de limiarização, então, é zerar todos os coeficientes que são menores que um determinado limite.

Estes coeficientes são utilizados zeros na transformação wavelet inversa para reconstruir o conjunto de dados.

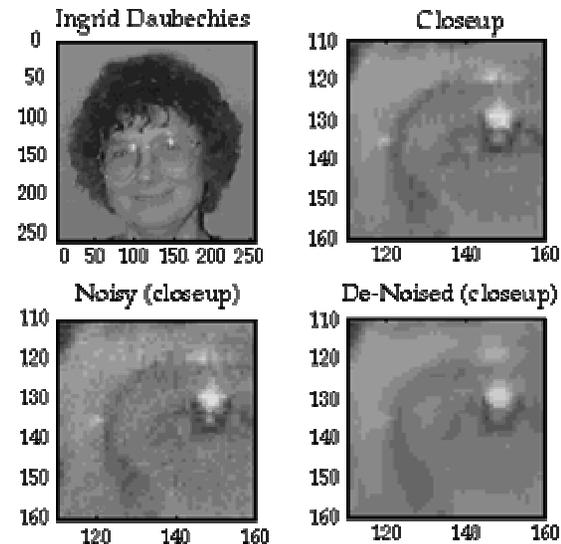
Essa técnica é um passo significativo na melhoria de dados ruidosos, pois o denoising é realizado sem que se perca as estruturas finas.

O resultado é um sinal mais limpo, mas que ainda mostra detalhes importantes.

Imagem de Ingrid Daubechies (de 1993) e closes em seus olhos: original, com adição de ruído e com denoise

Donoho denoise:

- a imagem é transformada para o domínio de wavelet (usando Coiflets-3)
- Aplica-se um limiar (threshold) e
- Faz-se a transformada inversa.



Wavelet denoising

- Identify **low** and **high** energy coefficients
- Modify noisy coefficients by **adaptive thresholding**
- We use the **optimal adaptive threshold [1-6]:**

$$T = \sigma_n^2 / \sigma$$

σ_n^2 = Noise variance

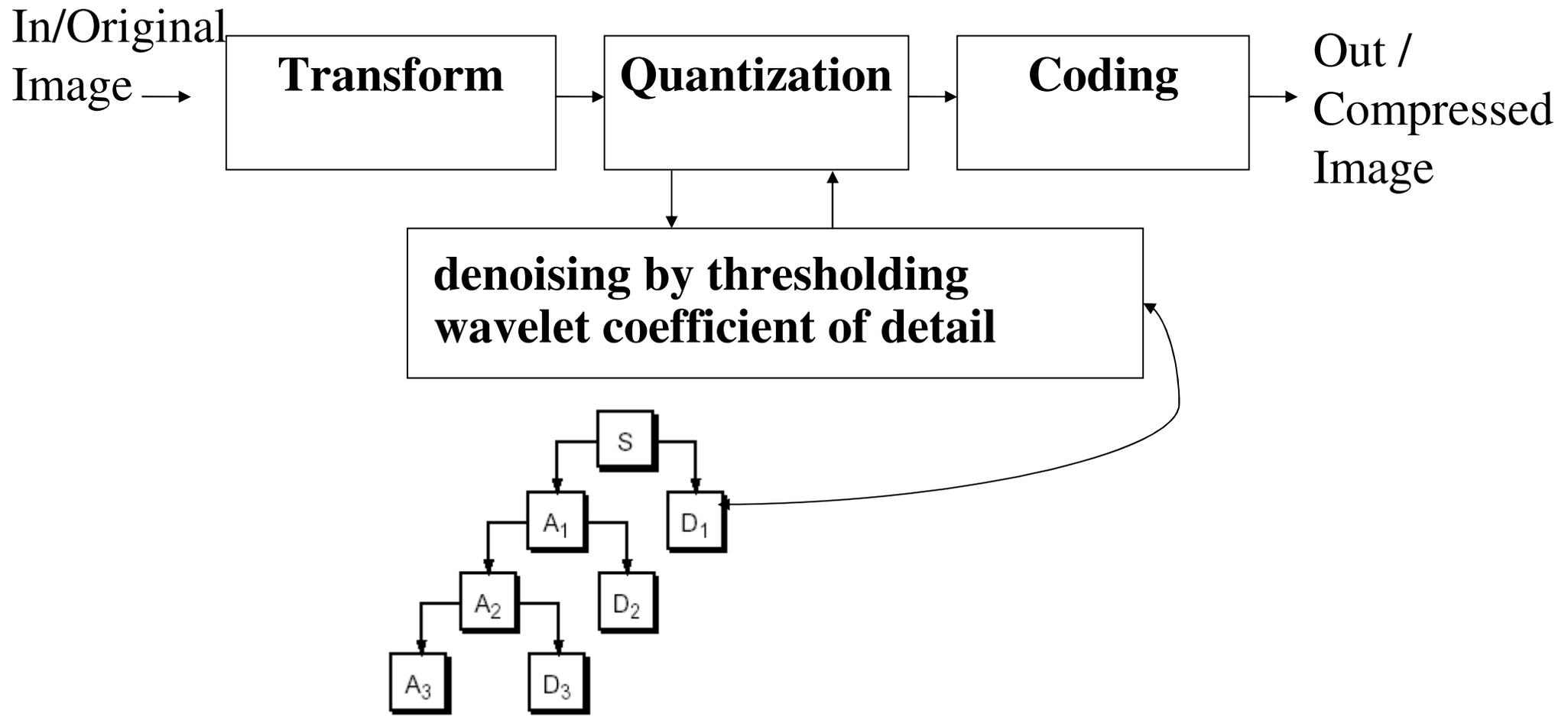
σ = Original Signal variance

(this is a Hard Thresholding approach)

Comparing the performance

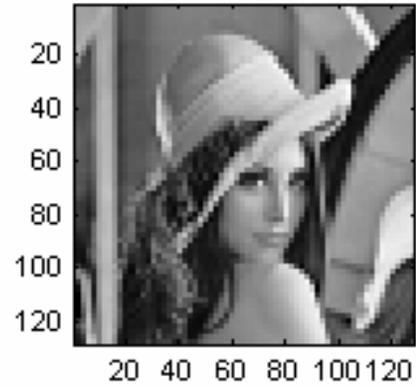
- Compression ratio
- Quality:
 - Root Mean Square Error (RMSE),
 - Sign Noise Ratio (SNR) and
 - Peak Sign Noise Ratio (PSNR)

Denoising Aspects

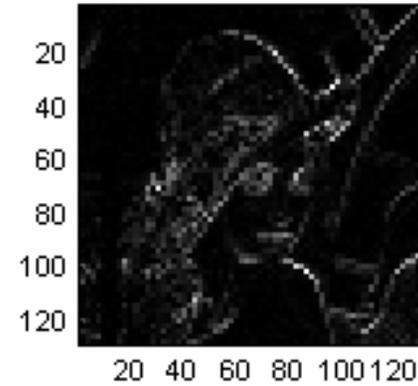


coefficient of detail

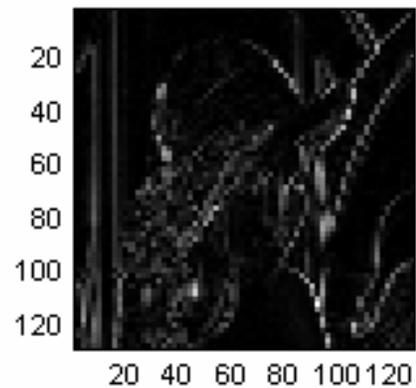
Aproximação A1



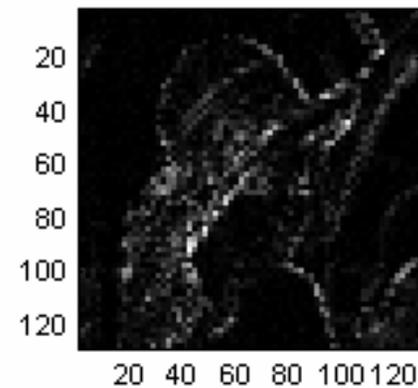
Detalhes Horizontal H1



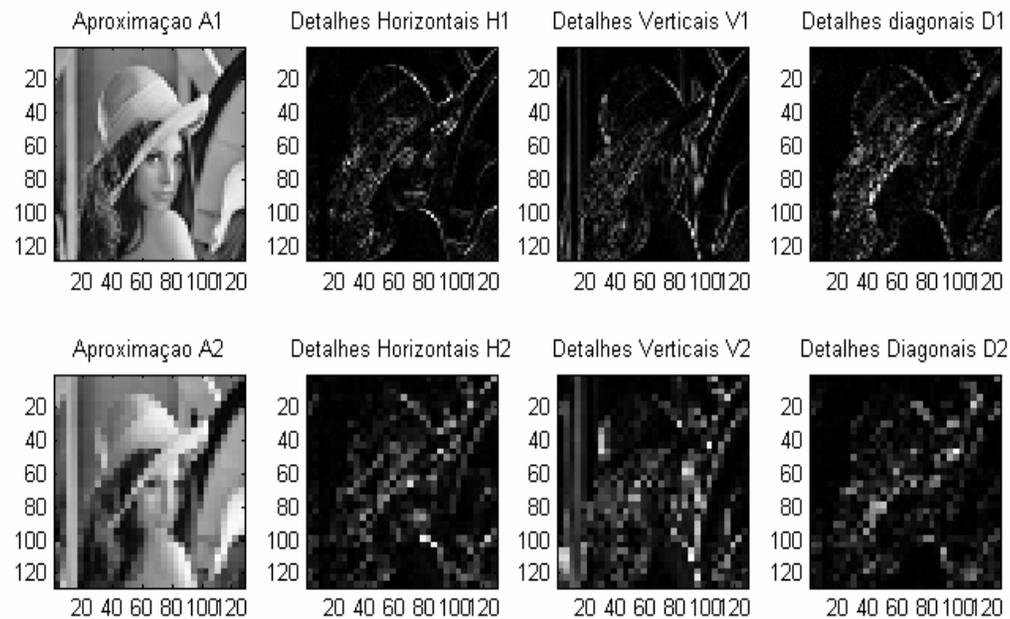
Detalhes Verticais V1



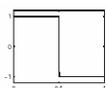
Detalhes Diagonais D1



Smooth regions: coefficients dominated by noise

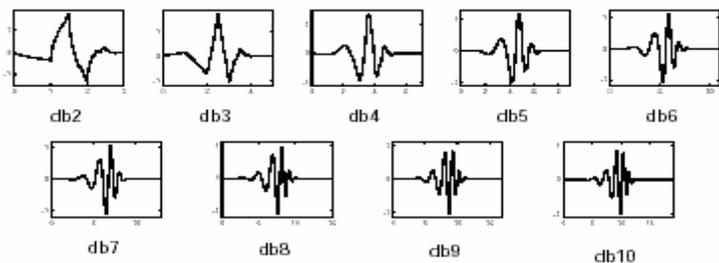


wavelet families : 36 variations

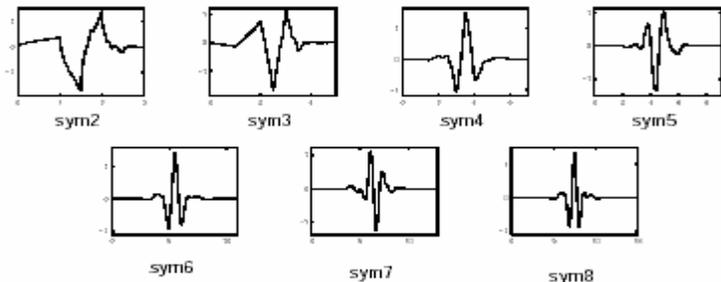


Haar

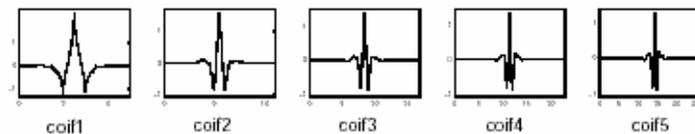
Daubechies



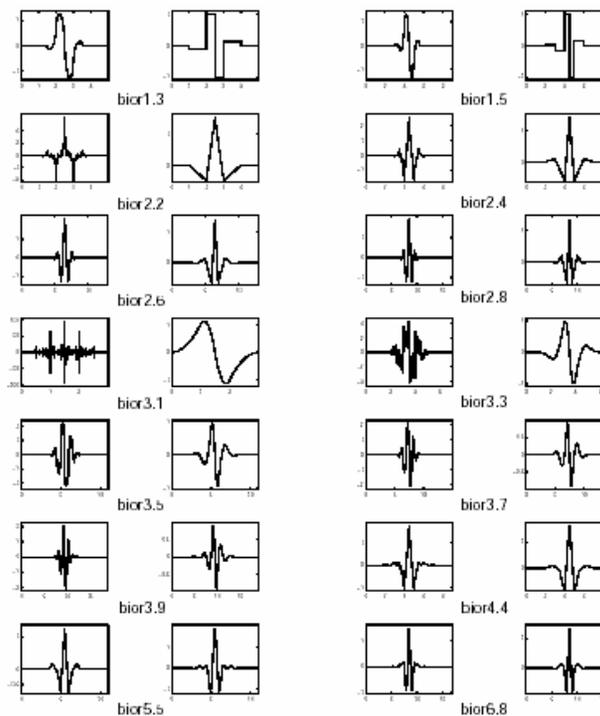
Symlets



Coiflets



Biortogonal



Tested Images



Lena



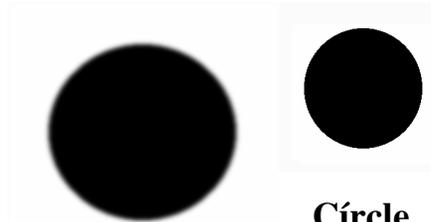
Cameramen



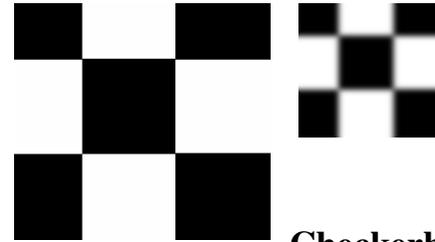
Goldhill



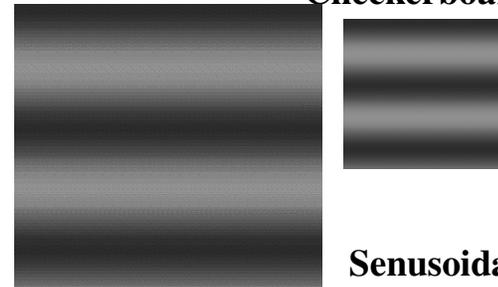
Peppers



Circle



Checkerboard



Sinusoidal

Acrobat 3.0 is the fastest way to publish any document.
 Take the PDF file you're looking at in Adobe PageMaker® software and use Acrobat 3.0 software to make it optimized for the Web. Then, use Acrobat 3.0 software to make it optimized for the Web, making it available on any platform—Macintosh, Windows, or Linux.

Text

3 resolution:
 28x128,
 256x256
 and
 512x512

3 degradation levels
 Additive
 White
 Gaussian
 Noise
 (AWGN):

☞ = 5,
 ☞ = 10,
 and
 ☞ = 20

The tested 3456 cases, presented on 96 tables or 288 graphs

Evaluation criteria:

$$RMSE = \sqrt{\left[\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [G(x, y) - F(x, y)]^2 \right]} \quad (1)$$

$$SNR_{ms} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} G(x, y)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [G(x, y) - F(x, y)]^2} \quad (2)$$

$$SNR_{rms} = \sqrt{SNR_{ms}} \quad (3)$$

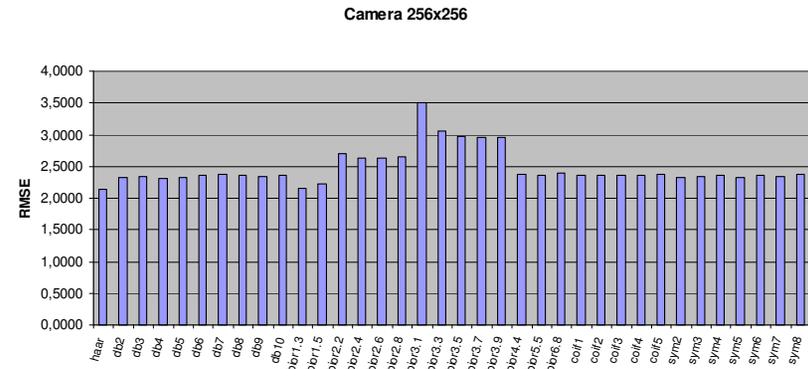
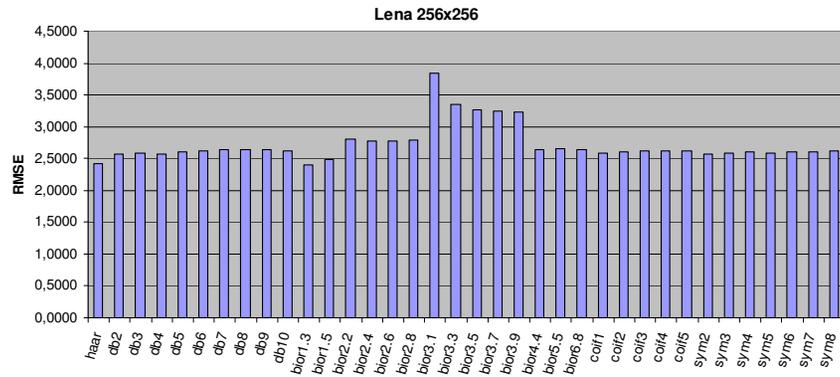
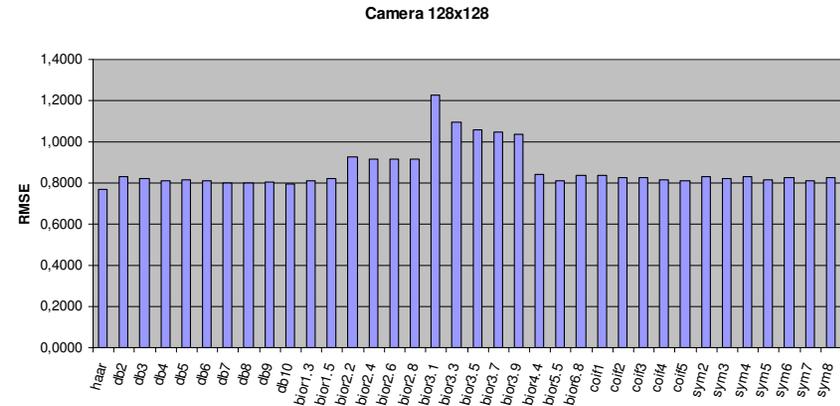
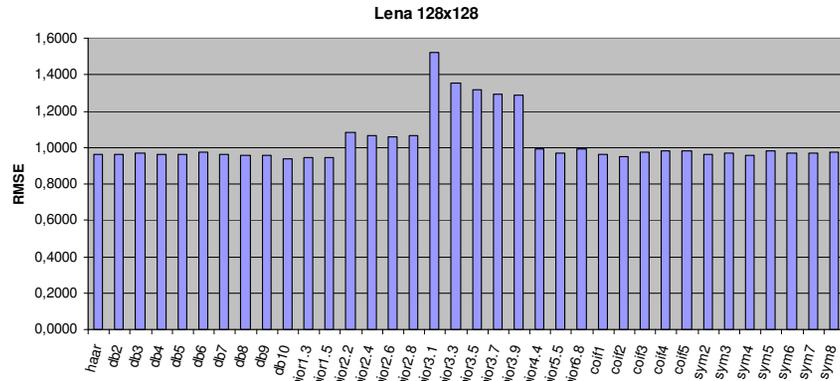
$$PSNR = 20 \log_{10} \left(\frac{2^n - 1}{RMSE} \right) \quad (4)$$

WaveLab

- é um biblioteca de rotinas para analise wavelets para Matlab desenvolvida em Stanford.
- Se usar o Matlab, copie os arquivos do diretório **~wavelet/matlab** para o seu diretório /matlab e divirta-se !
- A Universidade Rice, nos EUA, também desenvolveu um pacote para implementar analise e projeto de bancos de filtros em aplicações 1D e 2D para Matlab 4.1

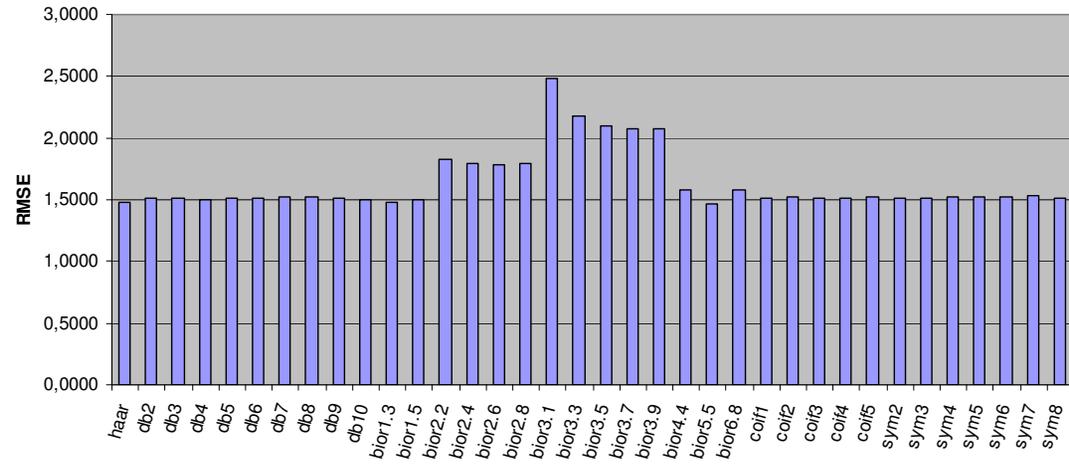
RMSE

Same image and σ in 3 levels

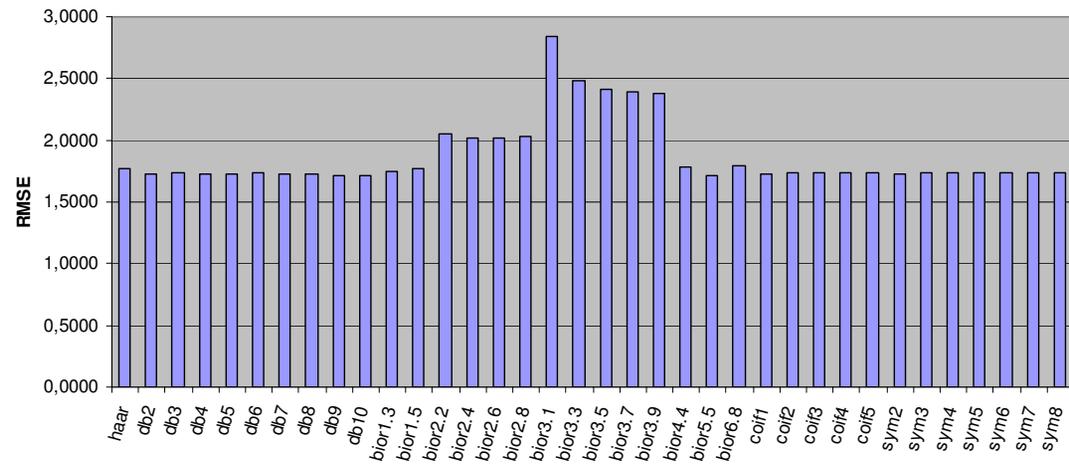


RMSE

Goldhill 128x128

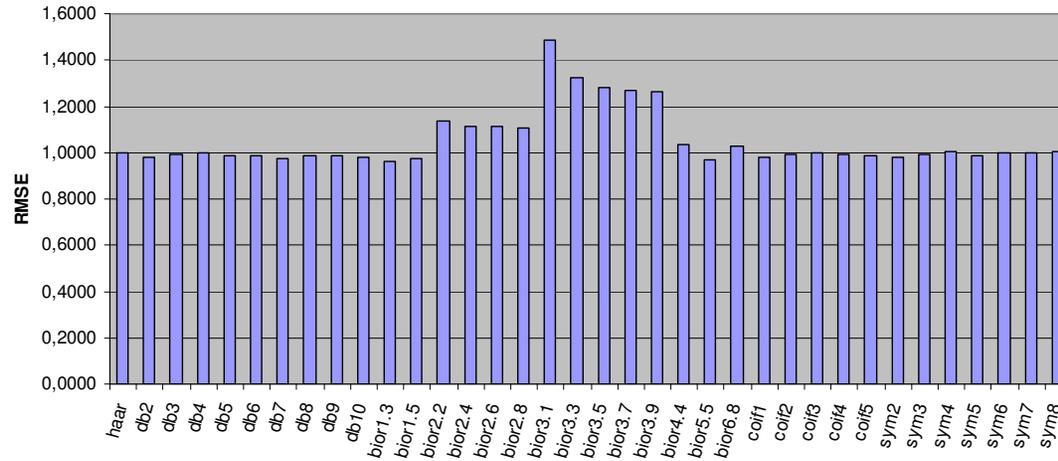


Goldhill 256x256

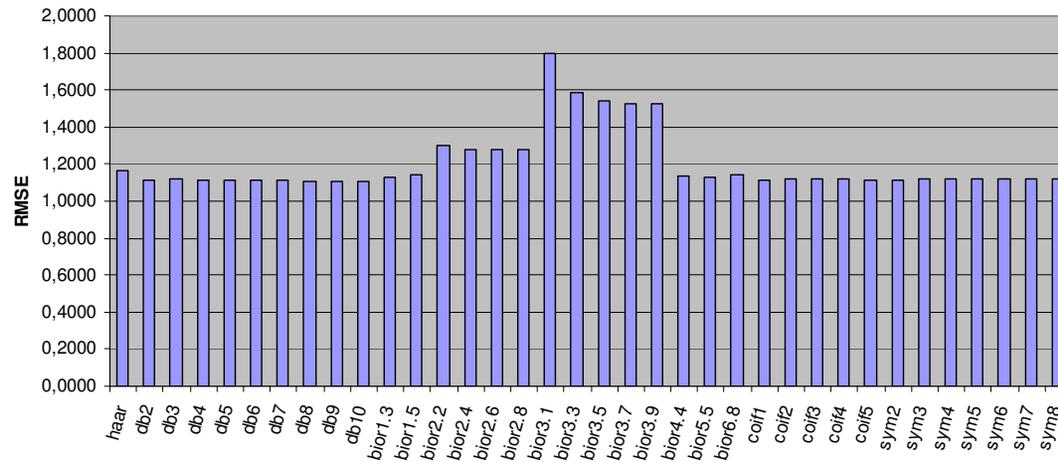


RMSE

Peppers 128x128

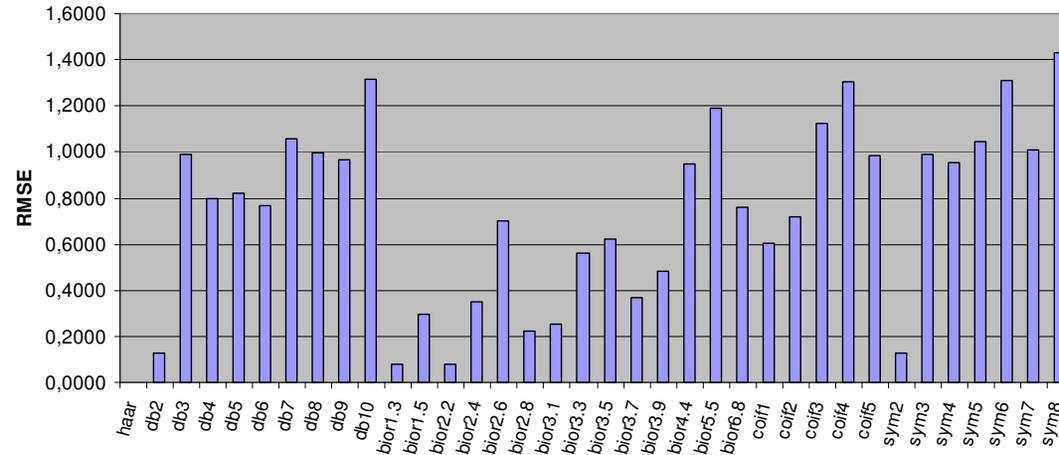


Peppers 512x512

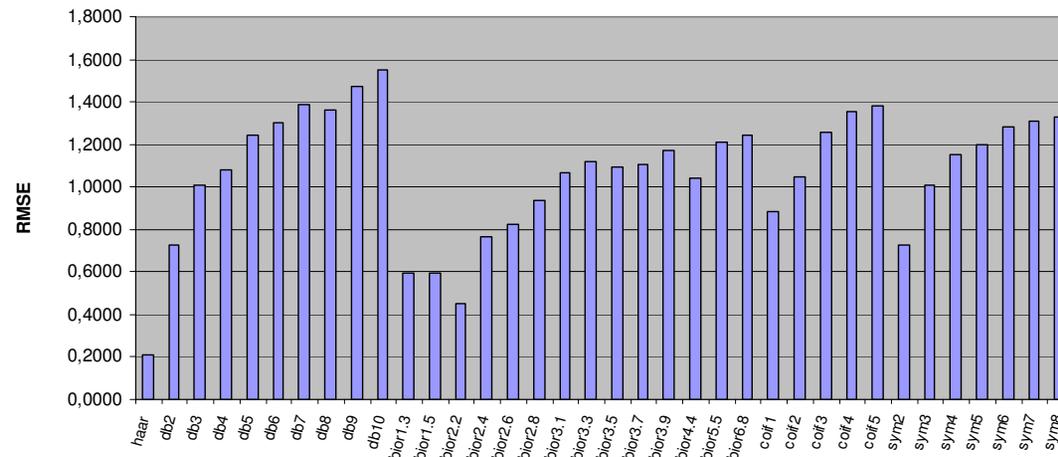


RMSE

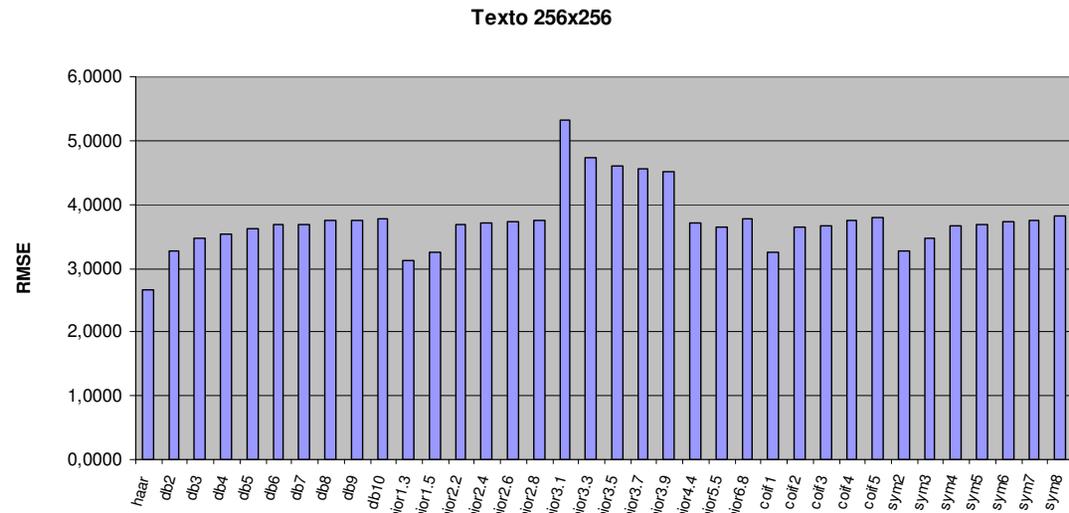
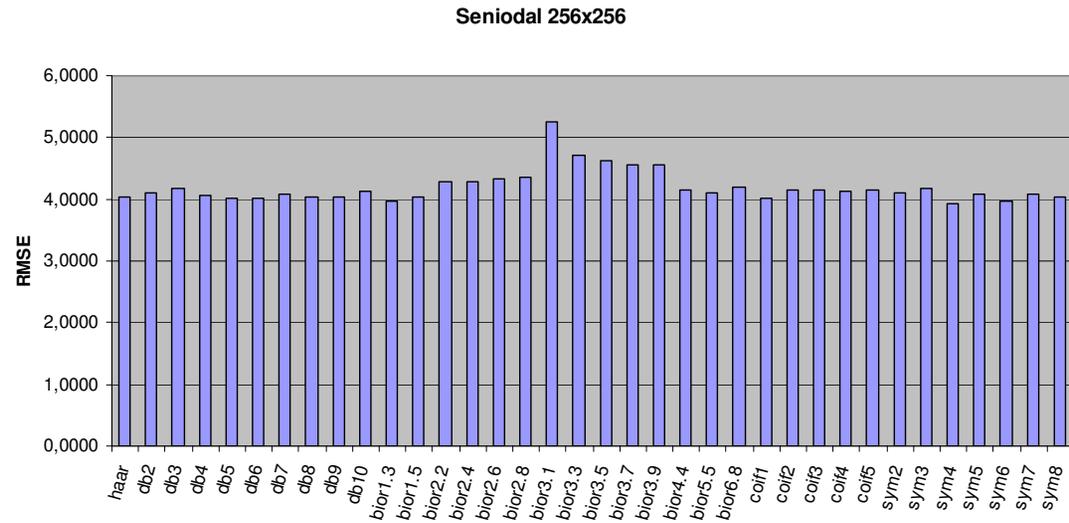
Xadrez 256x256



Círculo 256x256

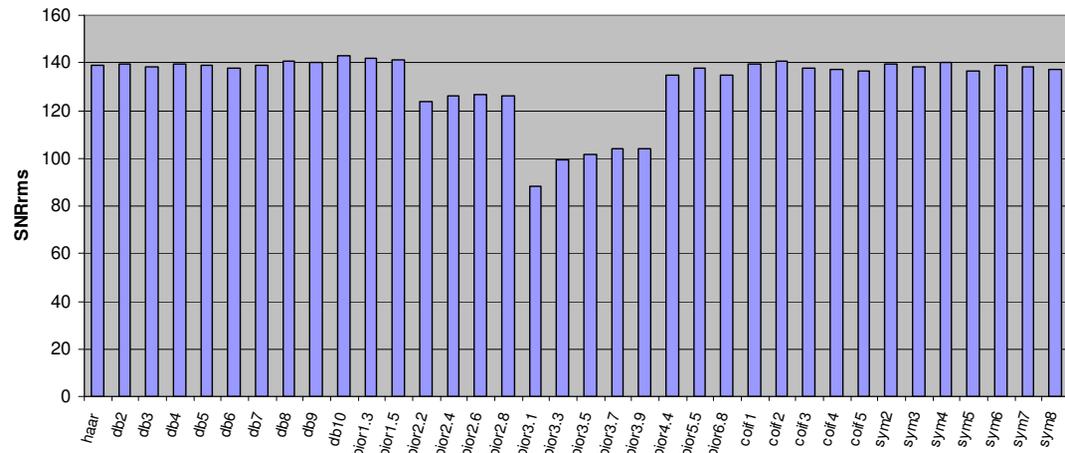


RMSE

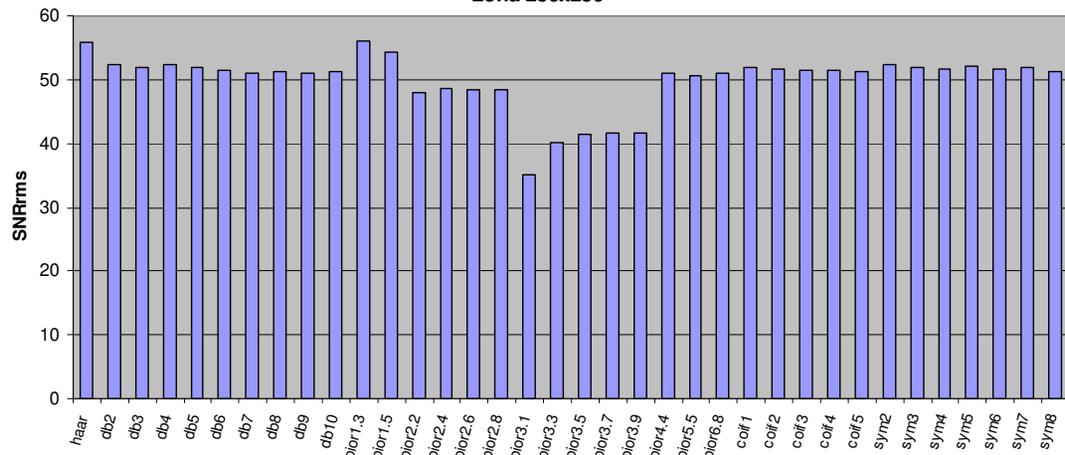


SNR

Lena 128x128

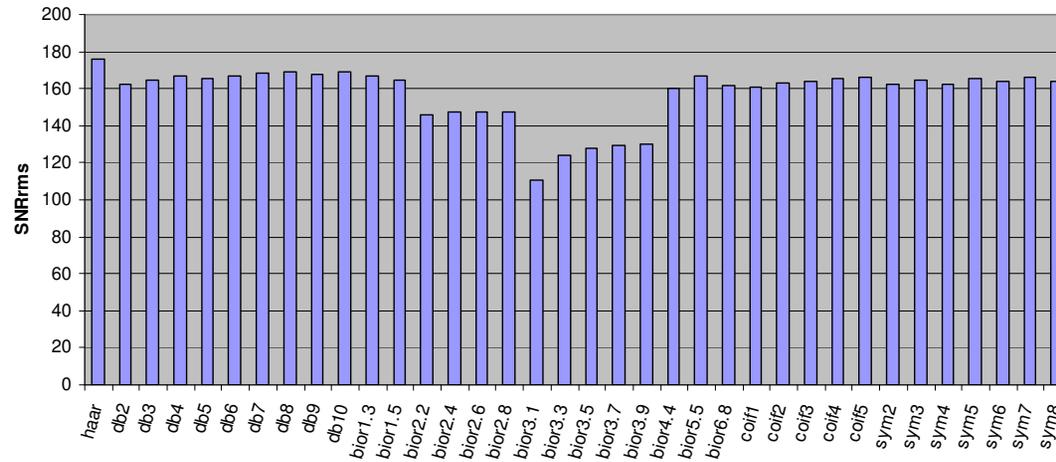


Lena 256x256

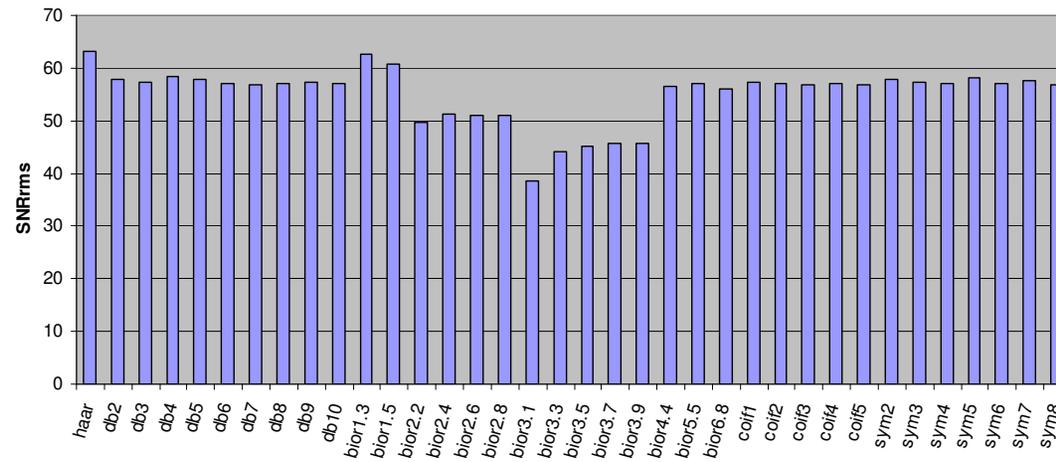


SNR

Camera 128x128

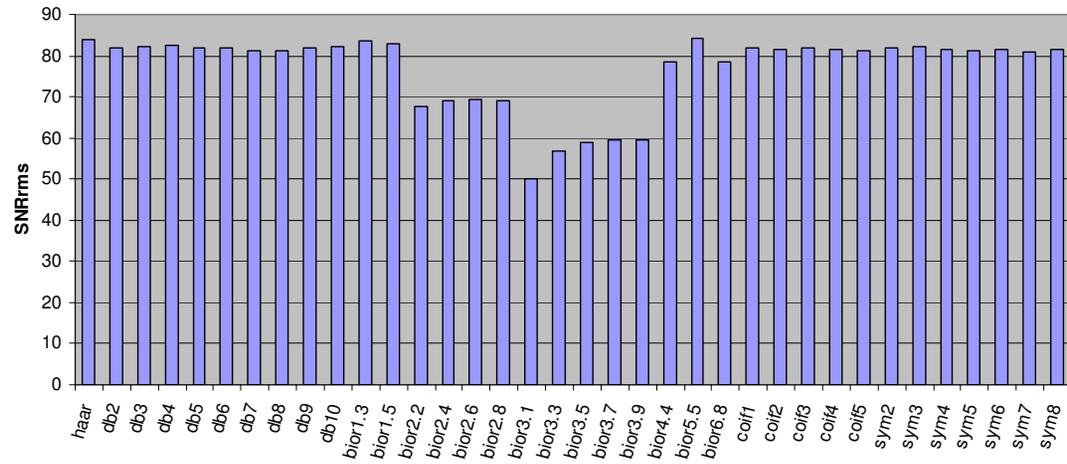


Camera 256x256

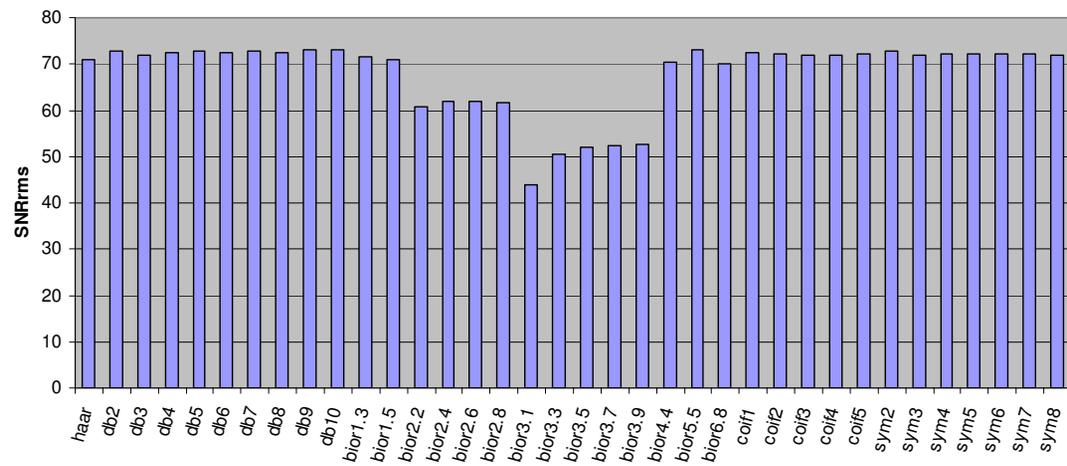


SNR

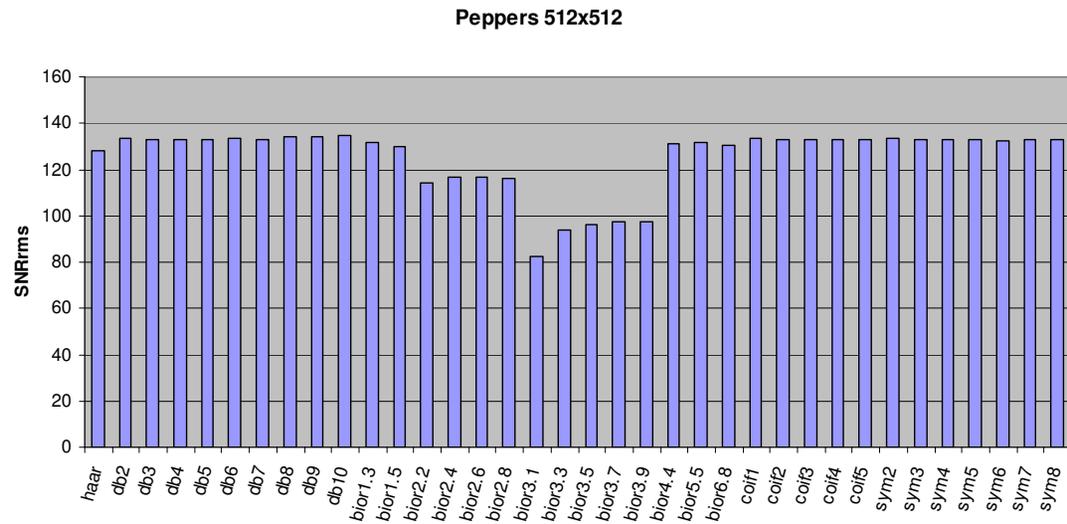
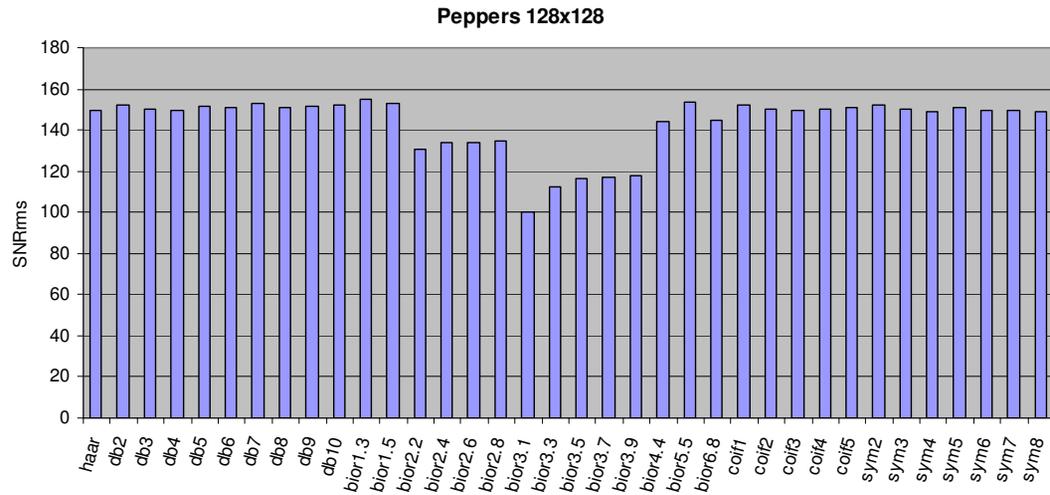
Goldhill 128x128



Goldhill 256x256

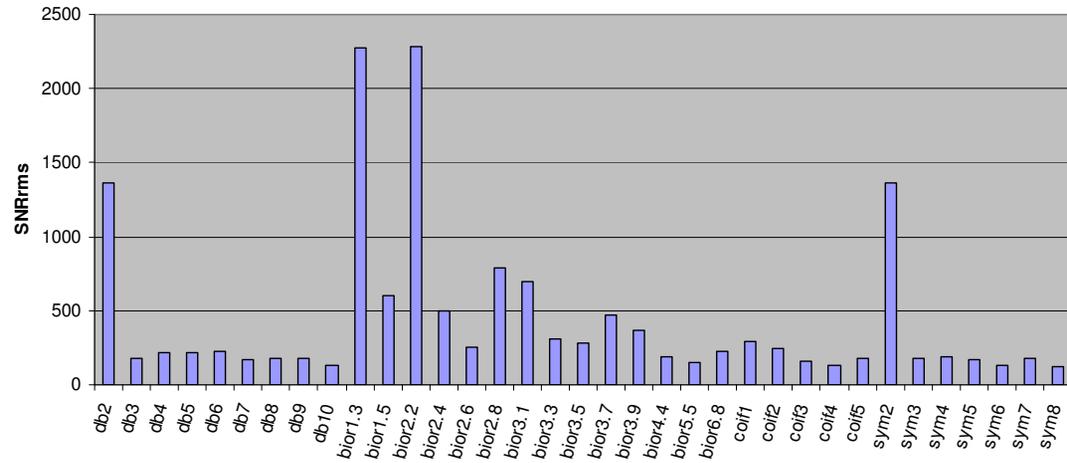


SNR

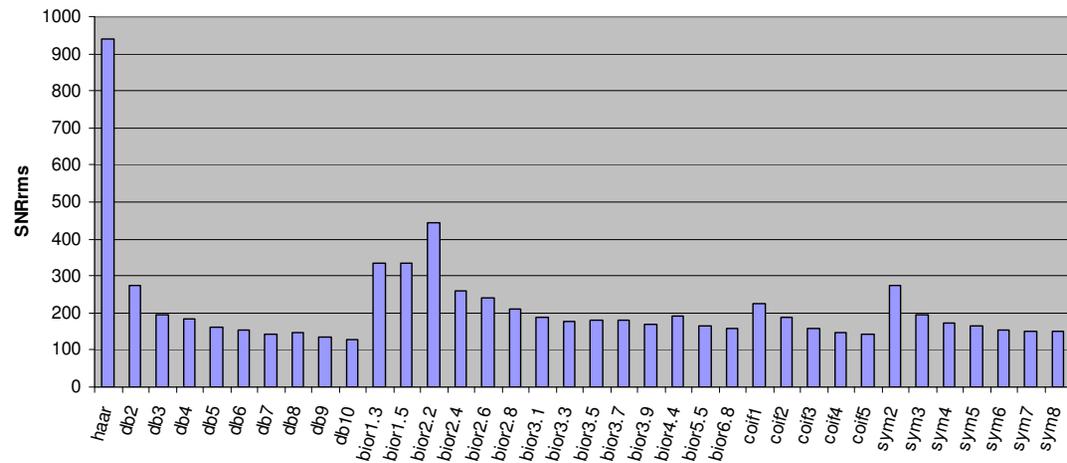


SNR

Xadrez 256x256

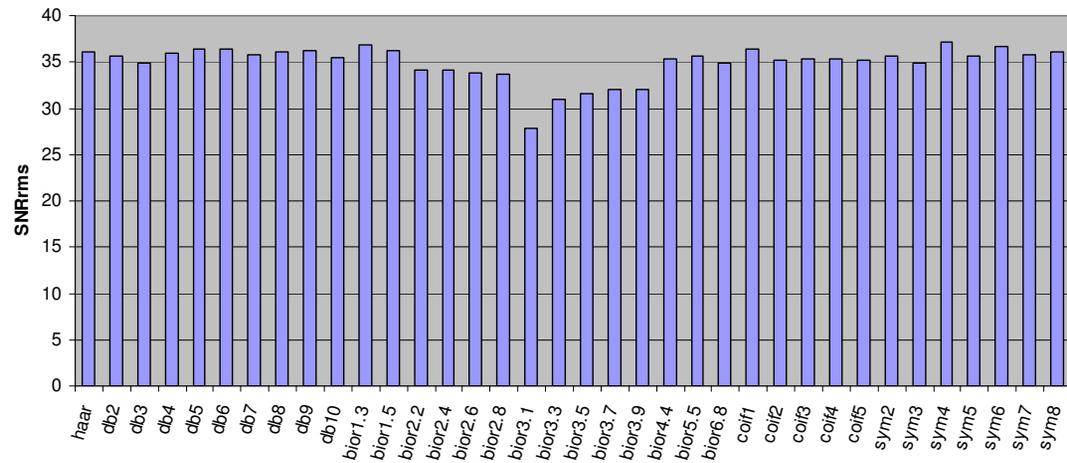


Círculo 256x256

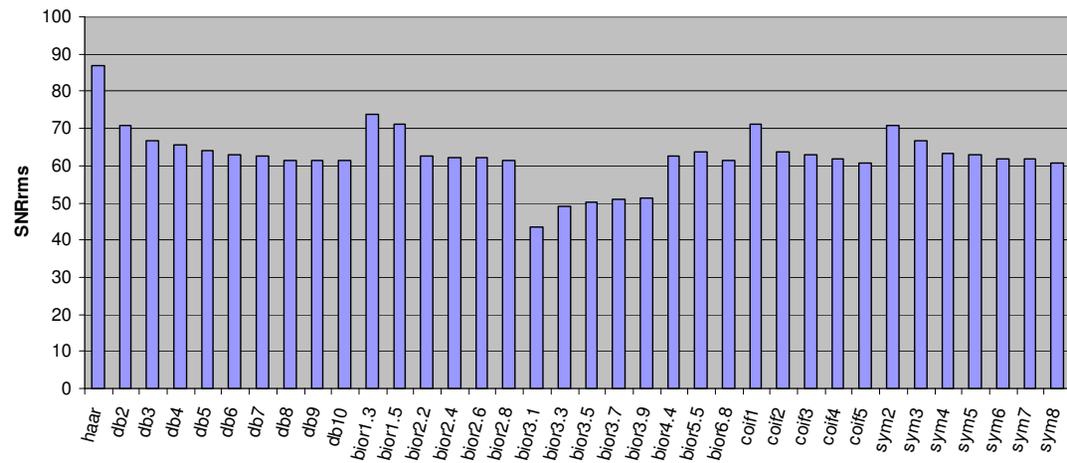


SNR

Seniodal 256x256

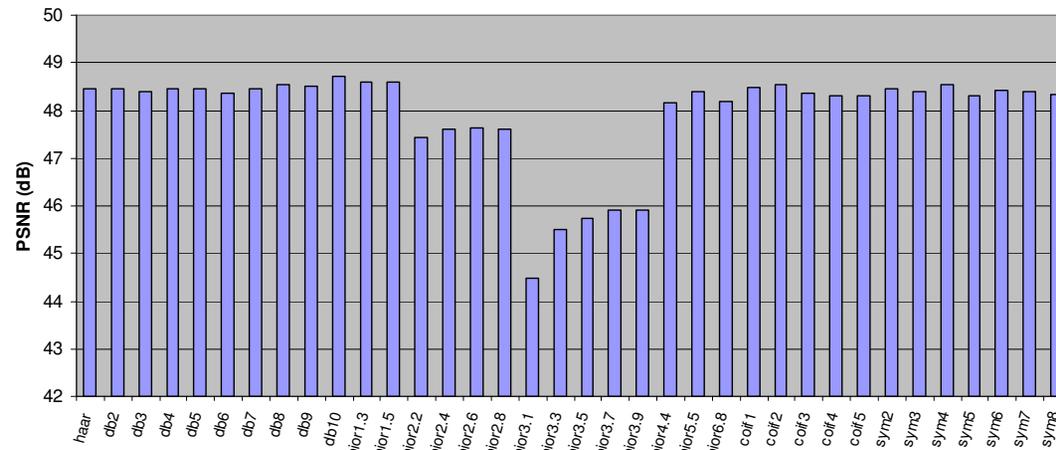


texto 256x256

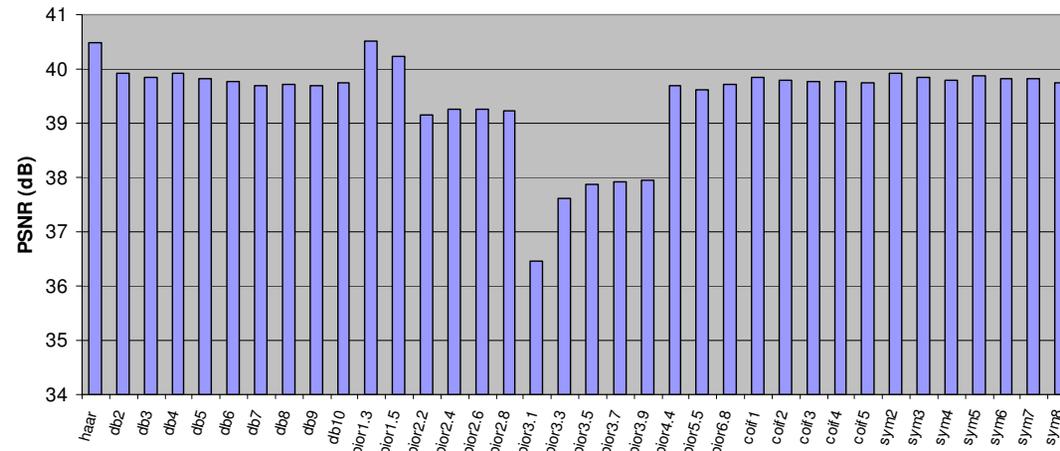


PSNR

Lena 128x128

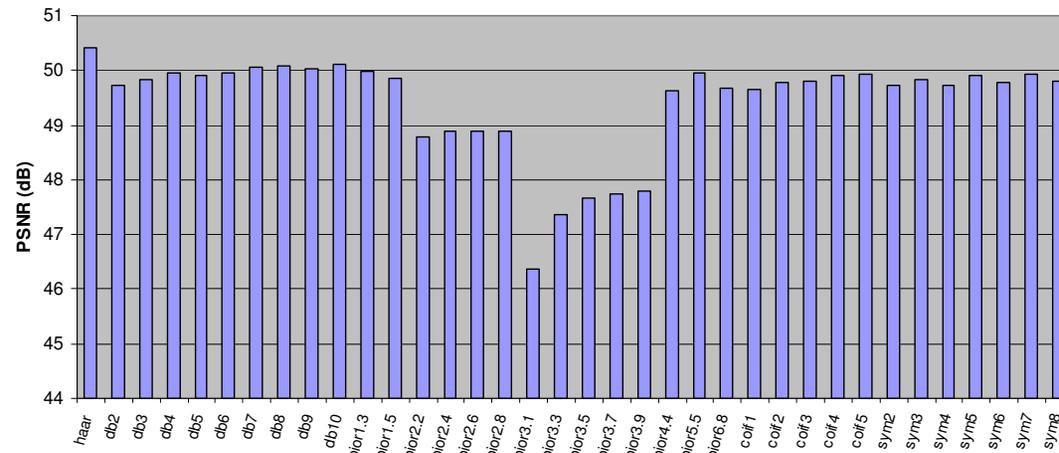


Lena 256X256

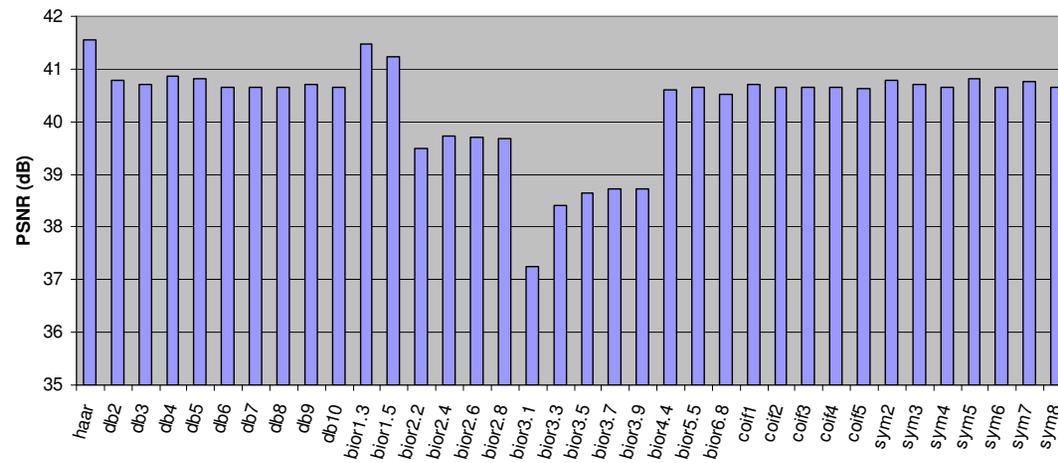


PSNR

Camera 128x128

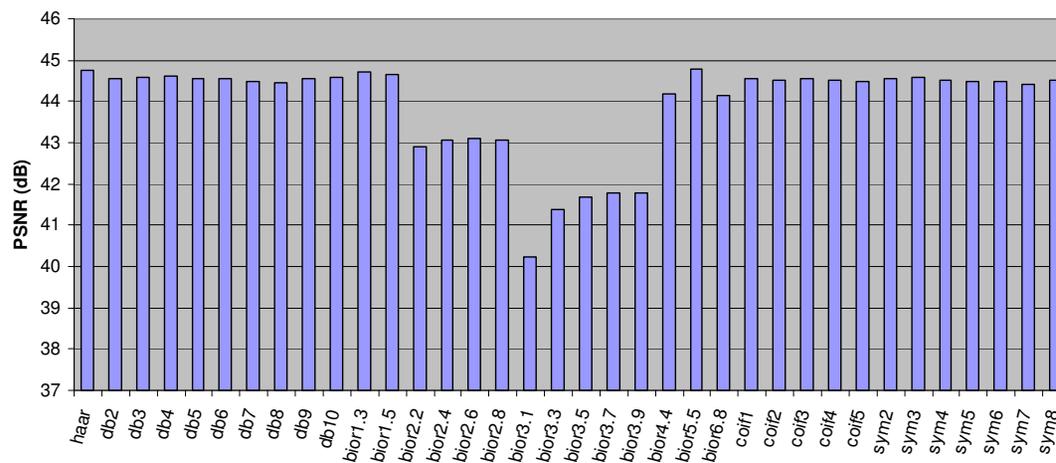


Camera 256x256

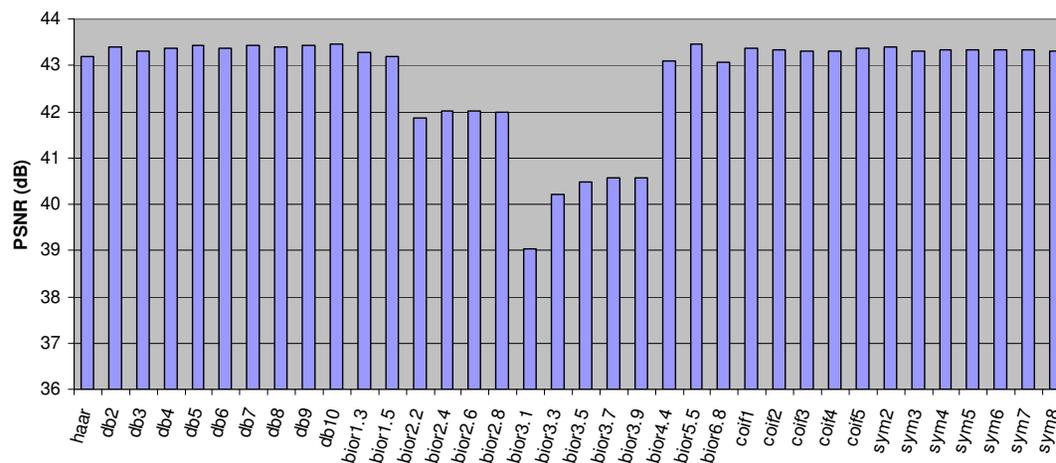


PSNR

Goldhill 128x128

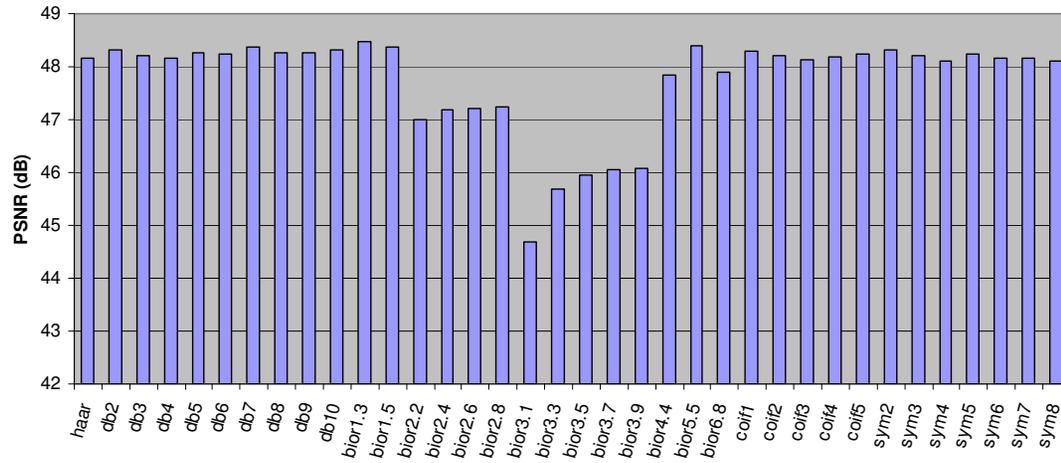


Goldhill 256x256

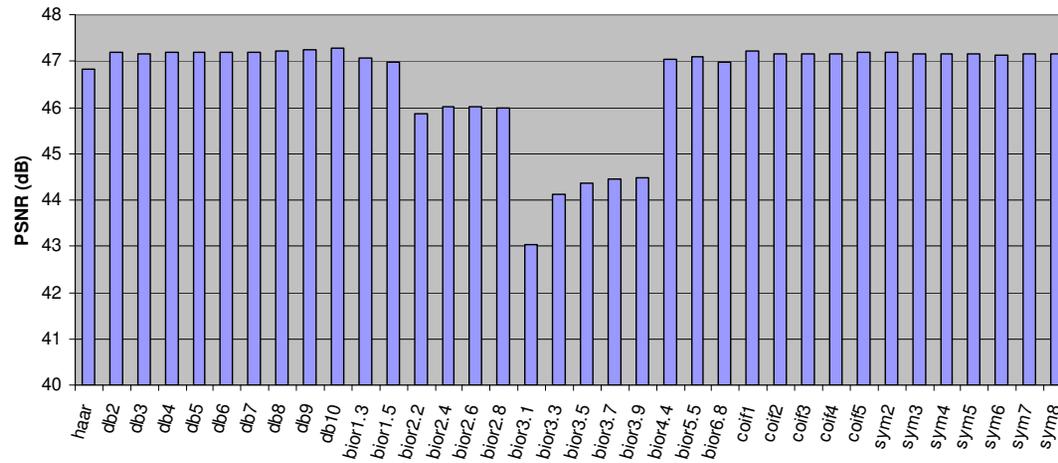


PSNR

Peppers 128x128

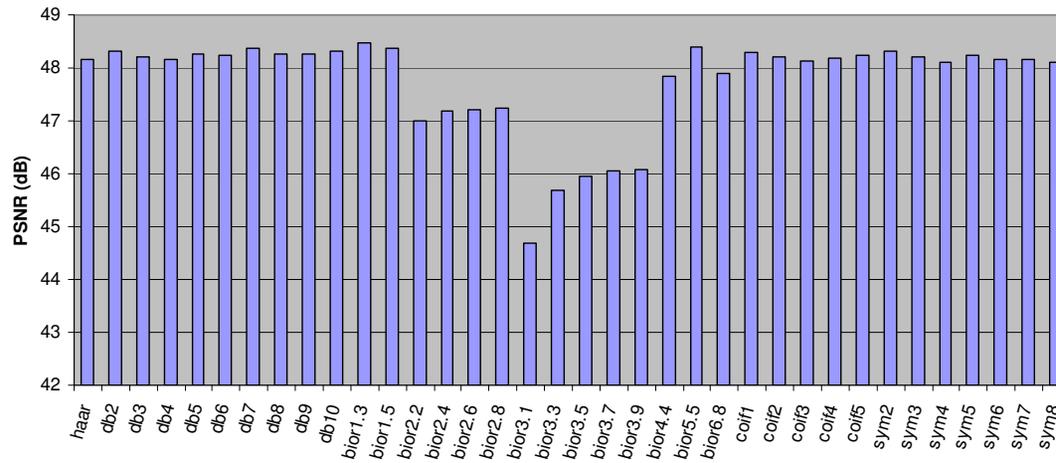


Peppers 512x512

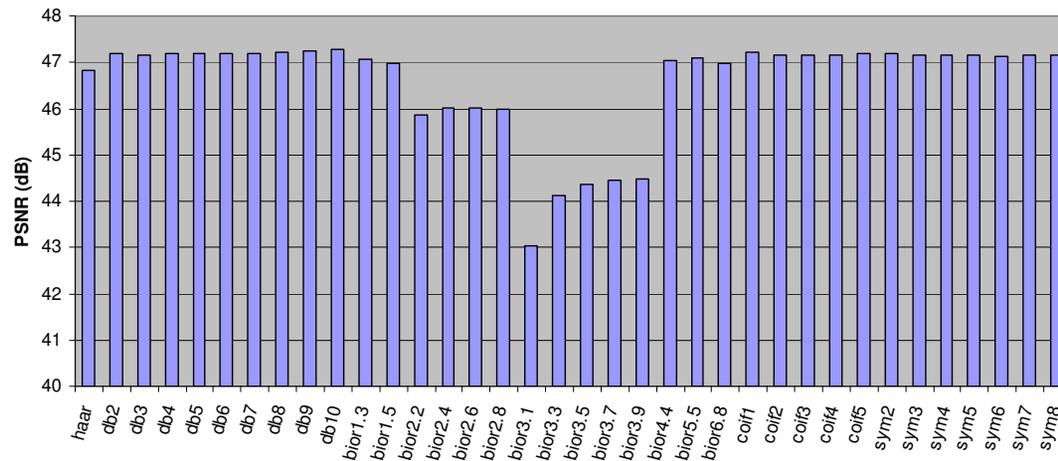


PSNR

Peppers 128x128

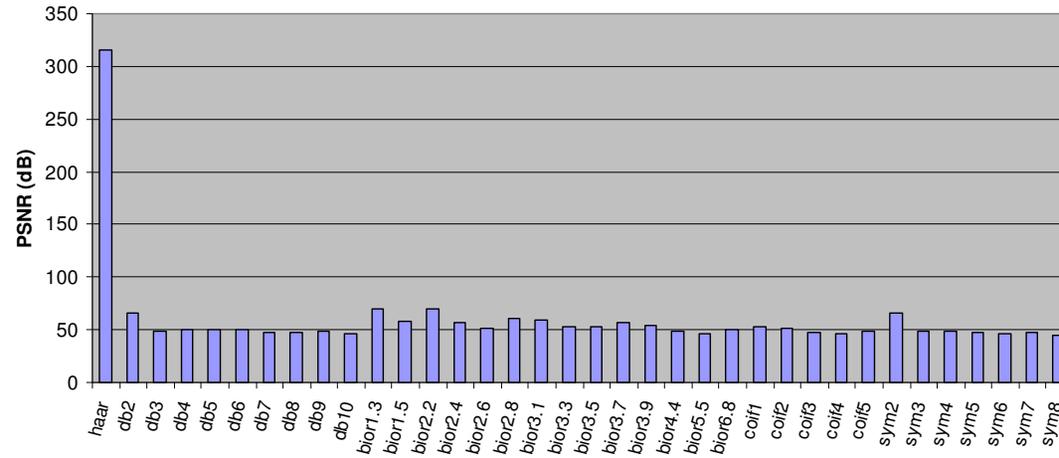


Peppers 512x512

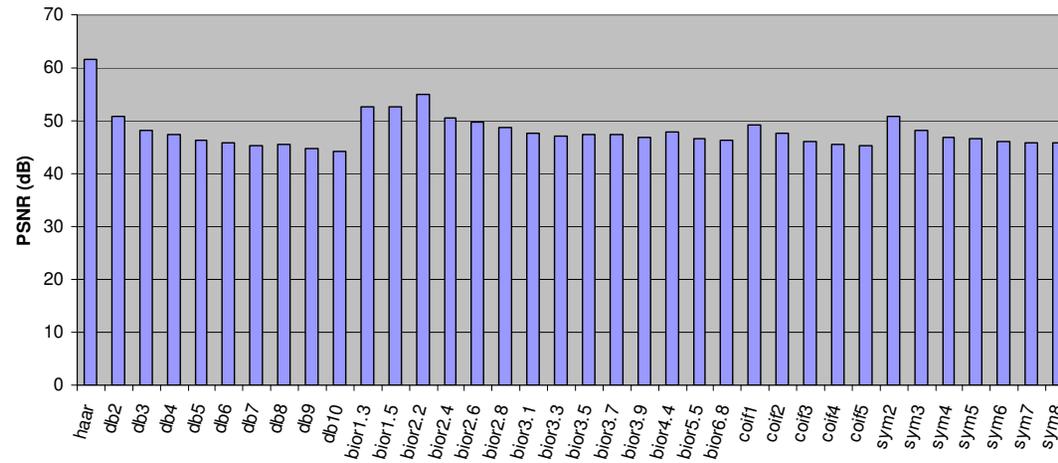


PSNR

Xadrez 256x256

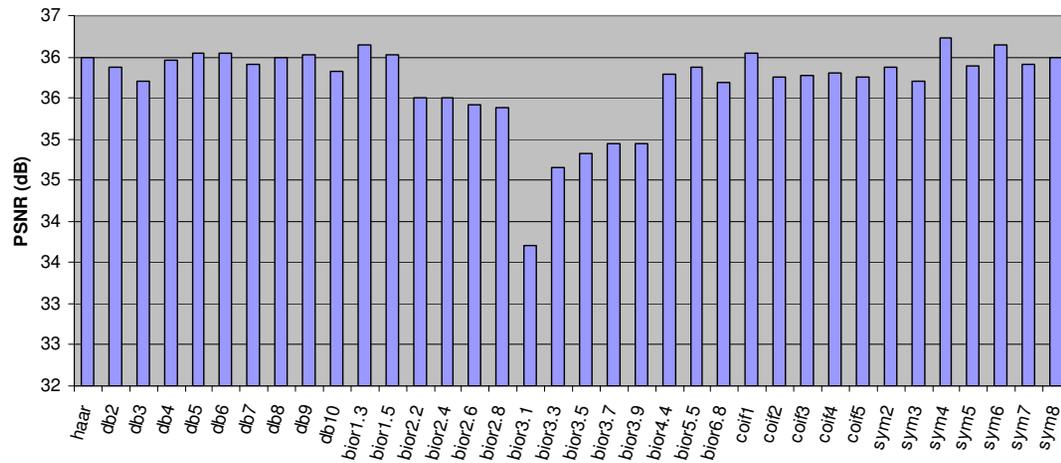


Círculo 256x256

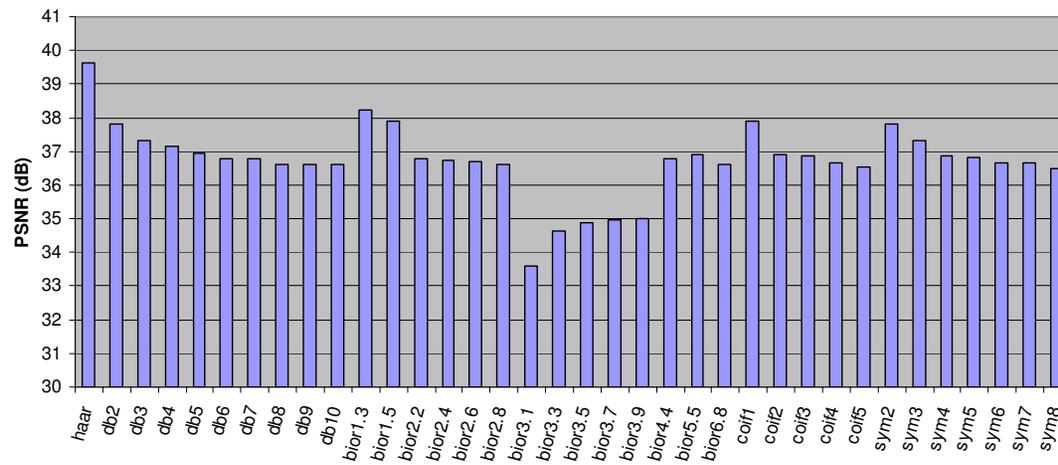


PSNR

Seniodal 256x256

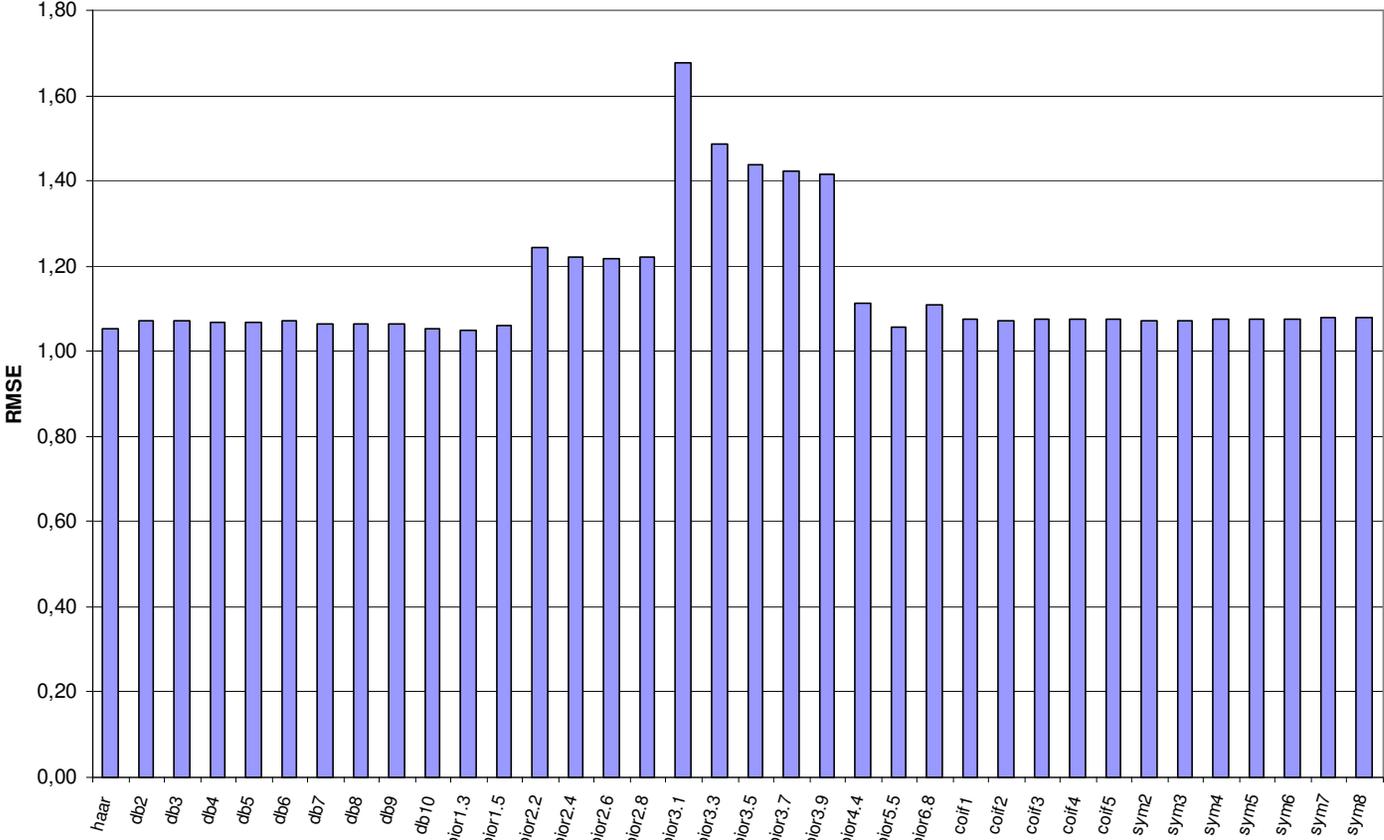


Texto 256x256



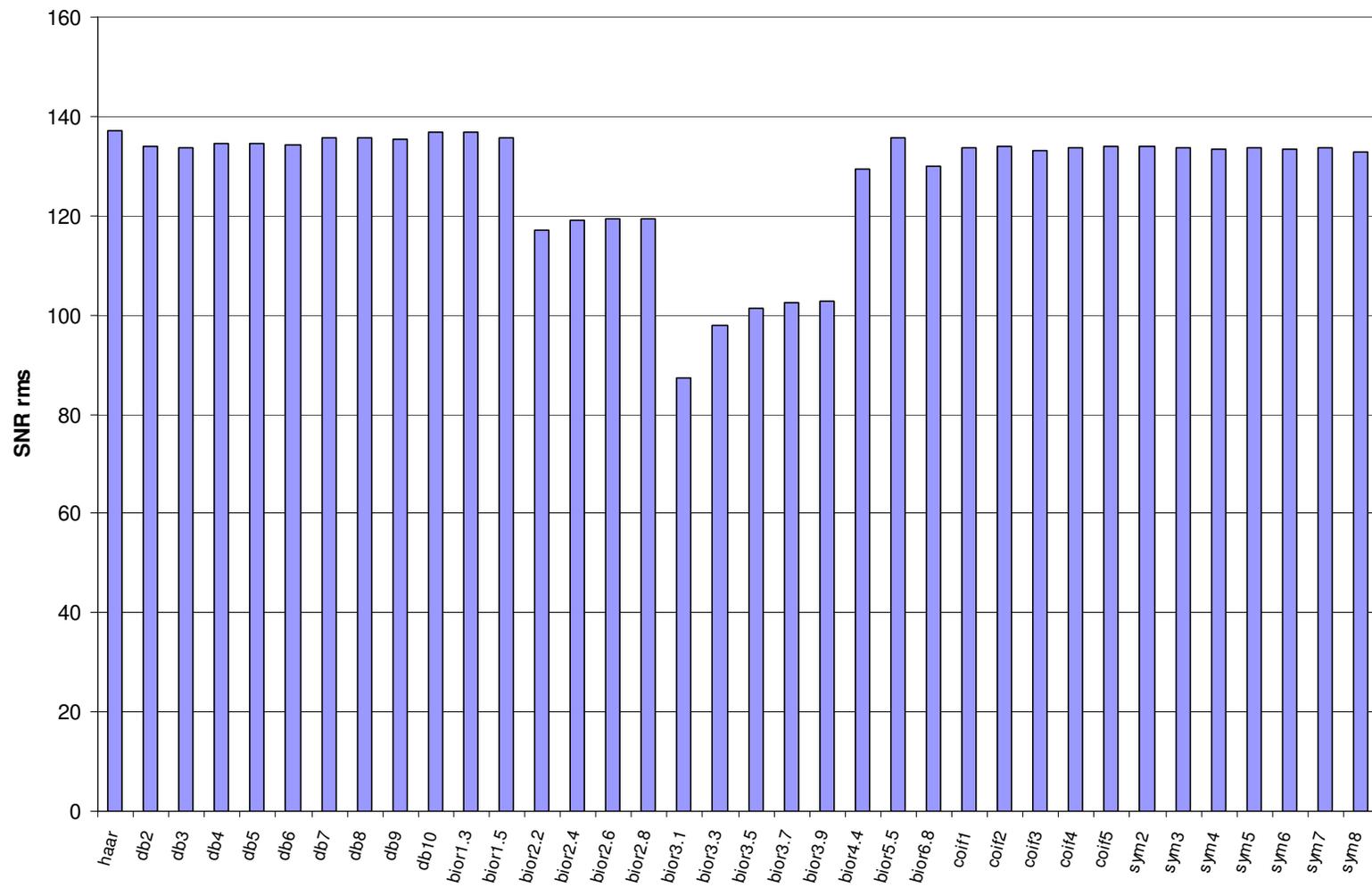
combining results photographic images

RMSE Fotográficas 128x128



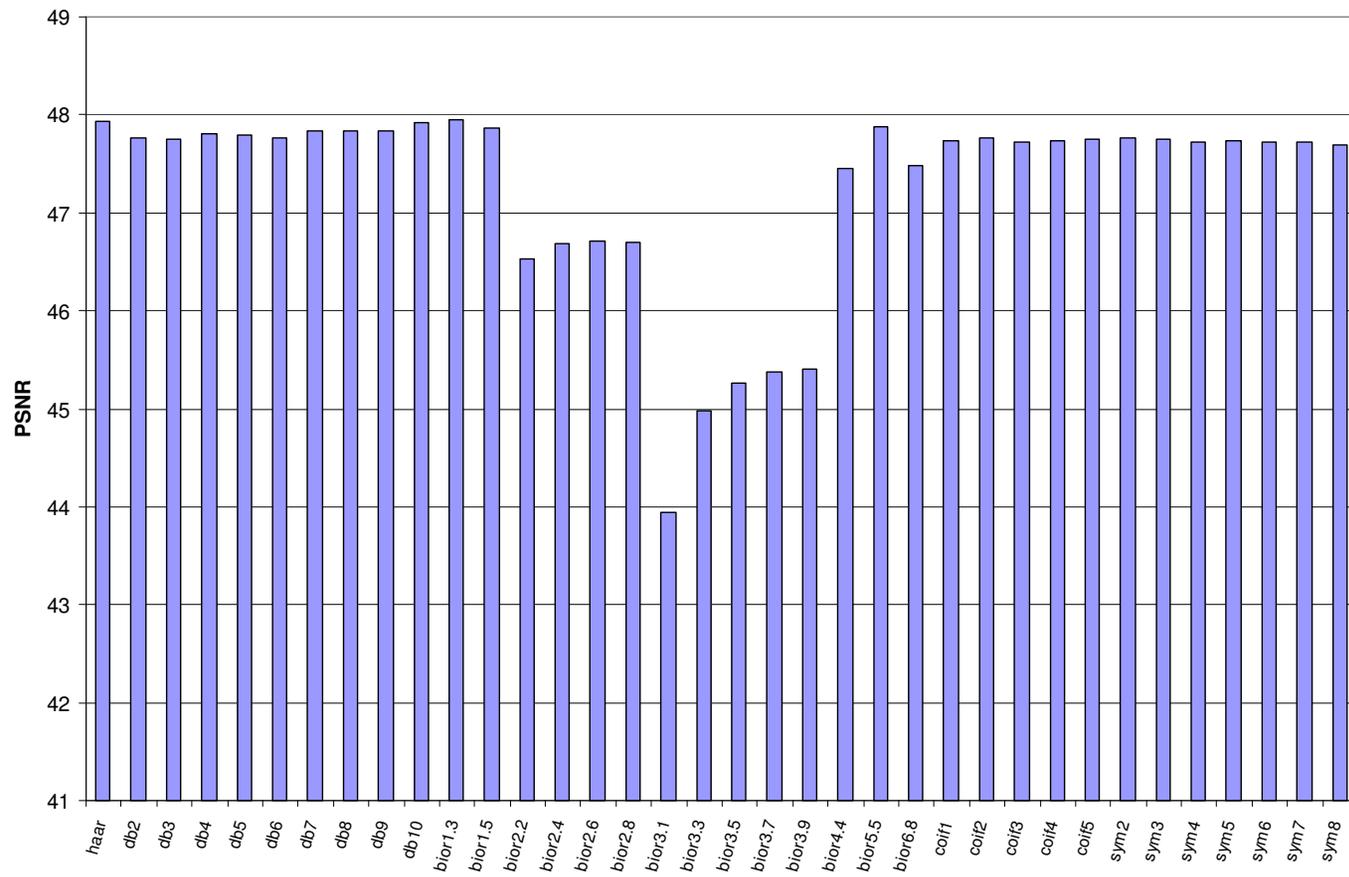
combining results photographic images

Fotográficas 128x128



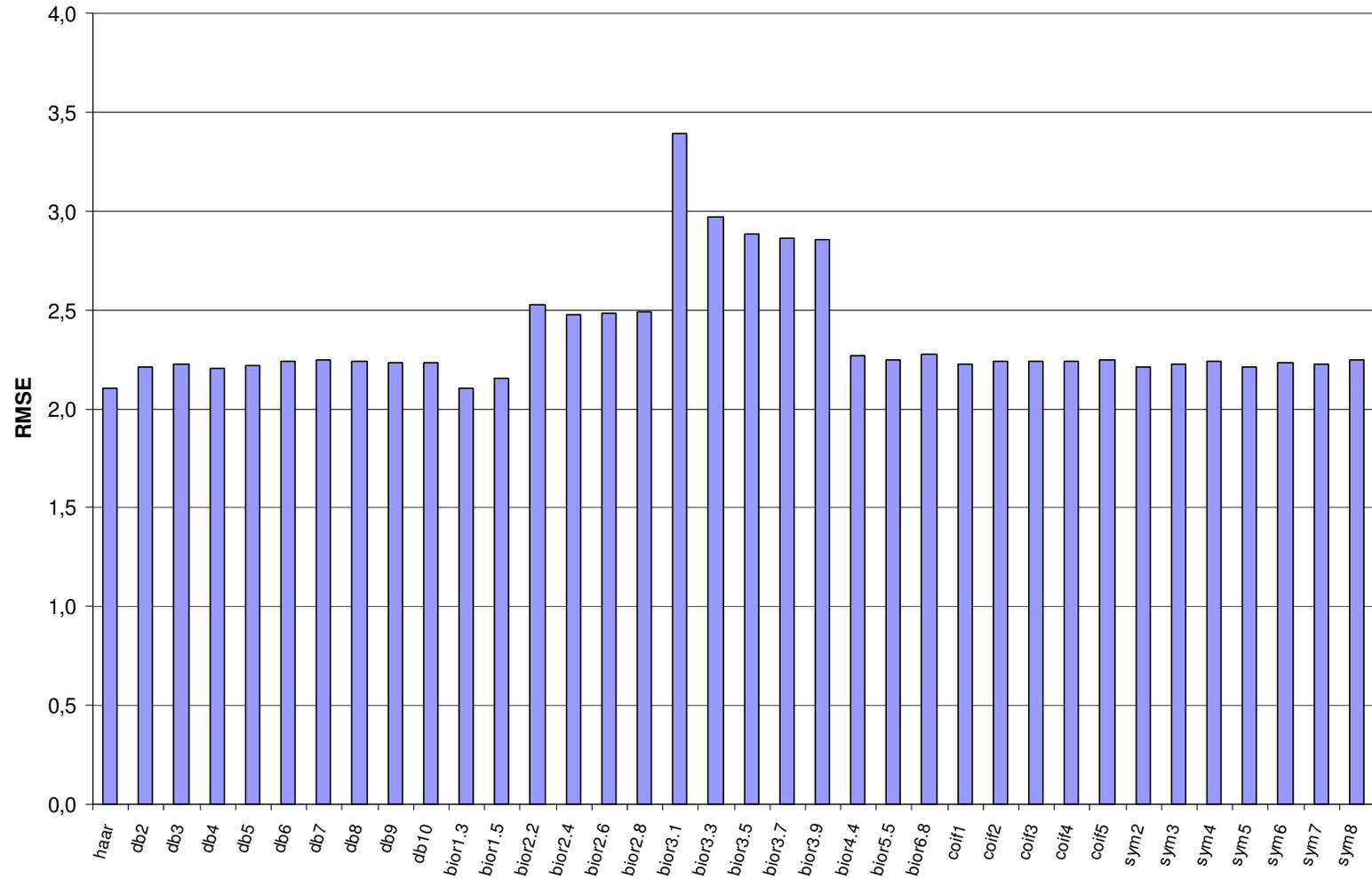
combining results

Fotográficas 128X128



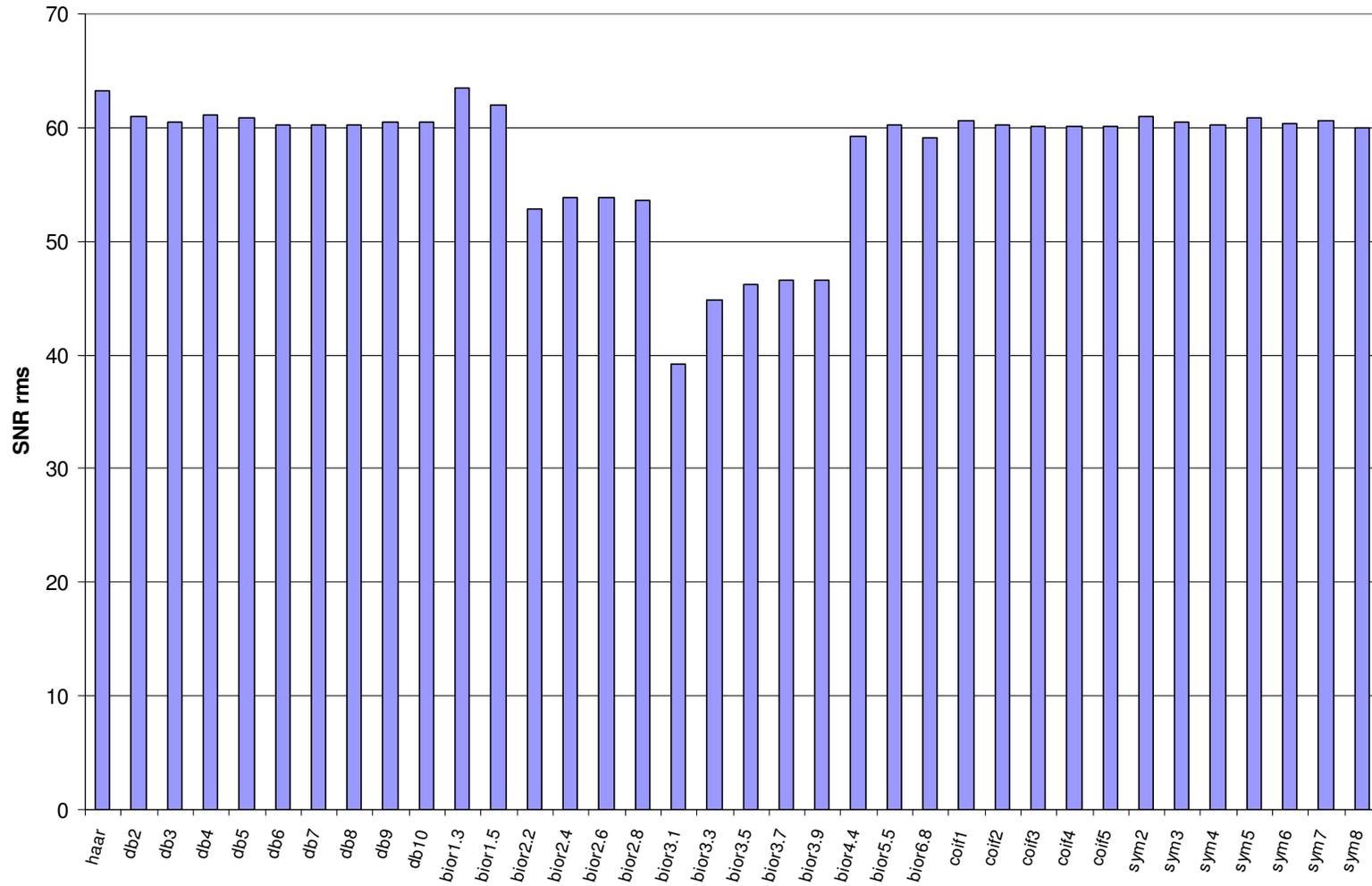
combining results

Fotográficas 256x256



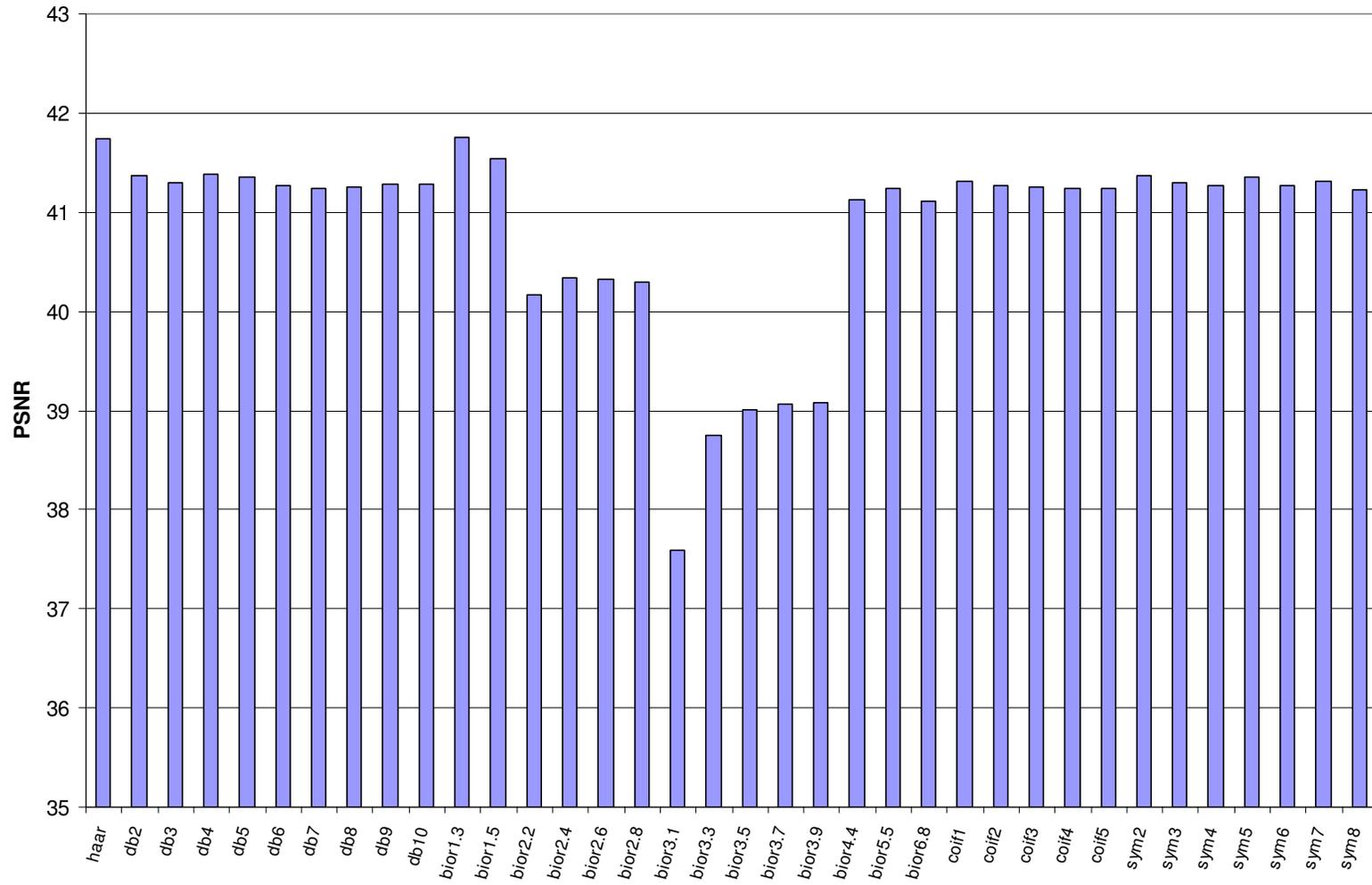
combining results

Fotográficas 256x256

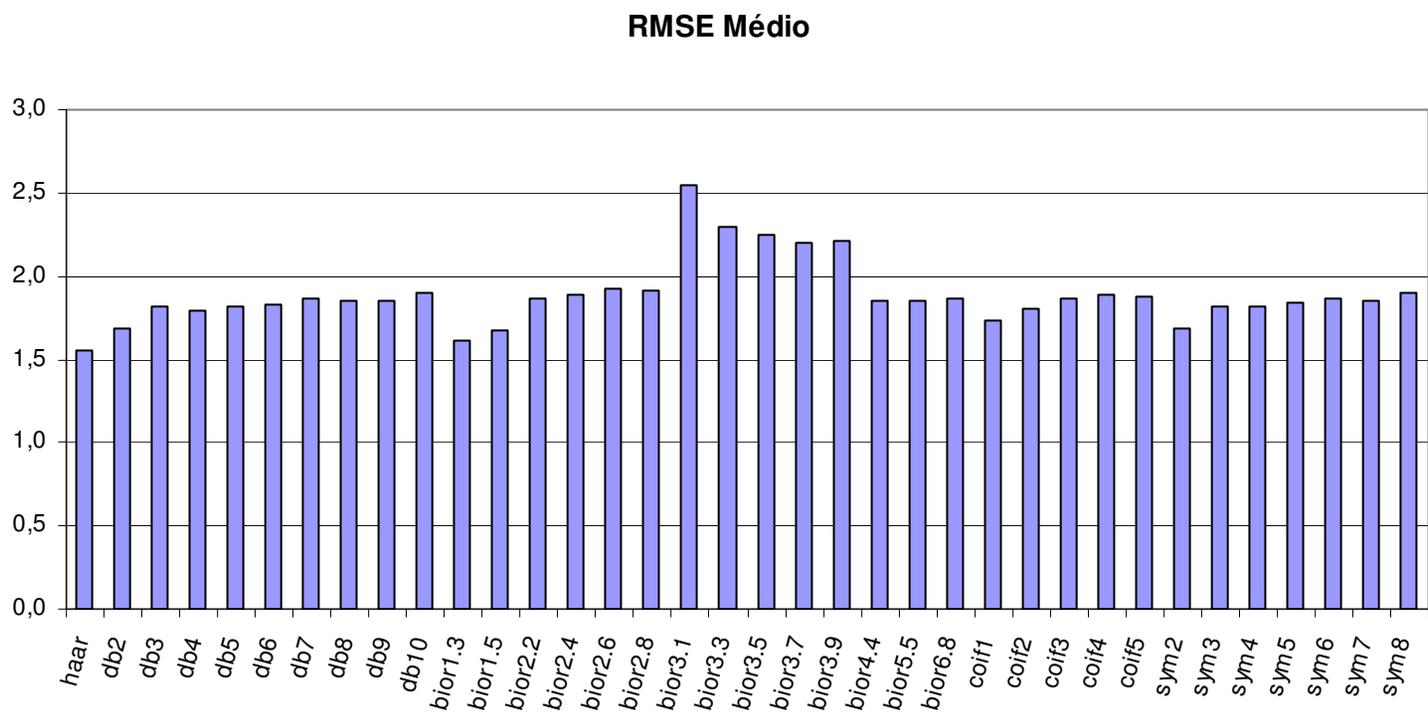


combining results

Fotográficas 256x256

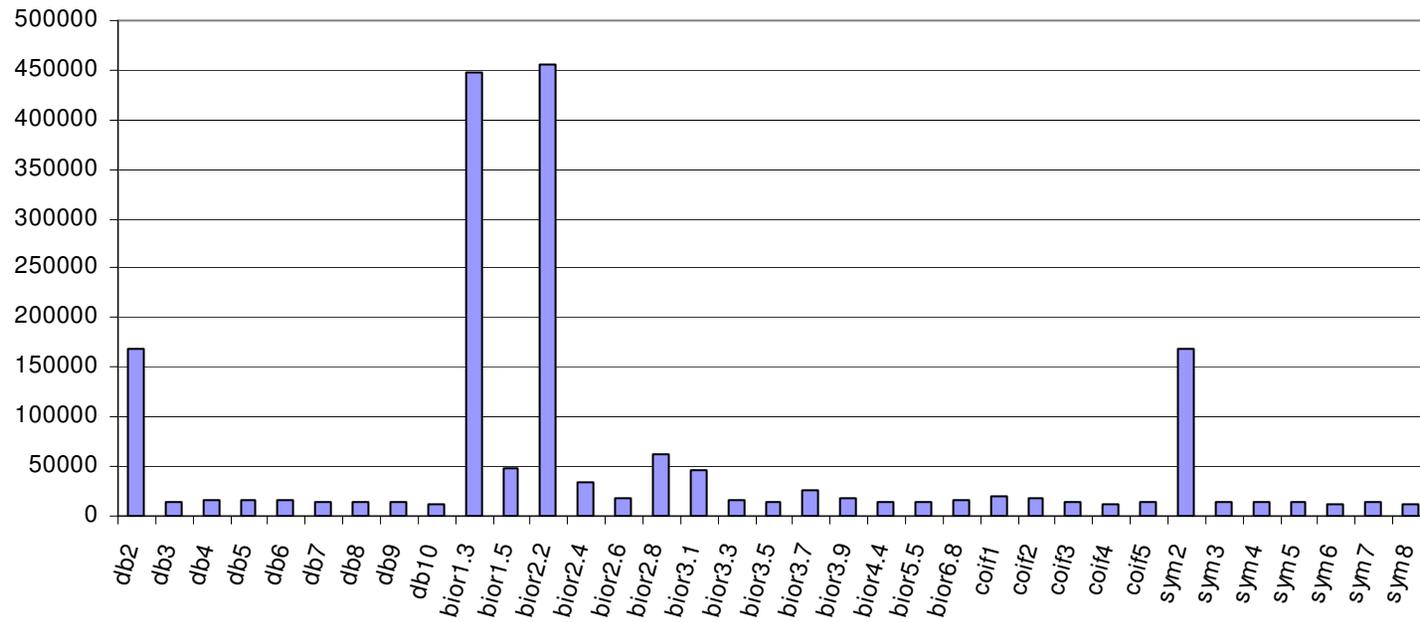


combining results



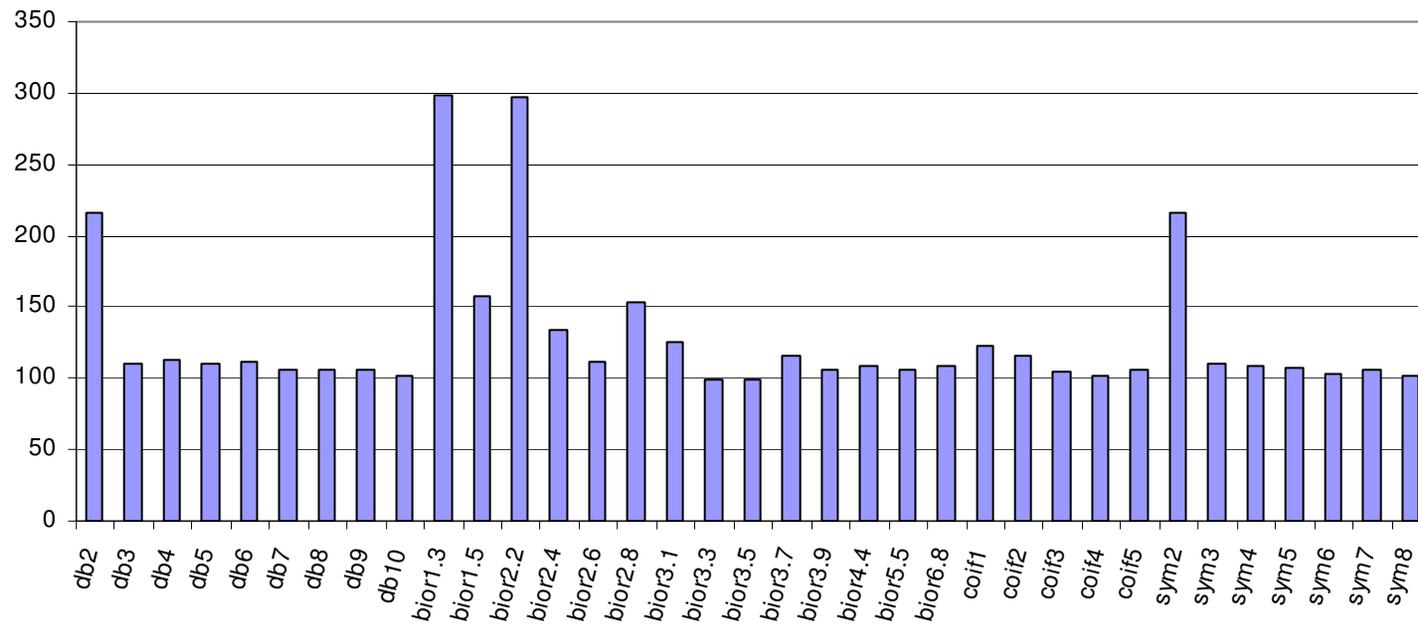
combining results

SNR ms Médio



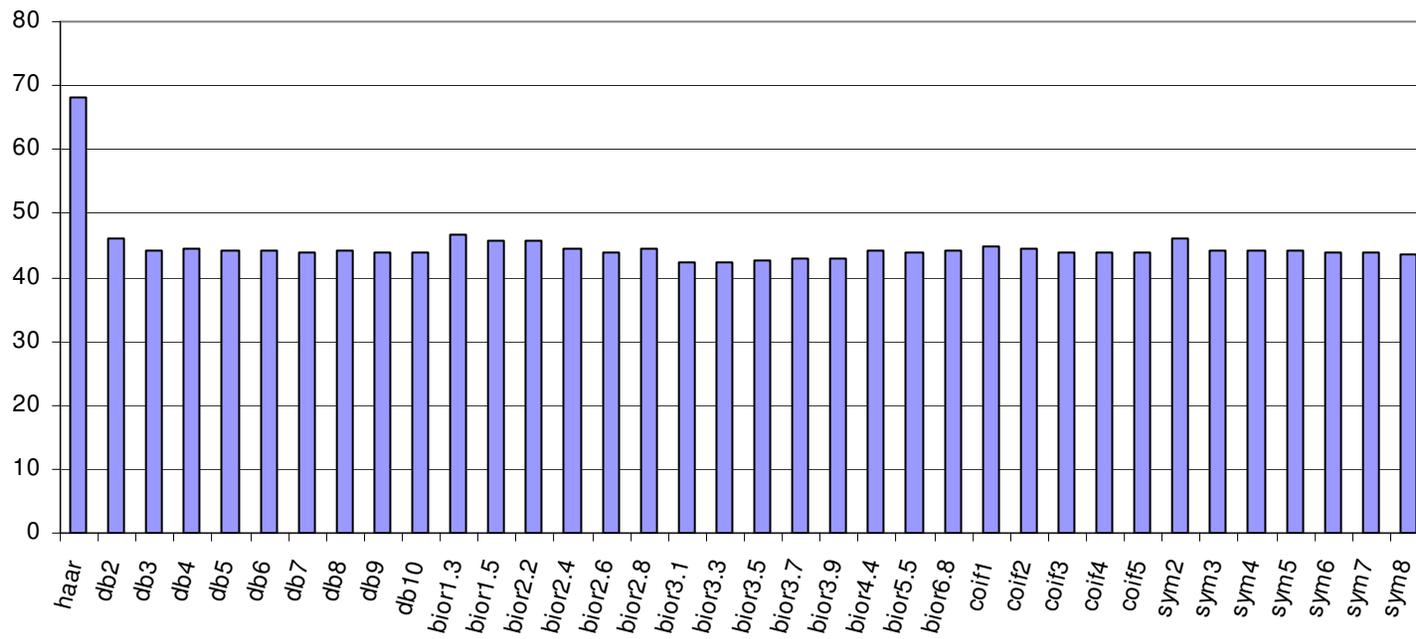
combining results

SNRrms Médio



combining results

PSNR Médio



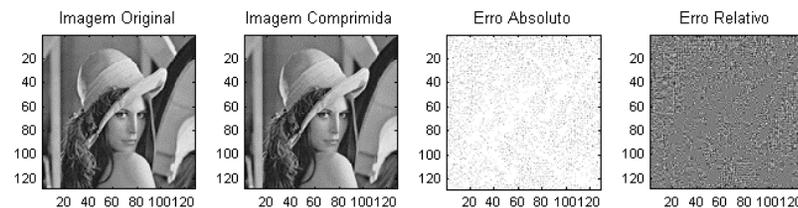
Conclusions

This work seeks to investigate the wavelet compression-denoising behavior related with 5 type of wavelet used (Haar, Daubechies, Biorthogonal, Coiflets and Symlets and a total of 36 subtypes) , image content and noise level.

Comparing thresholding process to remove additive noise from three noise level (low 5%, median 10% and high 20%) ; and 3 levels of resolution (128x128, 256x256 and 512x512)

The target of the work is to define which combination present the best and the worst results

The best choice related to quality is more dependent to the image content



Mining the results

- The graphs for **natural images** related with the **wavelets type** used have been presents **similar distribution** considering **all noise level** and **denoising approach**.
- The **non binary synthetic images** (text and sinusoidal) presents more **similar behaviour** for **median** and **high noise level**.
- But the **circle and chessboard images** present **no characteristics** that permits adequate conclusion related with what could be pointed as best **wavelet type** for compression and **denoising**.

rating

- **Haar and Biortogonal 1.3** types presents the **first** and **second** best quality.
- **Worst** results are obtained with the **Biortogonal 3. 1** type.
- Considering the **image content**, they show more **dependent** to the image type and wavelet (Haar, Daubechies, Biorthogonal, Coiflets, or Symlets) used than could be expected.

Trabalho 3 2015:

- **CONSIDERING THE WAVELET TYPE AND CONTENTS ON THE COMPRESSION-DECOMPRESSION ASSOCIATED WITH IMPROVEMENT OF BLURRED IMAGES**
- Two Level DWT of this image is computed with Daubechies' db4 wavelet.

Verificar a possibilidade do armazenamento das imagens térmicas por wavelets

- 1- usando a região das mamas das imagens térmicas (ROI) em escala diática ($N \times N$ onde $N=2^n$) Verifique a eficiência de seu armazenamento com perda por Wavelets.
- 2- Verifique a eficiência de comprimir o seu armazenamento, por limiarização dos coeficientes.

Referências

- Benedetto, J. J. and Frazier, M. (Eds.). *Wavelets: Mathematics and Applications*. Boca Raton, FL: CRC Press, 1994.
- Chui, C. K. *An Introduction to Wavelets*. San Diego, CA: Academic Press, 1992.
- Chui, C. K. (Ed.). *Wavelets: A Tutorial in Theory and Applications*. San Diego, CA: Academic Press, 1992.
- Chui, C. K.; Montefusco, L.; and Puccio, L. (Eds.). *Wavelets: Theory, Algorithms, and Applications*. San Diego, CA: Academic Press, 1994.
- Daubechies, I. *Ten Lectures on Wavelets*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1992.
- Erlebacher, G. H.; Hussaini, M. Y.; and Jameson, L. M. (Eds.). *Wavelets: Theory and Applications*. New York: Oxford University Press, 1996.
- Foufoula-Georgiou, E. and Kumar, P. (Eds.). *Wavelets in Geophysics*. San Diego, CA: Academic Press, 1994.
- Hernández, E. and Weiss, G. *A First Course on Wavelets*. Boca Raton, FL: CRC Press, 1996.
- Hubbard, B. B. *The World According to Wavelets: The Story of a Mathematical Technique in the Making, 2nd rev. upd. ed.* New York: A K Peters, 1998.
- Jawerth, B. and Sweldens, W. "An Overview of Wavelet Based Multiresolution Analysis." *SIAM Rev.* **36**, 377-412, 1994.
- Kaiser, G. *A Friendly Guide to Wavelets*. Cambridge, MA: Birkhäuser, 1994.
- Massopust, P. R. *Fractal Functions, Fractal Surfaces, and Wavelets*. San Diego, CA: Academic Press, 1994.
- Meyer, Y. *Wavelets: Algorithms and Applications*. Philadelphia, PA: SIAM Press, 1993.
- Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. "Wavelet Transforms." §13.10 in *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, pp. 584-599, 1992.
- Resnikoff, H. L. and Wells, R. O. J. *Wavelet Analysis: The Scalable Structure of Information*. New York: Springer-Verlag, 1998.
- Schumaker, L. L. and Webb, G. (Eds.). *Recent Advances in Wavelet Analysis*. San Diego, CA: Academic Press, 1993.
- Stollnitz, E. J.; DeRose, T. D.; and Salesin, D. H. "Wavelets for Computer Graphics: A Primer, Part 1." *IEEE Computer Graphics and Appl.* **15**, No. 3, 76-84, 1995.
- Stollnitz, E. J.; DeRose, T. D.; and Salesin, D. H. "Wavelets for Computer Graphics: A Primer, Part 2." *IEEE Computer Graphics and Appl.* **15**, No. 4, 75-85, 1995.
- Strang, G. "Wavelets and Dilation Equations: A Brief Introduction." *SIAM Rev.* **31**, 614-627, 1989.
- Strang, G. "Wavelets." *Amer. Sci.* **82**, 250-255, 1994.
- Taswell, C. *Handbook of Wavelet Transform Algorithms*. Boston, MA: Birkhäuser, 1996.
- Teolis, A. *Computational Signal Processing with Wavelets*. Boston, MA: Birkhäuser, 1997.
- Vidakovic, B. *Statistical Modeling by Wavelets*. New York: Wiley, 1999.
- Walker, J. S. *A Primer on Wavelets and their Scientific Applications*. Boca Raton, FL: CRC Press, 1999.
- Walter, G. G. *Wavelets and Other Orthogonal Systems with Applications*. Boca Raton, FL: CRC Press, 1994.
- Weisstein, E. W. "Books about Wavelets." <http://www.ericweisstein.com/encyclopedias/books/Wavelets.html>.
- Wickerhauser, M. V. *Adapted Wavelet Analysis from Theory to Software*. Wellesley, MA: Peters, 1994.