

Melhorando a Exatidão de Cálculo da Transformada Wavelet de Daubechies

Orientando: **Vinícius Rodrigues dos Santos**

Orientador: **Renata Hax Sander Reiser**

Coorientador: **Maurício Lima Pilla**

Colaborador: **Alice de Jesus Kozakevicius**

Universidade Federal de Pelotas
Ciência da Computação
Centro de Desenvolvimento Tecnológico

30/09/2016 - PELOTAS / RS

- 1 Introdução
- 2 Transformada Wavelet de Daubechies
- 3 Otimizações
- 4 Testes e Resultados
- 5 Considerações Finais

Contextualização

Transformadas Wavelets

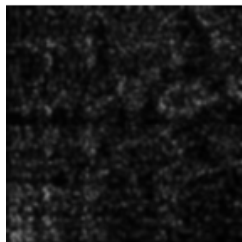
- Ferramentas matemáticas para a **decomposição hierárquica de funções**
- Utilizadas em **análise de sinais** para identificação de detalhes [1]
- Importante característica consiste na **composição** dos sinais transformados **reconstituindo** o sinal original

Aplicações em Bioinformática e Processamento de Imagens

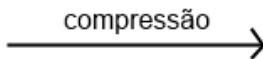
- **Compressão de imagens** [2] e remoção de ruído [3]
- Análise de **sinais de eletroencefalogramas** [4]
- **Reconstrução de marcas d'água** invisíveis em imagens digitais [5]
- Filtragem **sinais cardíacos** através de algoritmos adaptativos [6]
- Descrição de parâmetros estatísticos em **exames de mamografia** [7].



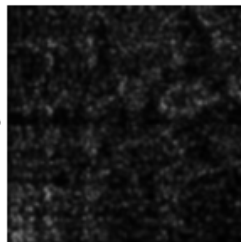
(a)



(b)



(d)



(c)

Principais Objetivos

- Desenvolver a biblioteca **Int-DWTs** com implementações intervalares e otimizadas para Transformadas Wavelet Discretas (TDWs) [8, 9]
- Prover **resultados confiáveis e automaticamente validados** para aplicações que utilizam de TDWs com foco em Computação Científica [10, 11]
- Aplicar **métricas de avaliação da qualidade** nos algoritmos implementados

Principais Contribuições

- **Otimização da extensão intervalar** para as Transformadas Wavelet de Haar (TWH) e de Daubechies (TWD) [12]
 - **Abordagens:** não-normalizada não-padrão, não-normalizada padrão, normalizada não-padrão e normalizada padrão
 - **Multi-dimensionais:** 1D e 2D
 - **Etapas:** transformação direta (decomposição) e de inversa (composição)



Funcionamento Básico

Decomposição	Composição
$C'[i] = (C[2i] + C[2i + 1]) / 2$	$C'[2*i] = C[i] + C[n + i]$
$C'[n/2 + i] = (C[2i] - C[2i + 1]) / 2$	$C'[2*i + 1] = C[i] - C[n + i]$

n = tamanho do vetor; $i = 0 \dots n/2 - 1$;

—	(de escala)	(wavelets)
Nível	Médias	Diferenças
2	[9 7 3 5]	-
1	[8 4]	[1 -1]
0	[6]	[2]

- Vetor original: [9 7 3 5]
- Vetor transformado: [6 2 1 -1]

```
procedure DecompositionStep(C: array [1..h] of reals)  
  for i  $\leftarrow$  1 to h/2 do  
     $C'[i] \leftarrow (C[2i - 1] + C[2i]) / \underline{\sqrt{2}}$    
     $C'[h/2 + i] \leftarrow (C[2i - 1] - C[2i]) / \underline{\sqrt{2}}$    
  end for  
  C  $\leftarrow$  C'  
end procedure
```

```
procedure Decomposition(C: array [1..h] of reals)  
  C  $\leftarrow$  C /  $\sqrt{h}$  (normalize input coefficients)  
  while h > 1 do  
    DecompositionStep(C[1..h])  
    h  $\leftarrow$  h/2  
  end while  
end procedure
```

Transformada Wavelet de Daubechies

$$h = \left(\frac{1 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{3 - \sqrt{3}}{4\sqrt{2}}, \frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (1)$$

$$g = \left(\frac{1 - \sqrt{3}}{4\sqrt{2}}, \frac{-3 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{-1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (2)$$

$$ih = \left(\frac{3 - \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{1 + \sqrt{3}}{4\sqrt{2}}, \frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (3)$$

$$ig = \left(\frac{1 - \sqrt{3}}{4\sqrt{2}}, \frac{-1 - \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{-1 + \sqrt{3}}{4\sqrt{2}} \right) \quad (4)$$

DecompositionStep ($C[1..n]$)

```
1  $i \leftarrow 1$  ;  
2  $j \leftarrow 1$  ;  
3  $half \leftarrow n/2$  ;  
4 while  $j \leq n - 3$  do  
5    $C'[i] \leftarrow C[j].h1 + C[j+1].h2 + C[j+2].h3 + C[j+3].h4$  ;  
6    $C'[i+half] \leftarrow C[j].g1 + C[j+1].g2 + C[j+2].g3 + C[j+3].g4$   
    $j \leftarrow j + 2$  ;  
7    $i \leftarrow i + 1$  ;  
8 end  
9  $C'[i] \leftarrow C[n-1].h1 + C[n].h2 + C[1].h3 + C[2].h4$  ;  
10  $C'[i+half] \leftarrow C[n-1].g1 + C[n].g2 + C[1].g3 + C[2].g4$  ;  
11  $C \leftarrow C'$  ;
```

$$h = a \left(\frac{1 + \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{3 - \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (5)$$

$$g = a \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{\sqrt{3} - 3}{4\sqrt{2}} \right) + c \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{-1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (6)$$

$$ih = a \left(\frac{3 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{1 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \quad (7)$$

$$ig = a \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{-1 - \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{\sqrt{3} - 3}{4\sqrt{2}} \right) \quad (8)$$

Otimizações

Base de Otimização

$$R = 2^{\frac{-j}{2}}$$

$$[9 \quad 7 \quad 3 \quad 5] \rightarrow [6 \quad 2 \quad 1 \quad -1] \rightarrow [6 \quad 2 \quad \frac{1}{\sqrt{2}} \quad \frac{-1}{\sqrt{2}}]$$

Entrada \rightarrow Decomposição \rightarrow Normalização

Simplificação dos Irracionais nos Filtros: h, g, ih, ig

$$\begin{aligned}h &= a \left(\frac{1 + \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{3 - \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \\&= \frac{a(1 + \sqrt{3}) + b(3 + \sqrt{3}) + c(3 - \sqrt{3}) + d(1 - \sqrt{3})}{4\sqrt{2}} \\&= \frac{a + a\sqrt{3} + 3b + b\sqrt{3} + 3c - c\sqrt{3} + d - d\sqrt{3}}{4\sqrt{2}} \\&= \frac{a + d + 3(b + c) + \sqrt{3}(a + b - c - d)}{4\sqrt{2}}\end{aligned}\tag{9}$$

Simplificação dos Irracionais nos Filtros: h, g, ih, ig

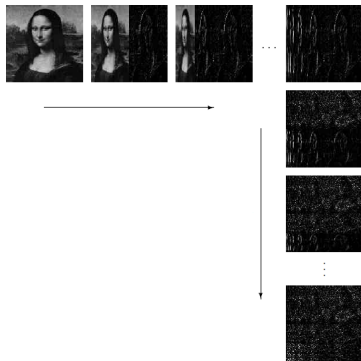
$$\begin{aligned}g &= a \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{\sqrt{3} - 3}{4\sqrt{2}} \right) + c \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{-1 - \sqrt{3}}{4\sqrt{2}} \right) \\&= \frac{a(1 - \sqrt{3}) + b(\sqrt{3} - 3) + c(3 + \sqrt{3}) + d(-1 - \sqrt{3})}{4\sqrt{2}} \\&= \frac{a - a\sqrt{3} + b\sqrt{3} - 3b + 3c + c\sqrt{3} - d - d\sqrt{3}}{4\sqrt{2}} \\&= \frac{a - d + 3(c - b) + \sqrt{3}(b + c - a - d)}{4\sqrt{2}}\end{aligned}\tag{10}$$

Simplificação dos Irracionais nos Filtros: h, g, ih, ig

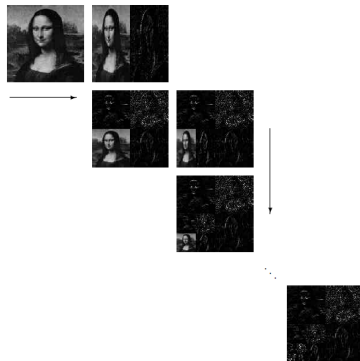
$$\begin{aligned} ih &= a \left(\frac{3 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{1 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) \\ &= \frac{a(3 - \sqrt{3}) + b(3 + \sqrt{3}) + c(1 + \sqrt{3}) + d(1 - \sqrt{3})}{4\sqrt{2}} \\ &= \frac{3a - a\sqrt{3} + 3b + b\sqrt{3} + c + c\sqrt{3} + d - d\sqrt{3}}{4\sqrt{2}} \\ &= \frac{c + d + 3(a + b) + \sqrt{3}(b + c - a - d)}{4\sqrt{2}} \end{aligned} \tag{11}$$

Simplificação dos Irracionais nos Filtros: h, g, ih, ig

$$\begin{aligned} ig &= a \left(\frac{1 - \sqrt{3}}{4\sqrt{2}} \right) + b \left(\frac{-1 - \sqrt{3}}{4\sqrt{2}} \right) + c \left(\frac{3 + \sqrt{3}}{4\sqrt{2}} \right) + d \left(\frac{\sqrt{3} - 3}{4\sqrt{2}} \right) \\ &= \frac{a(1 - \sqrt{3}) + b(-1 - \sqrt{3}) + c(3 + \sqrt{3}) + d(\sqrt{3} - 3)}{4\sqrt{2}} \\ &= \frac{a - a\sqrt{3} - b - b\sqrt{3} + 3c + c\sqrt{3} - 3d + d\sqrt{3}}{4\sqrt{2}} \\ &= \frac{a - b + 3(c - d) + \sqrt{3}(c + d - a - b)}{4\sqrt{2}} \end{aligned} \tag{12}$$



(a) Algoritmo Padrão



(b) Algoritmo Não-Padrão

Padrão de fatores de normalização (Algoritmo Padrão)

	...	3	2	1
⋮				
3	(3,3)	(3,2)	(3,1)	
2	(2,3)	(2,2)	(2,1)	
1	(1,3)	(1,2)	(1,1)	

(a)

$$(j', j'')$$

(b)

$$2^{\frac{-(j' + j'')}{2}}$$

(c)

(a) níveis de aplicação, (b) combinação de níveis, (c) regra de normalização

Metodologia

- **Performance:** Relação de desempenho das execuções
- **Accuracy:** Relação entre o diâmetro do intervalo e o erro na computação
- **Metrics:** Aplicação de medidas de qualidade dos resultados

$$\text{EUC}(\tilde{\mathbf{Y}}, \mathbf{Y}) = \sqrt{\sum_{j=0}^m \sum_{i=0}^n (\tilde{\mathbf{Y}}_{ij} - \mathbf{Y}_{ij})^2} \quad (13)$$

$$\text{MSE}(\tilde{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{nm} \sum_{j=0}^m \sum_{i=0}^n (\tilde{\mathbf{Y}}_{ij} - \mathbf{Y}_{ij})^2 \quad (14)$$

$$\text{PSNR}(\tilde{\mathbf{Y}}, \mathbf{Y}_{ij}) = 10 \cdot \log_{10} \frac{MAX^2}{\text{MSE}(\tilde{\mathbf{Y}}, \mathbf{Y})} \quad (15)$$

Configuração da Máquina

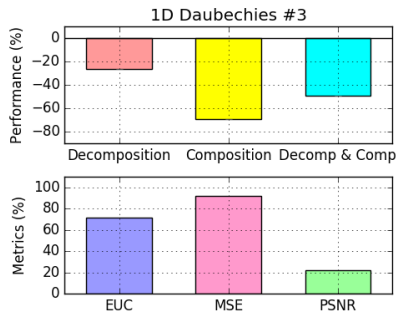
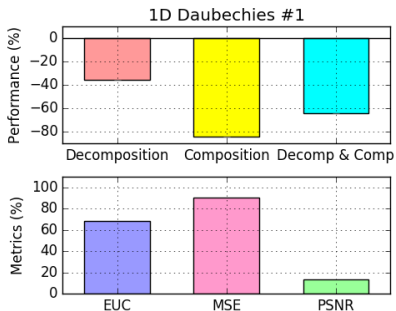
Processador: Intel[®] Core[™] i7 950 Processor @ 3.07GHz
Memória: 6GB DDR3 @ 1066MHz
SO: Windows 10
Compilador: Microsoft Visual C++ VS2013 (x64 Release)

Parâmetros de Testes

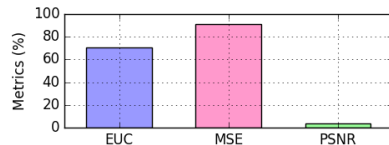
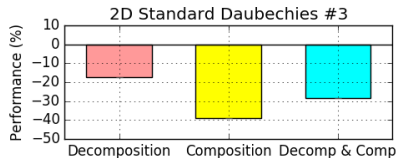
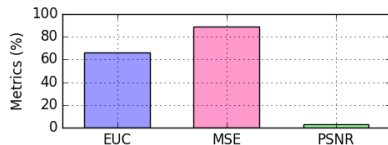
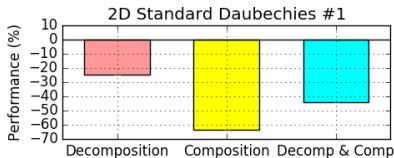
• TWD 1D e 2D

- **Teste #1:** 1024×1024 valores aleatórios - 2^{20} coeficientes
- **Teste #2:** 2048×2048 valores aleatórios - 2^{22} coeficientes
- **Teste #3:** 4096×4096 valores aleatórios - 2^{24} coeficientes

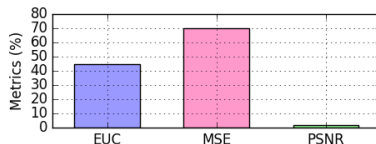
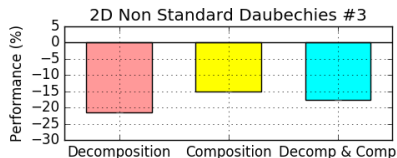
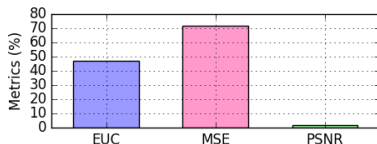
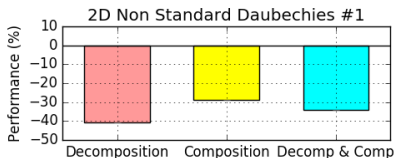
Testes #1 e #3 para a TWD 1D



Testes #1 e #3 para a TWD 2D Padrão



Testes #1 e #3 para a TWD 2D Não-Padrão



Número de Operações

Tabela: TWD: número de operações por decomposição

	1D	2D Padrão	2D Não-Padrão
Originais	$n \log(n)$	$n^2 \log(n)$	$n^2 + \log(\log(n) + \log(n))$
Desenvolvidos	$n \log(n) + n$	$n^2 \log(n) + n^2$	$n^2 + \log(\log(n) + \log(n))$

Conclusão

- Pelas métricas aplicadas, as otimizações da biblioteca Int-DWTs alcançaram ganhos satisfatórios quanto a **exatidão dos cálculos** e **qualidade dos resultados**
 - versões intervalares dos algoritmos originais
 - algoritmos otimizados (12)
- Pela análise do número de operações, foram discutidas as principais causas da perda de desempenho

Trabalhos Futuros

- Implementação concorrente dos algoritmos da biblioteca Int-DWTs suavizando execução em GPU
- Extensão da biblioteca Int-DWTs para outras classes de Transformadas Wavelet Discretas
- Utilização das otimizações da biblioteca Int-DWTs em aplicações na Bioinformática e no Processamento de Sinais

Bibliografia I

- [1] A.E. Brito and O. Kosheleva.
Interval + image = wavelet: for image processing under interval uncertainty, wavelets are optimal.
Reliable Computing, pages 291–301, 1998.
- [2] C. Christopoulos, A. Skodras, and T. Ebrahimi.
The jpeg2000 still image coding system: an overview.
Consumer Electronics, IEEE Transactions on, 46(4):1103–1127, Nov 2000.
- [3] D. Donoho.
De-noising by soft-thresholding.
IEEE Transactions on Information Theory, 41:613–627, 1995.
- [4] T.L.T. Silveira, A.J. Kozakevicius, and C.R. Rodrigues.
Classificação de estágios de sono através da aplicação de transformada wavelet discreta sobre um único canal de eletroencefalograma.
Master's thesis, 2016.
- [5] T. Minamoto and K. Aoki.
A blind digital image watermarking using interval wavelet decomposition.
Int. Journal of Signal proc., Image proc. and Pattern recognition, 3(2):59–72, 2010.
- [6] G. Perin and A. J. Kozakevicius.
Filtragem wavelet de sinais cardíacos através de algoritmos adaptativos.
RITA (in portuguese), 20(3):95–111, 2013.

Bibliografia II

- [7] M. Asadzadeh, E. Hashemi, and A. Kozakevicius.
On efficiency of combined daubechies wavelets and statistical parameters applied in mammography.
Applied and Computational Mathematics, 12(3):289–306, 2013.
- [8] O. M. Nielsen.
Wavelets in scientific computing, 1998.
Supervisor: Per Christian Hansen, pcha@dtu.dk, DTU Compute.
- [9] Ingrid Daubechies.
Orthonormal bases of compactly supported wavelets.
Communications on Pure and Applied Mathematics, 41(7):909–996, 1988.
- [10] R.E. Moore.
Methods and Applications of Interval Analysis.
SIAM, Philadelphia, 1979.
- [11] W. Hofschuster, W. Kramer, and M. Neher.
C-XSC and Closely Related Software Packages.
Numerical Validation in Current Hardware Architectures. Springer-Verlag, Universitat Wuppertal, 2008.
- [12] E. J. Stollnitz, T. D. DeRose, and D. H. Salesin.
Wavelets for computer graphics: A primer, part 1.
IEEE Computer Graphics and Applications, 15(3):76–84, 1995.

Agradecimento e Dúvidas

Agradecimentos

- PROBIT/FAPERGS (08/2015 a 07/2016)
- Projeto PqG/2013 FAPERGS

Dúvidas

Contatos:

- vrdsantos@inf.ufpel.edu.br

Melhorando a Exatidão de Cálculo da Transformada Wavelet de Daubechies

Orientando: **Vinícius Rodrigues dos Santos**

Orientador: **Renata Hax Sander Reiser**

Coorientador: **Maurício Lima Pilla**

Colaborador: **Alice de Jesus Kozakevicius**

Universidade Federal de Pelotas
Ciência da Computação
Centro de Desenvolvimento Tecnológico

30/09/2016 - PELOTAS / RS

Evolução do Trabalho

Principais Publicações 2015-2016

- **ERAD 2015:** Estudo de Desempenho sobre a Biblioteca Int-Haar
- **SIM 2015:** Int-HWT: Increasing Performance and Exactitude of 1D and 2D Haar Wavelet Transforms
- **WEIT 2015:** Transformada de Haar Não Decimada Otimizada e sua Extensão Intervalar
- **ERAD 2016:** Transformada de Haar À Trous Otimizada e Análise de Desempenho
- **CNMAC 2016:** Otimização para Aumento de Exatidão da Transformada Wavelet de Haar À-Trous
- **TCC 2016/1:** Int-DWTs Library - Algebraic Simplifications Increasing Performance and Accuracy of Discrete Wavelet Transforms

Publicação em Periódico - Computational and Applied Mathematics

- Int-HWT: Algebraic Simplifications Increasing Performance And Exactitude of the Haar Wavelet Transforms

Métricas

Error

```
real INT_error(interval *x, int n)
{
    real r = 0.0;

    for (int i = 0; i < n; i++)
        if (INT_diameter(x[i]) > r) r = INT_diameter(x[i]);

    return r;
}
```

Distância Euclidiana

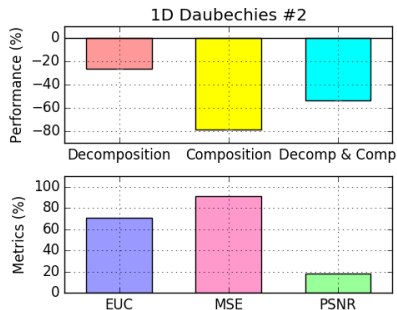
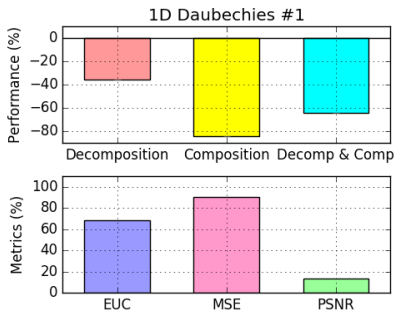
$$\text{EUC}(\tilde{\mathbf{Y}}, \mathbf{Y}) = \sqrt{\sum_{j=0}^m \sum_{i=0}^n (\tilde{\mathbf{Y}}_{ij} - \mathbf{Y}_{ij})^2}.$$

Mean Squared Error

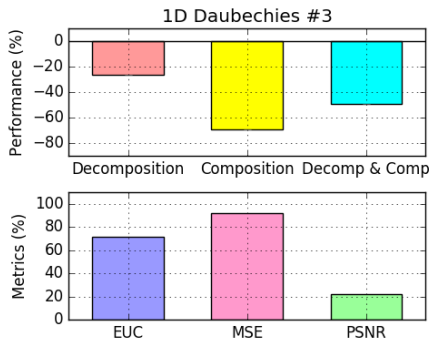
$$\text{MSE}(\tilde{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{nm} \sum_{j=0}^m \sum_{i=0}^n (\tilde{\mathbf{Y}}_{ij} - \mathbf{Y}_{ij})^2$$

Peak Noise-To-Signal

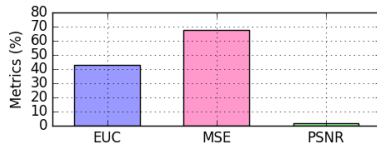
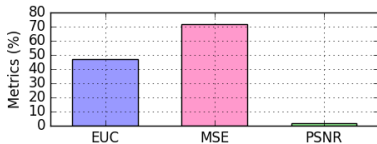
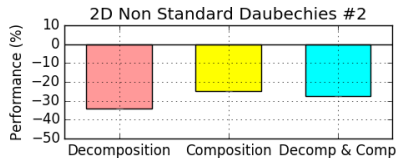
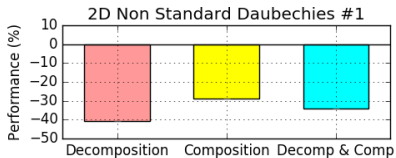
$$\mathbf{PSNR}(\tilde{\mathbf{Y}}, \mathbf{Y}_{ij}) = 10 \cdot \log_{10} \frac{MAX^2}{\mathbf{MSE}(\tilde{\mathbf{Y}}, \mathbf{Y})}.$$



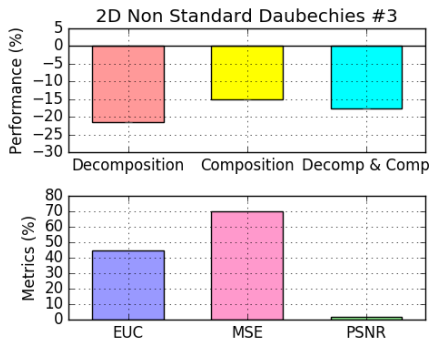
Testes #1 e #2 para a **TWD 1D** utilizando 1048576 e 4194304 valores aleatórios.



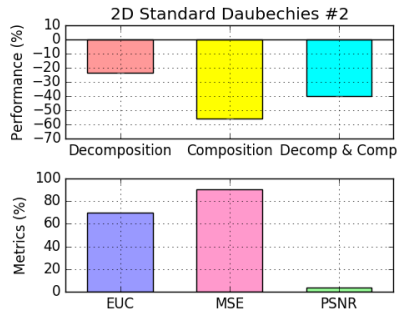
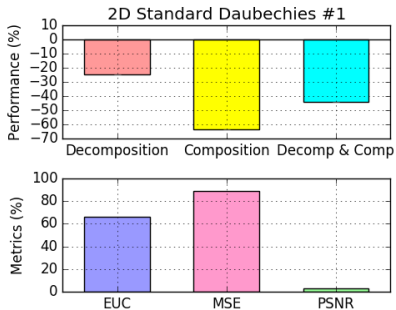
Teste #3 para a **TWD 1D** utilizando 16777216 valores aleatórios.



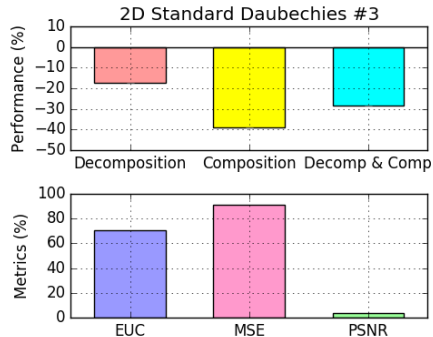
Testes #1 e #2 para a **TWD 2D Não-Padrão** utilizando 1024x1024 e 2048x2048 valores aleatórios.



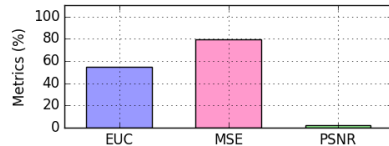
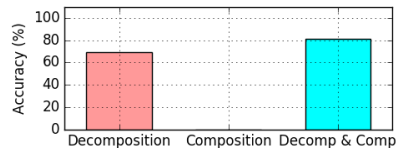
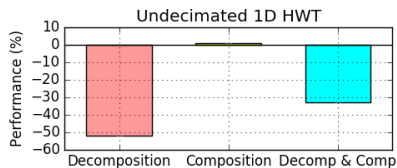
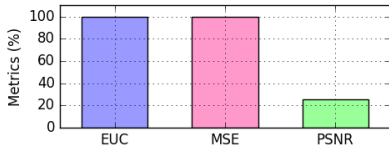
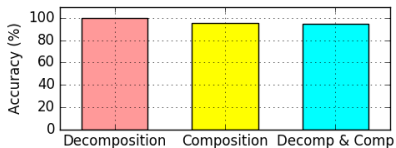
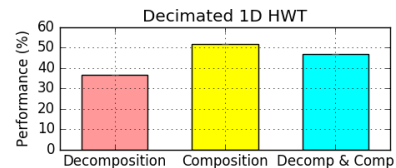
Teste #3 para a **TWD 2D Não-Padrão** utilizando 4096x4096 valores aleatórios.



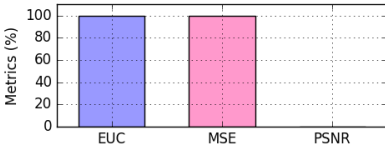
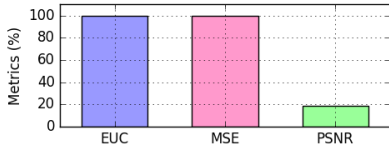
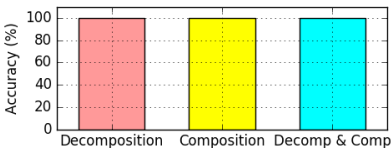
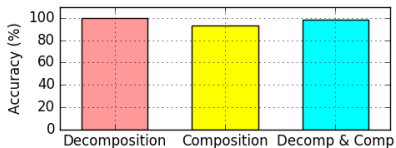
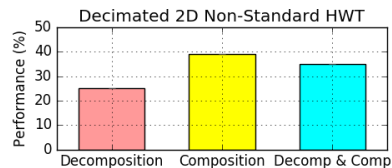
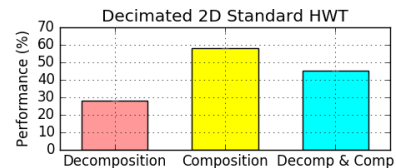
Testes #1 e #2 para a **TWD 2D Padrão** utilizando 1024x1024 e 2048x2048 valores aleatórios.



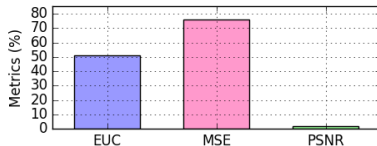
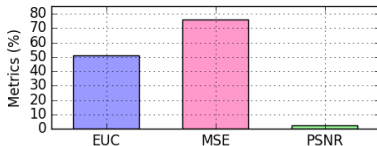
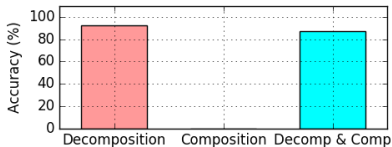
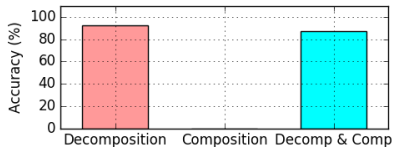
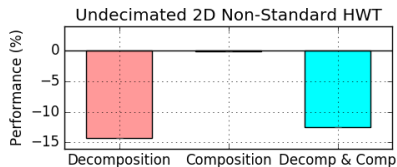
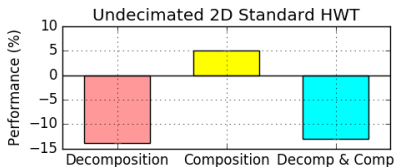
Teste #3 para a **TWD 2D Padrão** utilizando 4096x4096 valores aleatórios



Testes para a **TWH 1D** utilizando 1048576 valores aleatórios.



Testes para a **TWH 2D Padrão e Não-Padrão** utilizando 1024x1024 valores aleatórios.



Testes para a **TWH À-Trous 2D Padrão e Não-Padrão** utilizando 1024x1024 valores aleatórios.