

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE COMUNICAÇÕES



DISSERTAÇÃO DE MESTRADO

IMPACTO DA UTILIZAÇÃO DE TÉCNICAS DE *LIFTING* EXPLORANDO
DIFERENÇA ENTRE FERRAMENTAS INTERPRETADA E COMPILADA
NA VELOCIDADE COMPUTACIONAL DA CODIFICAÇÃO *WAVELET*
DE IMAGENS ESTÁTICAS

Lucas de Oliveira

Orientador: Prof. Dr. Yuzo Iano

Co-orientadora: Dra. Ana Lúcia M. C. S. Silva

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Telecomunicações e Telemática.

Banca Examinadora:

Prof. Dr. Omar Carvalho Branquinho (PUCC)

Prof. Dr. Luiz César Martini (FEEC – UNICAMP)

Prof. Dr. Fernando Silvestre da Silva (UNISAL)

Prof. Dr. José Antonio Siqueira Dias (FEEC – UNICAMP)

Campinas, SP

2009

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Oliveira, Lucas de
OL41i Impacto da utilização de técnicas de lifting
explorando diferença entre ferramentas interpretada e
compilada na velocidade computacional da codificação
wavelet de imagens estáticas / Lucas de Oliveira.
–Campinas, SP: [s.n.], 2009.

Orientadores: Yuzo Iano, Ana Lúcia Mendes Cruz
Silvestre da Silva.

Dissertação de Mestrado - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Processamento de imagens - Técnicas digitais. 2.
Teoria da codificação. 3. Wavelet (Matemática). 4.
Compressão de imagens. 5. Comunicações digitais. I. Iano,
Yuzo. II. Silva, Ana Lúcia Mendes Cruz Silvestre da. III.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. IV. Título.

Título em Inglês: Impact of using lifting techniques to explore the differences between
interpreted and compiled tools in the computational speed of wavelet
static image coding

Palavras-chave em Inglês: Image processing - Digital techniques, Coding theory, Wavelet
(Mathematics), Digital compression, Digital communications

Área de concentração: Telecomunicações e Telemática

Titulação: Mestre em Engenharia Elétrica

Banca Examinadora: Omar Carvalho Branquinho, Luiz César Martini

Data da defesa: 30/07/2009

Programa de Pós Graduação: Engenharia Elétrica

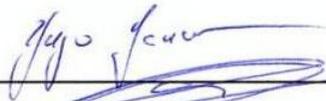
COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Lucas de Oliveira

Data da Defesa: 30 de julho de 2009

Título da Tese: "Impacto da Utilização de Técnicas de Lifting Explorando Diferença entre Ferramentas Interpretada e Compilada na Velocidade Computacional da Codificação Wavelet de Imagens Estáticas"

Prof. Dr. Yuzo Iano (Presidente):



Prof. Dr. Omar Carvalho Branquinho:



Prof. Dr. Luiz César Martini:



RESUMO

O campo de análise e compressão de imagens tem recebido especial atenção da comunidade científica recentemente por abranger os principais sistemas de TV digital, comunicações via satélite e comunicações móveis. Neste contexto, as Transformadas *Wavelets* Discretas (TWD) surgem como uma ferramenta poderosa e suas inúmeras vantagens permitiram sua inserção em importantes padrões tais como JPEG2000, MPEG4 e MPEG7. O método *lifting* realiza a TWD sub-amostrando o sinal antes do processo de filtragem, implicando em rápido processamento por fazer uso das similaridades entre filtros passa-alta e passa-baixa, acelerando o cálculo dos coeficientes. Na literatura, ganhos aritméticos teóricos de diferentes proporções foram obtidos sobre o método tradicional, destacando-se os trabalhos de Daubechies e Sweldens e de Reichel. O objetivo desta pesquisa consiste em estender esta discussão de resultados teóricos quando implementados através de ferramentas, interpretada e compilada, focando no tempo de processamento necessário para a realização (decomposição e reconstrução) de diferentes imagens estáticas empregando *wavelets* pertencentes às famílias de Daubechies, Symlets e Biorrtogonais. Medidas de PSNR foram utilizadas de forma a corroborar a perfeita implementação do *lifting*, dado que o mesmo não pode implicar em deterioração na imagem reconstruída.

ABSTRACT

The field of analysis and compression of images has been received special attention of the scientific community because of including in the main systems of digital TV, satellite communications and mobile communications. In this context, the Discrete Transform Wavelet (TWD) arise how a powerful tool and its many advantages allowed your insertion in important standards such as JPEG2000, MPEG4 and MPEG7. The lifting method realizes the TWD sub-sampling the signal before the filtering, resulting in faster processing by using the similarities between high and low pass filters, speeding up the calculation of coefficients. In the literature, arithmetic theoretical gains of different proportions were obtained on the traditional method, especially the works of Daubechies and Sweldens and of Reichel. The objective of this research is to extend this discussion of theoretical results when implemented through tools, interpreted and compiled, targeting the processing time required to perform (decomposition and reconstruction) of different images using wavelets belonging to the families of Daubechies, and Symlets Biorrtogonais. Measures of PSNR were used to corroborate the perfect implementation of the lifting, since that cannot result in deterioration in the reconstructed image.

AGRADECIMENTOS

Para a realização deste trabalho, gostaria de prestar meus agradecimentos às pessoas e instituições que contribuíram direta ou indiretamente para concretização do mesmo.

Primeiramente, gostaria de agradecer ao meu orientador, Professor Doutor Yuzo Iano, que durante todas as reuniões realizadas, sempre esteve estimulando o grupo de pesquisas com o intuito de buscar novas idéias e soluções para os problemas encontrados. Muito obrigado pela oportunidade!

Agradeço imensamente à minha co-orientadora, Professora Doutora Ana Lúcia Mendes Cruz Silvestre da Silva, por todas orientações sugeridas, as quais foram fundamentais para a realização do trabalho. Obrigado por toda atenção, pelos conselhos, oportunidades e por ter estado presente em todos os momentos que precisei .

Sinceros agradecimentos ao Professor Doutor Fernando S. Silva. Muito obrigado pela atenção, pelas sugestões e por todo conhecimento proporcionado!

Aos professores Dr. Luiz César Martini (UNICAMP) e Dr. Omar Carvalho Branquinho (PUCC) por participarem da banca examinadora.

À Universidade Estadual de Campinas (UNICAMP) pela oportunidade e apoio recebido durante a realização desta pesquisa. À Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo suporte financeiro concedido.

Aos meus pais, Antonio Celino e Clemilda, pela formação que me deram e por terem sido ponto de referência e apoio em toda minha vida. Muito me orgulho por ter vocês em minha vida.

Agradeço a meu pai, por sempre ter sido meu grande amigo, pela presença em toda minha criação, sempre me orientando em todas etapas de minha vida e por tudo que significa para mim. Será sempre o homem que vou buscar me espelhar, principalmente no que se refere ao seu modo extremamente organizado, pela sua dedicação em tudo que faz e por ter me ensinado que tudo que alcançamos na vida com nosso próprio esforço, possui um valor diferenciado.

À minha mãe, pela sua bondade de pessoa, por estar sempre animada e sorrindo para a vida, sempre alegrando não só a casa, como também meu dia a dia. Obrigado pela paciência durante todas as fases da minha vida e muito obrigado por sempre ter dado aquele “jeitinho” em todas as situações que precisei. Agradeço também por todas as orações realizadas, pois tenho certeza que estas sempre iluminaram meus passos.

Ao meu irmão, Leandro, que durante todas minhas fases sempre foi meu conselheiro, meu parceiro e o meu melhor amigo. Tenho muito carinho por você e sempre estarei disposto a lhe ajudar, pois ter você como meu irmão me ajudou a amadurecer. Te desejo todo o sucesso desse mundo.

À Bruna, minha noiva, companheira e melhor amiga, por estar presente em mais esta etapa da minha vida, sempre me confortando com seu amor, com sua forma carinhosa, atenciosa e prestativa de ser. Serei sempre grato pelo apoio e por todas as palavras encorajadoras, sempre me fortalecendo

em meus momentos mais difíceis. Mesmo não sendo da área, muito me ajudou dando forças e ânimo ao passar dias ao meu lado corrigindo este presente trabalho.

Agradeço a todos meus amigos do laboratório de Comunicações Visuais. Um agradecimento especial ao Ricardo, pessoa de bom coração, prestativa e de boa vontade, que muito me ajudou em um dos momentos mais difíceis nesta caminhada.

Agradeço aos meus amigos, em especial a Luís Otávio e Paulo. Pode ter certeza que nunca esquecerei de todas nossas conversas e brincadeiras, as quais certamente nos ajudaram a amenizar nossas dificuldades.

A Deus, agradeço por mais esta conquista, por sempre ter estado presente em todos os momentos da minha vida, sempre iluminando minha mente, guiando meus passos e me dando forças para conquistar meus objetivos.

Lucas de Oliveira

"O único homem que está isento de erros, é aquele que não arrisca acertar."

Albert Einstein

Sumário

Lista de Figuras	xiii
Lista de Tabelas	xvii
Lista de Símbolos Principais	xxi
Glossário	xxii
1 INTRODUÇÃO	1
2 SVH E DIFERENTES TIPOS DE IMAGENS	7
2.1 Introdução	7
2.2 Características do SVH	7
2.3 Tipos de Informações	11
2.4 Imagens e Artefatos Típicos	12
2.5 <i>Wavelets</i> em Processamento de Sinais	13
2.5.1 Imagens Médicas	14
2.5.2 Imagens de Satélite	16
2.5.3 Imagens Naturais	18
2.6 Medidas Objetivas Tradicionais de Avaliação de Imagens	20
2.7 Outras Medidas Objetivas de Avaliação de Imagens	21
2.7.1 SSIM	21
2.7.2 Filtro de Contraste	22
2.8 Comentários	23
3 TRANSFORMADA WAVELET CONTÍNUA (TWC)	25
3.1 Da Série de Fourier à TW	26
3.2 Transformada <i>Wavelet</i> Contínua 1D	27
3.3 Interpretação da TWC por Banco de Filtros	28

3.4	TWC Bidimensional	31
3.5	Série de Coeficientes <i>Wavelets</i> (SCW)	32
3.5.1	<i>Wavelets</i> Diádicas	33
3.5.2	SCW Bidimensional	34
3.6	Comentários	35
4	INTRODUÇÃO AOS PRINCÍPIOS DA TRANSFORMADA WAVELET DISCRETA	37
4.1	Codificação por sub-banda	37
4.2	Introdução ao Algoritmo Rápido para TWD	40
4.3	Comentários	41
5	ANÁLISE MULTIRESOLUÇÃO E PROJETO DA TWD	43
5.1	Propriedades da Multiresolução em $L^2(R)$	44
5.2	Análise Multiresolução Ortogonal	45
5.2.1	O Subespaço Vetorial $\{V_{2^j}\}$ na Análise Multiresolução 1D	45
5.2.2	Implementação da Transformada Multiresolução Ortogonal: Decomposição (Análise)	47
5.2.3	O Subespaço Vetorial $\{W_{2^j}\}$ na Análise Multiresolução 1D	49
5.2.3.1	Relação entre os Filtros	52
5.2.4	Implementação da <i>Wavelet</i> Multiresolução Ortogonal: Reconstrução (Síntese)	54
5.3	Análise Multiresolução Bidimensional	55
5.3.1	Implementação da Transformada <i>Wavelet</i> Bidimensional: Decomposição	56
5.3.2	Representação da Transformada <i>Wavelet</i> Ortogonal Bidimensional	59
5.3.3	Implementação da Transformada <i>Wavelet</i> Ortogonal Bidimensional: Reconstrução	65
5.4	Comentários	67
6	BIORTOGONALIDADE	69
6.1	Teoria de <i>Frames</i>	70
6.2	Análise de Multiresolução para <i>Wavelets</i> Biortogonais	72
6.3	Reconstrução a partir da Representação <i>Wavelet</i> Biortogonal	75
6.4	Comentários	76
7	REGULARIDADE E SELEÇÃO DAS WAVELETS	79
7.1	Seleção das <i>Wavelets</i> e <i>Wavelets</i> Utilizadas	79
7.1.1	<i>Wavelets</i> de Daubechies	80
7.1.2	<i>Wavelet</i> de Symlets	81

SUMÁRIO	xi
7.1.3 <i>Wavelets</i> Biortogonais	82
7.1.3.1 <i>Wavelets Spline</i> Biortogonais Cohen - Daubechies - Feauveau . . .	83
7.1.3.2 Variante de <i>Wavelets Spline</i> Biortogonais	84
7.1.3.3 <i>Wavelets</i> Próximas às Ortonormais	85
7.2 Principais Diferenças entre <i>Wavelets</i> Ortogonais e Biortogonais	85
7.3 Tipos de <i>Wavelets</i>	85
7.4 Comentários	86
8 LIFTING	87
8.1 O Método <i>Lifting</i>	88
8.2 Decomposição da <i>Wavelet</i> de Haar Utilizando <i>Lifting</i>	89
8.3 Implementação do Esquema <i>Lifting</i> : Decomposição e Reconstrução	91
8.4 Transformada <i>Wavelet</i> e o Esquema <i>Lifting</i>	92
8.5 O Algoritmo Euclidiano e a Fatoração	97
8.6 Implementação da Decomposição e Reconstrução	98
8.7 Complexidade Computacional	100
8.8 Comentários	101
9 RESULTADOS EXPERIMENTAIS	103
9.1 Observações sobre as Simulações	104
9.2 Resultados com Análise Objetiva	106
9.2.1 Implementação via Ferramenta Compilada	107
9.2.2 Implementação via Ferramenta Interpretada	120
9.3 Reconstrução Perfeita	133
9.4 Observações sobre Ferramenta Compilada x Ferramenta Interpretada	135
9.5 Sumário da Análise de Resultados	136
9.6 Comentários	138
10 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	139
10.1 Comentários sobre o Contexto	139
10.2 Breves Conclusões sobre os Capítulos e Relação com a Proposta	140
10.3 Comentários sobre Contribuições, Simulações e Resultados	141
10.4 Sugestões para Trabalhos Futuros	143
Apêndice A	151
Apêndice B	157

Lista de Figuras

2.1	Esquema simplificado da seção reta horizontal do olho humano [8].	8
2.2	Distribuição de cones e bastonetes do olho [8].	9
2.3	Resposta da sensibilidade de cones e bastonetes (rods) [11].	10
2.4	Resposta referente à sensibilidade dos cones [8].	11
2.5	Tipos de imagens médicas.	15
2.6	Esquematização do processo de sensoriamento remoto [34].	17
2.7	Imagem Lena (256x256 <i>pixels</i>).	20
2.8	Função de sensibilidade à contraste [42].	23
3.1	Tipos de transformadas (a) Transformada de Fourier Janelada (b) Transformada <i>Wavelet</i> [43].	27
3.2	Interpretação da TWC por analogia de um banco de filtros [3].	29
3.3	Análise espectral com banco de filtros [43].	30
3.4	Discretização deslocamento-escala para <i>wavelets</i> diádicas [3].	33
4.1	Decomposição em mais de duas sub-bandas.	39
4.2	Decomposição da imagem em diferentes escalas.	39
4.3	Algoritmo de Mallat (TWD direta) [24].	40
4.4	Algoritmo de Mallat (TWD inversa) [24].	41
5.1	Esquema ilustrativo da relação entre $\phi_{j,n}(x)$ e V_{2^j} para $j \in [-1, 1]$ [3].	46
5.2	Diagrama em blocos do esquema de decomposição [24].	47
5.3	Espaço vetorial W_{2^j}	50
5.4	Diagrama em blocos do esquema de reconstrução da representação <i>wavelet</i> [24].	55
5.5	Diagrama de blocos de um estágio de decomposição <i>wavelet</i> de uma imagem [24].	58
5.6	Imagens resultantes da decomposição <i>wavelet</i> ortogonal bidimensional para 3 níveis [24].	61
5.7	Imagem original Mandrill.	62

5.8	Representação <i>wavelet</i> ortogonal bidimensional para $j=3$	62
5.9	Diagrama em blocos de um estágio da reconstrução da imagem [24].	66
6.1	Diagrama em blocos do algoritmo de decomposição e síntese da TWD biortogonal [24].	76
7.1	<i>Wavelets</i> de Daubechies para: (a) $N=1$; (b) $N=2$; (c) $N=3$; (d) $N=4$; (e) $N=5$; (f) $N=6$; (g) $N=7$; (h) $N=8$	81
7.2	<i>Wavelets</i> de Symlets para: (a) $N=2$; (b) $N=3$; (c) $N=4$; (d) $N=5$; (e) $N=6$; (f) $N=7$; (g) $N=8$.	82
7.3	<i>Wavelets</i> Biortogonais $\psi(x)$ e suas duais $\tilde{\psi}(x)$: (a)bior1.5; (b)bior2.8; (c)bior4.4; (d)bior6.8.	83
8.1	Decomposição do esquema <i>lifting</i> [79].	88
8.2	Método <i>lifting</i> representado por três bandas [6].	89
8.3	Implementação do esquema <i>lifting</i> da transformada de Haar [81].	90
8.4	TWD em 3 níveis ($j=3$) [82].	91
8.5	Reconstrução do sinal em 3 níveis [82].	92
8.6	TWD direta e inversa do sinal [1].	93
8.7	Representação polifásica da transformada <i>wavelet</i> [1].	94
8.8	Esquema <i>primal lifting</i>	96
8.9	Esquema <i>dual lifting</i>	96
8.10	Ganho aritmético alcançado pelo método <i>lifting</i> [2].	101
9.1	Imagens testadas (256x256 <i>pixels</i>): (a) Bird; (b) Bridge; (c) Lena; (d) Goldhill; (e) Crosses.	105
9.2	Imagens testadas (512x512 <i>pixels</i>): (a) Barb ; (b) Boat ; (c) Goldhill ; (d) Mandrill; (e) Lena ; (f) Washsat ; (g) Peppers ; (h) Zelda.	105
9.3	Comparação $T_p \times n$ para <i>wavelets</i> Daubechies utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	107
9.4	Comparação $T_p \times n$ para <i>wavelets</i> Symlets utilizando ferramenta compilada: (a) De- composição; (b) Decomposição e reconstrução.	109
9.5	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem 1) utilizando ferramenta com- pilada: (a) Decomposição; (b) Decomposição e reconstrução.	110
9.6	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem 2) utilizando ferramenta com- pilada: (a) Decomposição; (b) Decomposição e reconstrução.	111
9.7	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem 3) utilizando ferramenta com- pilada: (a) Decomposição; (b) Decomposição e reconstrução.	111

9.8	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	112
9.9	Comparação $T_p \times n$ para <i>wavelets</i> de Daubechies utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	114
9.10	Comparação $T_p \times n$ para <i>wavelets</i> Symlets utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	116
9.11	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem1) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	117
9.12	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem2) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	118
9.13	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem3) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	118
9.14	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.	119
9.15	Comparação $T_p \times n$ para <i>wavelets</i> de Daubechies utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	121
9.16	Comparação $T_p \times n$ para <i>wavelets</i> Symlets utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	122
9.17	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem1) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	124
9.18	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem2) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	124
9.19	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	125
9.20	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	125
9.21	Comparação $T_p \times n$ para <i>wavelets</i> de Daubechies utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e Reconstrução.	127
9.22	Comparação $T_p \times n$ para <i>wavelets</i> Symlets utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e Reconstrução.	128
9.23	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem1) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	130
9.24	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem2) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	130

9.25	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	131
9.26	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.	131
9.27	Reconstrução obtida pelo método <i>lifting</i> (Goldhill - 256x256): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.	133
9.28	Reconstrução obtida pelo método <i>lifting</i> (Bird - 256x256): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.	134
9.29	Reconstrução obtida pelo método <i>lifting</i> (Lena - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.	134
9.30	Reconstrução obtida pelo método <i>lifting</i> (Boat - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.	134
9.31	Reconstrução obtida pelo método <i>lifting</i> (Boat - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.	135
9.32	T_p obtido por diferentes versões do Matlab.	136
10.1	Comparação $T_p \times n$ para <i>wavelets</i> de Daubechies para a imagem Crosses: (a) Decomposição; (b) Decomposição e Reconstrução.	152
10.2	Comparação $T_p \times n$ para <i>wavelets</i> Symlets para a imagem Crosses: (a) Decomposição; (b) Decomposição e Reconstrução.	153
10.3	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem1) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.	153
10.4	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem2) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.	154
10.5	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais (ordem3) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.	154
10.6	Comparação $T_p \times n$ para <i>wavelets</i> Biortogonais para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.	155

Lista de Tabelas

5.1	Distribuição de energia dos coeficientes para Imagem Bird 256x256.	63
5.2	<i>Wavelets</i> de Daubechies: energia dos coeficientes para as imagens 256x256 testadas.	63
5.3	<i>Wavelets</i> de Symlets: energia dos coeficientes para as imagens 256x256 testadas.	64
5.4	<i>Wavelets</i> Biortogonais: energia dos coeficientes para as imagens 256x256 testadas.	64
5.5	<i>Wavelets</i> de Daubechies: energia dos coeficientes para as imagens 512x512 testadas.	64
5.6	<i>Wavelets</i> de Symlets: energia dos coeficientes para as imagens 512x512 testadas.	64
5.7	<i>Wavelets</i> Biortogonais: energia dos coeficientes para as imagens 512x512 testadas.	65
7.1	Coeficientes dos filtros das <i>wavelets</i> de Daubechies.	81
7.2	Coeficientes dos filtros das <i>wavelets</i> de Symlets.	82
7.3	Coeficientes dos filtros <i>spline</i> $l = 3$ e $\tilde{q} = 2$	84
7.4	Coeficientes dos filtros variante <i>spline</i> com comprimentos semelhantes $l = q = 4$	84
7.5	Coeficientes dos filtros variante <i>spline</i> com comprimentos muito semelhantes $l = q = 2$	85
8.1	Esquematização dos parâmetros na decomposição por esquema <i>lifting</i> [82].	92
9.1	Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> de Daubechies.	108
9.2	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> de Daubechies.	108
9.3	Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> Symlets.	109
9.4	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> Symlets.	109
9.5	Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> Biortogonais.	113
9.6	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> Biortogonais.	113

9.7	Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> Daubechies.	115
9.8	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> Daubechies.	115
9.9	Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> Symlets.	116
9.10	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> Symlets.	117
9.11	Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> Biortogonais.	119
9.12	Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> Biortogonais.	120
9.13	Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> de Daubechies.	121
9.14	Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> de Daubechies.	122
9.15	Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> Symlets	123
9.16	Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> Symlets	123
9.17	Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando <i>wavelets</i> Biortogonais.	126
9.19	Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando <i>wavelets</i> Biortogonais.	126
9.20	Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> de Daubechies.	128
9.21	Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> de Daubechies.	128
9.22	Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> Symlets.	129
9.23	Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> Symlets.	129
9.24	Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando <i>wavelets</i> Biortogonais	132

9.25 Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando <i>wavelets</i> Biortogonais	132
9.26 Resultados médios e máximos de tempo de processamento em linguagem compilada para diferentes famílias de <i>wavelets</i>	137
10.1 <i>Wavelets</i> de Daubechies: energia dos coeficientes para as imagens 256x256 testadas.	151
10.2 <i>Wavelets</i> de Symlets: energia dos coeficientes para as imagens 256x256 testadas. . .	151
10.3 <i>Wavelets</i> Biortogonais: energia dos coeficientes para as imagens 256x256 testadas. .	152
10.4 Coeficientes dos filtros utilizados nas implementações.	162

Lista de Símbolos Principais

$\psi(x)$ – *Wavelet* mãe ou *wavelet* básica.

$\phi(x)$ – Função escala.

$\psi_{j,n}(x)$ – *Wavelets* filhotes, versões deslocadas e dilatadas ou comprimidas da *wavelet*-mãe.

$\phi_{j,n}(x)$ – Versões deslocadas e dilatadas ou comprimidas da função-escala.

Glossário

AVC	– <i>Advanced Video Codec</i>
BMP	– <i>Bitmap</i>
CSF	– <i>Função de Sensibilidade ao Contraste - Contrast Sensitivity Function</i>
DICOM	– <i>Digital Imaging and Communicating in Medicine</i>
DVB-H	– <i>Digital Video Broadcasting - Handheld</i>
ECW	– <i>Enhanced Compression Wavelet</i>
EZW	– <i>Embedded Zerotree Wavelet Coding</i>
GIF	– <i>Graphics Interchange Format</i>
GPS	– <i>Sistema de Posicionamento Global - Global Position System</i>
SVH	– <i>Sistema Visual Humano - Human Visual System</i>
HDTV	– <i>Televisão de Alta Definição - High Definition Television</i>
ISDB	– <i>Integrated Services Digital Broadcasting</i>
JPEG	– <i>Joint Photographic Expert Groups</i>
MrSID	– <i>Multiresolution Seamless Image Database</i>
MSE	– <i>Erro Médio Quadrático - Mean Square Error</i>
PNG	– <i>Portable Network Graphics</i>
PSD	– <i>Photoshop Document</i>
PDI	– <i>Processamento Digital de Imagem - Digital Image Processing</i>
PSNR	– <i>Razão Sinal Ruído de Pico - Peak Signal Noise Ratio</i>
RAW	– <i>Informação não processada</i>
SBTVD	– <i>Sistema Brasileiro de Televisão - Brazilian Digital Television System</i>
SSIM	– <i>Método da Similaridade Estrutural - Structural Similarity</i>
SPIHT	– <i>Set Partitioning in Hierarchical Trees</i>
TIFF	– <i>Tagged Image File Format</i>
TF	– <i>Transformada de Fourier - Fourier Transform</i>
TW	– <i>Transformada Wavelet - Wavelet Transform</i>
TWC	– <i>Transformada Wavelet Contínua - Continuous Wavelet Transform</i>
TWD	– <i>Transformada Wavelet Discreta - Discrete Wavelet Transform</i>
VQEG	– <i>Video Quality Experts Group</i>

Capítulo 1

INTRODUÇÃO

A área de Processamento Digital de Imagens está evoluindo continuamente, despertando considerável interesse no que se refere à compressão de imagens, morfologia matemática, reconhecimento de imagens, redes neurais, sistemas de análise baseados em conhecimento, dentre outros. Com o intuito de solucionar problemas, em especial, referente à compressão, detecção de bordas e análise de textura, muitos pesquisadores mantêm-se interessados em novas técnicas, transformadas e combinações de diferentes tecnologias em busca de tal melhoramento, beneficiando áreas tais como comunicação via satélite, tratamento de imagens geológicas, radares e sonares, teleconferência, imagens e vídeo via celular, cinema digital, HDTV (TV de Alta Definição - *High Definition Television*), dentre outras, seja para aplicações de pesquisa, mercado ou lazer.

Apesar das grandes vantagens, como maior robustez a erros de transmissão e facilidade de manuseio, a imagem no formato digital implica em um maior volume de dados, exigindo uma maior capacidade de armazenamento e de transmissão. Conseqüentemente, o grande desafio da utilização de imagens digitais é a compressão/codificação da grande quantidade de bits necessários para sua representação e sua velocidade de processamento. Neste contexto, para evitar o estrangulamento dos meios físicos existentes, pesquisas contínuas em busca de novos procedimentos ou na melhoria dos já existentes se tornam mandatórios, onde o desenvolvimento de algoritmos capazes de comprimir esta massa de informação facilitando seu armazenamento, transmissão e processamento destes dados estão em forte ascensão.

As formas tradicionais de compressão de imagens baseiam-se, em sua maioria, na técnica de codificação por transformada. Esta consiste na aplicação de transformações lineares em um sinal, produzindo uma representação mais compacta do mesmo em um domínio diferente. Entre as transformadas mais utilizadas, destaca-se a TWD (Transformada *Wavelet* Discreta - *Discret Wavelet Transform*), fruto de inúmeras pesquisas das mais diversas áreas, tais como a matemática, a engenharia, a estatística, a física, a computação gráfica, dentre outras.

Assim como a TF (Transformada de Fourier - *Fourier Transform*), a TW (Transformada *Wavelet* - *Wavelet Transform*) decompõe o sinal em uma base de funções e, dentre as inúmeras características que apresenta, pode-se destacar que a mesma capta a informação da localização tempo-frequência do sinal, apresenta algoritmos rápidos, boa capacidade de concentrar a energia do sinal em poucos coeficientes, transmissão progressiva, além de não introduzir artefatos de blocagem. Essas vantajosas características permitiram a inserção da TW em importantes padrões recentes de compressão de imagem e vídeo, tais como o JPEG2000, MPEG-4 e MPEG-7 [3].

A TWD tem sido tradicionalmente implementada por meio de esquema de bancos de filtros [4]. Entretanto, o método *lifting* realiza a TWD de uma maneira diferente, sub-amostrando o sinal de entrada antes do processo de filtragem, provendo um rápido processamento, facilitando e acelerando a implementação e conseqüentemente, o cálculo dos coeficientes na TWD [1] [2]. Por este fato, seu desempenho tornou-se atrativo e conseqüentemente, pesquisas foram motivadas a investigar no quanto pode-se representar este desempenho.

O ganho proporcionado tomou proporções distintas na literatura: Daubechies e Sweldens [1] estimam ganho de até duas vezes para os maiores comprimentos de filtro. Reichel [2] vai além, afirmando que este ganho pode chegar a até quatro vezes. Diversas tentativas recentes foram realizadas com o intuito de analisar o estudo do impacto das técnicas de *lifting* [5] [6] [7].

Contribuições e Organização deste Trabalho

É objeto desta dissertação dar seqüência à linha de estudos teóricos apresentados por Daubechies e Sweldens [1] e Reichel [2], analisando na prática em que proporção o desempenho oferecido pelo método *lifting* é superior ao método tradicional, em termos de tempo de processamento, via ferramenta interpretada e compilada, apresentando simulações e análises comparativas para diversas imagens-teste, resoluções e famílias de *wavelets*.

Várias *wavelets* foram empregadas dentre três famílias distintas e conseqüentemente, comprimentos de filtros apresentando diferentes tamanhos foram obtidos. Como apresentado em [1] e [2], conforme os filtros apresentem maior comprimento, o método *lifting* deve apresentar ganhos mais significativos, justamente por fazer uso da sub-amostragem do sinal antes do processo de filtragem, provendo um rápido processamento, facilitando e acelerando a implementação e conseqüentemente, o cálculo dos coeficientes.

Como critério de análise dos resultados, foi considerado o tempo de processamento necessário para executar apenas a decomposição do sinal e uma outra análise para todo o processo (decomposição e reconstrução). Medidas de PSNR (Razão Sinal Ruído de Pico - *Peak Signal Noise Ratio*) foram igualmente utilizadas de forma a corroborar a perfeita implementação do *lifting*, dado que o mesmo

não pode implicar em deterioração na imagem reconstruída, sendo obtido valor infinito para esta medida, implicando assim em uma imagem idêntica a original. Além desta medida estar coerente, pode ser realizada uma breve análise subjetiva nas imagens obtidas, comprovando assim a acuracidade da mesma.

Objetivando o embasamento teórico dos principais conceitos empregados nesta pesquisa, os mesmos são demarcados da seguinte maneira:

Capítulo 2: SVH (Sistema Visual Humano - *Human Visual System*) e Diferentes Tipos de Imagens

Neste capítulo é realizado um apanhado bibliográfico sobre as principais características referente ao SVH (Sistema Visual Humano - *Human Visual System*) assim como a importância das *wavelets* utilizadas em processamento de sinais. Como o campo de estudo de imagens é bastante vasto para análise de imagens médicas, de satélite e imagens naturais, as principais características e o estado da arte são abordados, sendo apresentada a importância da TW nestas aplicações. Medidas objetivas tradicionais e novas medidas objetivas finalizam a apresentação deste capítulo.

Capítulo 3: Transformada *Wavelet* Contínua (TWC)

Apresenta-se neste capítulo os princípios básicos da TWC, englobando as principais características e princípios de funcionamento para o caso 1D e 2D. Conceito de grande importância abordado neste capítulo refere-se à análise da TW por intermédio da utilização de bancos de filtros.

Capítulo 4: Introdução aos Princípios da Transformada *Wavelet* Discreta (TWD)

Os sinais resultantes da decomposição *wavelet* englobam o sinal passa-baixa e os sinais de detalhes, os quais são enfocados nos próximos capítulos. Visando melhor compreensão, torna-se necessário neste capítulo realizar um breve estudo sobre a codificação por sub-banda, a qual resulta na implementação prática da TWD através dos filtros do algoritmo de Mallat.

Capítulo 5: Análise Multiresolução e Projeto da TWD

A decomposição e síntese da TWD para o caso ortogonal, tanto para a circunstância unidimensional quanto bidimensional, assim como os sinais passa-baixas e de detalhes resultantes do processo de decomposição são analisados. Para contornar as questões conflitantes impostas pela ortogonalidade, surgiu a biortogonalidade, a qual possibilitou o uso da TWD na compressão de imagens com excelente desempenho, sendo melhor detalhada no Capítulo 6.

Capítulo 6: Biortogonalidade

O mesmo raciocínio já utilizado na decomposição e síntese será mantido, porém no capítulo em questão será abordado para o caso biortogonal. Dentro desse contexto, os filtros biortogonais possibilitam suporte compacto, simetria e regularidade, conforme visto no Capítulo 7, requisitos estes essenciais para se obter um ótimo desempenho para a compressão de imagens, quando estendido para o caso bidimensional.

Capítulo 7: Regularidade e Seleção das *Wavelets*

Neste capítulo são apresentadas as características desejáveis para a TWD na compressão de imagens, em que uma breve descrição das principais diferenças entre *wavelets* ortogonais e biortogonais é realizada, assim como um breve apanhado dos diversos tipos de *wavelets* geralmente encontrados na literatura. O foco do capítulo é voltado com destaque para a regularidade, culminando com as principais características das *wavelets* utilizadas.

Capítulo 8: *Lifting*

Embora o esquema *lifting* e o esquema por banco de filtros resultem no mesmo resultado no que diz respeito à realização da TWD, o custo computacional é diferente. Objetivando-se analisar tal diferença no capítulo seguinte, torna-se essencial neste capítulo realizar uma breve introdução aos principais conceitos desta técnica, tais como a transformada direta e inversa, assim como decomposição e reconstrução do sinal. O capítulo é finalizado apresentando, brevemente, alguns valores dos ganhos teóricos obtidos por Daubechies e Sweldens e também por Reichel.

Capítulo 9: Resultados Experimentais

Este capítulo concentra os resultados fundamentais da presente dissertação analisando na prática em que proporção o desempenho oferecido pelo método *lifting* é superior ao método tradicional, em termos de tempo de processamento, via ferramenta interpretada e ferramenta compilada, dando seqüência à linha de estudos teóricos apresentados por Daubechies e Sweldens [1] e Reichel [2]. Medidas de PSNR foram igualmente utilizadas de forma a corroborar a perfeita implementação do *lifting*, dado que o mesmo não pode implicar em deterioração na imagem reconstruída.

Neste sentido, os resultados foram analisados via ferramenta interpretada e compilada, buscando a que oferece melhor interpretação para os resultados obtidos. Percebe-se que ambas oferecem reconstrução perfeita, através da obtenção de um PSNR infinito.

Observações sobre condições e particularidades de simulações; e análises e conclusões acerca dos resultados obtidos finalizam o capítulo.

Capítulo 10: Conclusões e Sugestões para Trabalhos Futuros

A partir da obtenção dos resultados, finaliza-se o trabalho com os devidos comentários e conclusões, além de sugerir possíveis alternativas encontradas durante a realização da pesquisa.

Capítulo 2

SVH E DIFERENTES TIPOS DE IMAGENS

2.1 Introdução

O fundamento da compressão de imagens está relacionado, basicamente, no conceito de informação visual da imagem. Vale ressaltar que, esse conceito não está ligado a Teoria da Informação, mas sim a um conceito subjetivo de qualidade de imagem dado pelo observador ou pelo sistema, ao qual se destina a imagem.

Em projetos envolvendo análises de imagens, o usuário final nem sempre é o ser humano. Em diversas aplicações os computadores fazem as vezes de usuários finais, exigindo por parte dos algoritmos o conhecimento das características importantes das imagens sob o ponto de vista do processamento computacional. Mas na grande maioria das aplicações, o observador humano é responsável por avaliar a qualidade dos processos de compressão e descompressão de imagens, tendo como foco, a preservação do aspecto visual. Neste contexto, o sistema destino SVH processa características diferentes do sinal em diferentes estágios e com diferentes sensibilidades.

Neste capítulo, será realizada uma breve abordagem sobre o mecanismo da visão humana, buscando focar brevemente suas principais características. Além disso, é abordada a importância das *wavelets* em processamento de sinais, em seu contexto atual, nas mais diversas aplicações. Finalizando, são apresentadas as medidas objetivas clássicas, as quais foram utilizadas na avaliação das imagens em questão e, ainda, um breve comentário que se destina a novas medidas objetivas que começam a se destacar para fins de documentação do tema.

2.2 Características do SVH

Dentro do objetivo proposto, o foco é voltado para o ser humano como usuário final, a quem compete julgar a qualidade final do sinal. Deste modo, torna-se necessário um breve conhecimento

do mecanismo da visão humana, permitindo definir aos algoritmos como concentrar as perdas em regiões de percepção que apresentem menor sensibilidade. Perdas podem ser aceitáveis dependendo da aplicação em que a imagem está submetida.

A visão humana, por sua vez, é composta basicamente por órgãos receptores (olhos), uma unidade de processamento (cérebro) e por nervos que transmitem informação dos olhos para o cérebro. A Fig. 2.1 ilustra a seção horizontal do olho humano.

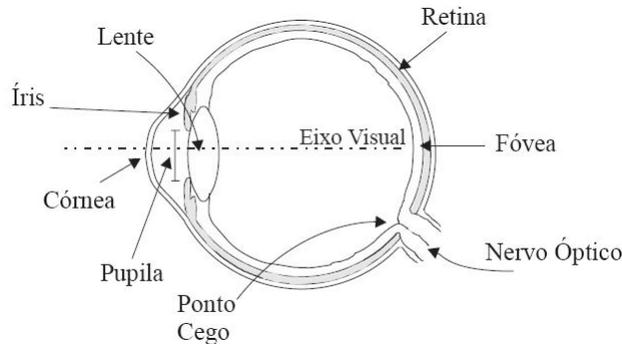


Figura 2.1: Esquema simplificado da seção reta horizontal do olho humano [8].

O olho é o órgão responsável por traduzir estímulos luminosos em impulsos nervosos. A luz atinge inicialmente a córnea, que age como uma lente focalizadora da luz incidente, e em seguida, passa por um líquido transparente, denominado humor aquoso [8]. Uma parcela da luz incidente é barrada pela íris. A íris é um anel de musculatura lisa, cujo círculo interior, chamado de pupila, pode aumentar ou diminuir de tamanho, controlando assim a quantidade de luz que efetivamente entra no interior do olho.

A luz que passa pela pupila chega a uma lente (cristalino), responsável pela focalização dinâmica do olho. O cristalino é uma estrutura biconvexa, elástica e transparente. Através da ação dos músculos ciliares, seu raio de curvatura pode ser alterado, possibilitando a focalização de objetos situados a distâncias variadas. Quando um objeto distante é observado, a lente se alonga e seu poder de convergência diminui, focalizando adequadamente o objeto. O contrário acontece quando um objeto próximo é observado. Após passar pela lente, a luz atravessa outro meio transparente: o humor vítreo.

Por fim a luz chega a retina, que por sua vez, é uma membrana ocular interna que corresponde à superfície receptora que capta os sinais luminosos e os transforma em impulsos nervosos, sendo esta, composta por dois tipos de receptores: bastonetes e cones. Por curiosidade, um olho contém cerca de 6.5 milhões de cones e 100 milhões de bastonetes distribuídos sobre a retina [9].

Os bastonetes existem em maior número e apresentam maior sensibilidade que os cones em condições de baixa intensidade luminosa, onde sua resposta visual é chamada de visão escotópica [8]. Além do que, os bastonetes não são capazes de detectar cores [10].

Os cones existem em menor quantidade no olho e não são distribuídos de forma uniforme na retina, concentrando-se na região central do olho, a *fovea centralis*. De fato, esta é a região mais sensível a cores e menos sensível a baixas iluminações do olho humano. A Fig. 2.2 ilustra a distribuição de cones e bastonetes na retina.

Apesar do maior número de bastonetes, o Sistema Visual Humano é capaz de discernir apenas dezenas de níveis de cinza em comparação com os milhares de tons e intensidades de cores, em outras palavras, o olho humano é mais sensível às variações de cores do que às variações de tons de cinza, sendo que na observação de imagens monocromáticas, os olhos respondem apenas as diferenças de brilho.

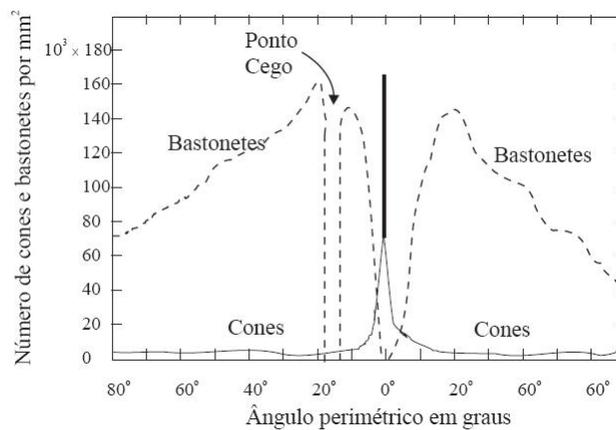


Figura 2.2: Distribuição de cones e bastonetes do olho [8].

Os cones necessitam de mais intensidade luminosa que os bastonetes para apresentar resposta, sendo conhecida como visão fotópica. No entanto, eles são responsáveis pela visão de cores e nitidez de imagens e a grande maioria das aplicações de imagens encontra-se na faixa de visão fotópica [10]. A fim de complementar o assunto, a Fig. 2.3 mostra a sensibilidade de cones e bastonetes segundo Wald [11].

Na vizinhança do nervo óptico não há cones e bastonetes, sendo portanto desprovido de visão e conseqüentemente, possuindo um ponto cego nessa região [8]. Esta região é totalmente ocupada pelo feixe de fibras, sendo que desta forma os objetos de uma cena que são focalizados no ponto cego não são vistos efetivamente [10]. Após os impulsos nervosos serem gerados, estes são conduzidos pelo feixe de fibras, que compõem os nervos ópticos, até o sistema nervoso central. A informação que segue pelo nervo óptico percorre um caminho até chegar ao córtex visual, que corresponde à região do cérebro diretamente relacionada à visão.

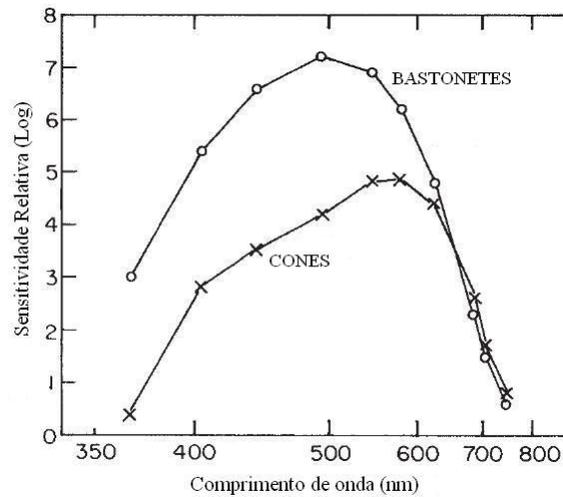


Figura 2.3: Resposta da sensibilidade de cones e bastonetes (rods) [11].

Segundo Resnikoff em [12], o fato de não perceber um ponto de cegueira em nosso campo visual parece contradizer a existência do ponto cego. Na verdade, a visão humana completa a informação luminosa correspondente ao ponto cego com a informação de regiões vizinhas, onde maiores detalhes sobre este processo de completamento e comprovações da existência do ponto cego podem ser vistas em [12].

A literatura [13]-[15] relata que as cores são percebidas por três tipos de células cones para a recepção de estímulos luminosos na retina, gerando três componentes de cor para a informação visual. Esses cones possuem características de absorção diferentes e podem ser caracterizados em três tipos: L-cones, M-cones e S-cones, já que eles são sensíveis a comprimentos de onda longos, médios e pequenos, respectivamente. A Fig. 2.4 ilustra suas respectivas curvas de absorção espectral onde cada um responde melhor a uma determinada faixa de frequências da luz, podendo ser observado que os cones do tipo M possuem melhor absorção.

Pode-se observar que, cones do tipo L referem-se à percepção da luz vermelha, enquanto os do tipo M e S correspondem às percepções de luz verde e azul, respectivamente. Vale destacar a sobreposição das curvas de absorção em praticamente todo espectro de luz visível. Sendo assim, uma cor nunca estimula apenas um tipo de cone, pois a percepção humana de cores é dada pela composição das respostas dos três tipos de cones que atingem os olhos.

Pode-se concluir desta forma, que mesmo que uma fonte de luz azul emita a mesma quantidade de energia luminosa que uma fonte verde, será percebida a luz verde como mais intensa, justamente devido às curvas de absorção apresentadas. Todas as cores são vistas como combinações variadas das três cores primárias: vermelho, verde e azul. Variando a quantidade relativa destas, torna-se possível produzir uma enorme variedade de cores, incluindo diversos tons para cada uma.

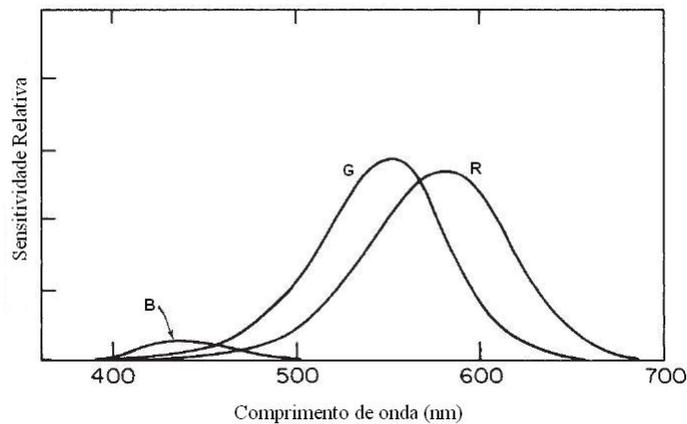


Figura 2.4: Resposta referente à sensibilidade dos cones [8].

Vale ressaltar que, a forma mais comum de representar imagens digitais consiste em considerar três componentes de cores, onde as leis de Grassman contendo seus oito axiomas regem este modelo tridimensional. Maiores detalhes podem ser encontrados em [16].

Uma das componentes é chamada componente de luminância (acromática), que se refere à imagem observada sem cores, em outras palavras, é sua representação monocromática (como as utilizadas nas simulações). As outras duas componentes são as componentes de crominância que trazem a informação adicional necessária para a percepção humana de cores.

2.3 Tipos de Informações

Uma das principais considerações a ser feita no processo de visualização é a determinação de qual técnica deve ser empregada em uma determinada aplicação ou situação. Esta escolha é certamente dependente do tipo de informação que está sendo tratada e das tarefas que precisam ser realizadas pelo usuário.

A informação visual pode ser dividida em duas classes distintas [17]: informação supérflua e informação necessária. A informação supérflua refere-se à informação da imagem que se retirada da imagem original provoca pouca ou nenhuma mudança na imagem, sendo que esta pode ser utilizada ou bem avaliada pelo sistema-destino (SVH). Em contrapartida, se a informação necessária for retirada, mesmo que seja parcialmente da imagem original, provoca grandes deteriorações na qualidade subjetiva da imagem dados os parâmetros de percepção e tolerância do SVH.

No que diz respeito à informação supérflua, esta pode ser dividida em duas classes distintas segundo a sua origem [17]:

- **Redundância Estatística:** informação que está relacionada à correlação e à interdependência

dos dados. Com a remoção desta informação, não se tem deterioração da qualidade da imagem, pois ela pode ser completamente predita a partir do restante da imagem.

- **Informação irrelevante:** informação relacionada às características que podem ser removidas causando apenas perdas toleráveis para o sistema que receberá a imagem. Como esse tipo de informação não pode ser recuperado após sua eliminação, deve-se realizar a eliminação com critério de modo evitar perdas significativas na qualidade da imagem.

Desta forma, as técnicas de compressão são classificadas de acordo com o modo com que a informação é explorada, resultando em compressão com perdas e compressão sem perdas.

A compressão sem perdas não provoca deterioração da imagem, pois explora somente a redundância estatística dos dados, apresentando a vantagem de não introduzir erros e conseqüentemente, apresenta baixa taxa de compressão.

Por sua vez, a imagem reconstruída por intermédio da compressão com perdas contém degradações em relação à original, sendo explorada a redundância subjetiva. Isto implica, conseqüentemente, em um nível de compressão muito maior, fato este que, torna este tipo de compressão bastante atrativa, já que tais degradações podem ou não serem percebidas.

Sendo assim, cada aplicação explora uma determinada característica da imagem, manipulando-a de forma a alcançar a melhor taxa de compressão possível, com uma perda de qualidade tolerável e que seja aceitável para o SVH.

2.4 Imagens e Artefatos Típicos

Para conseguir redução da taxa, podem ser exploradas tanto as propriedades psicovisuais do sistema visual humano como a redundância estatística dos dados da imagem em sistemas de compressão.

Na compressão, por sua vez, algumas degradações podem ser introduzidas através de filtragem, conversão, transformação, quantização e/ou transmissão. Dentre os indesejados efeitos do processo de compressão, tem-se perda de resolução, efeito de bloco, ruído de quantização, erros de bloco, *blurring* e *ringing*, estes dois últimos artefatos típicos da compressão *wavelet*.

Tais degradações são influenciadas pelo algoritmo de compressão utilizado, conteúdo da imagem, origem do material fonte e condições do canal de transmissão [18], podendo-se destacar os seguintes artefatos:

- **Blocagem:** efeito de bloco ou blocagem possui uma deterioração na qual a imagem reconstruída apresenta padrões retangulares que não estavam presentes na imagem original. Normalmente

esta degradação está presente quando elevadas taxas de compressão são necessárias (quantização pouco refinada) e as descontinuidades entre os blocos da imagem processadas se tornam visíveis.

- Edge Business: degradação esta, que resulta no aparecimento de uma distorção concentrada nas bordas dos objetos, caracterizada pela variação temporal ou espacial da borda, causada geralmente, pela quantização de um bloco contendo *pixels* que delimitam áreas planas com diferença significativa entre os níveis médios em cada uma das bordas.
- Blurring: esta degradação refere-se à redução da definição em bordas e em áreas com muitos detalhes especiais. Como se sabe, os componentes de alta frequência têm, usualmente, menos energia e são menos perceptíveis ao olho humano, tornando possível assim o descarte desta informação para que se atinja a taxa desejada. A perda de resolução é uma degradação caracterizada pela perda, ou quantização pouco refinada, dos componentes de alta frequência. Na prática, em uma dada situação de taxa, há um compromisso entre alta resolução e baixo ruído, sendo este ajuste subjetivo e dependente da aplicação.
- Ruído: ruídos podem ser causados por erros na transmissão de dados, na aquisição da imagem na hora de sua digitalização, adicionados propositalmente à imagem para produzir algum efeito especial, ou artificialmente por procedimentos para o estudo de técnicas de eliminação dos mesmos. Os *pixels* corrompidos pelo ruído, têm alguns bits alterados, causando assim, uma diferença brusca de tons com seus vizinhos. Vale lembrar que a quantização fundamental no processo de compressão produz o tradicional ruído de quantização que tende a ser descorrelacionado do sinal e não uniformemente distribuído ao longo da imagem.

2.5 Wavelets em Processamento de Sinais

Wavelets têm sido amplamente exploradas em um grande número de aplicações em processamento de sinais, merecendo destaque para compressão de sinais atingindo altas taxas de compressão quando combinadas a esquemas, tais como, EZW (*Embedded Zerotree Wavelet Coding*) ou SPIHT (*Set Partitioning in Hierarchical Trees*) [3]. Estudos mais recentes também têm sido realizados para otimizar a TWD [19] e para realizar a compressão *wavelet* de imagens estáticas explorando características geométricas das imagens, obtendo melhores resultados em termos de qualidade visual com menor complexidade [20]. Codificadores de vídeo tais como codificadores Rududu (que utiliza *Libdwic Directional Wavelet Image Compression*), codificador Dirac, Pixlet (codec intraframe para *Apple*), padrão ECW (*Enhanced Compression Wavelet*), MrSID (*Multiresolution Seamless Image Database*)

e formato de vídeo Bink foram também desenvolvidos utilizando esta transformada. Imagens codificadas com *wavelet* também apresentam vantagens ao serem reconstruídas no caso de perdas de pacotes na rede [21] [22].

Talvez as *wavelets* mais populares sejam as *wavelets* de Daubechies, por sua ortogonalidade e suporte compacto. As *wavelets* de Gabor também se mostram populares em aplicações em segmentação de imagens [23]. As ferramentas de implementação variam de aplicação para aplicação, entretanto, percebe-se uma grande utilização de bancos de filtros e de algoritmos piramidais de filtragem para implementação da TW. Em processamento de imagens há versões naturalmente extrapoladas para duas dimensões dos algoritmos *wavelets*, empregando, em especial, os esquemas piramidais baseados em implementações em multiresolução. Em computação gráfica, *wavelets* têm sido usadas para edição de curvas, manipulação de superfícies, análise de textura, compressão de imagens, segmentação, segmentação de imagens texturizadas, quantização, enfoque, magnificação, interpolação, pintura em multiresolução, representação de curvas, análise de superfícies, representação de fluxo de luz, radiosidade, modelamento geométrico, modelos fractais, visão computacional e óptica difrativa, entre outras.

Mallat descreveu o uso de *wavelets* em discriminação de textura e análise fractal em imagens [24]. Em [25] e [26] são retratadas as capacidades seletivas das *wavelets* na edição de curvas em imagens, alterando aspectos globais de contorno sem afetar detalhes na imagem. *Wavelets* têm sido aplicadas para remoção de ruídos, reconhecimento e extração de padrões, e realce de características em sinais. Graps e Vidakovic abordaram o uso de *wavelets* em aplicações de extração de ruídos de sinais de ressonância magnética nuclear [27] e de sinais sísmicos [28]. Têm sido empregadas em interpretação de imagens e sinais biomédicos, detecção de sinais; em interpretações sísmicas e em geofísica; em análise de voz, acústica, análise e síntese, música, representações auditivas, psicoacústica e identificação de padrões sônicos.

Como apresentado, a TWD está presente em diversas áreas da ciência. Como o campo de estudo de imagens é bastante vasto para análise de imagens médicas, de satélites e imagens naturais, faz-se necessário apresentar de forma mais detalhada, a importância da TW nestas aplicações, as quais seguem nas sub-seções seguintes.

2.5.1 Imagens Médicas

A utilização de imagens médicas digitais está presente e em ascensão na maioria dos hospitais e clínicas do mundo, tornando imperativo o desenvolvimento de técnicas que viabilizem o armazenamento e a transmissão destas imagens em redes de computadores.

As vantagens de se ter imagens digitais na área médica são inúmeras, destacando-se a boa resolução; a rapidez para localizá-las e disponibilizá-las para o paciente, permitindo assim a comparação

com outras imagens do mesmo; a facilidade no processo de realce de detalhes como contornos e contraste para diagnósticos; e a não deterioração da qualidade com o tempo.

No que diz respeito à compressão e à descompressão de imagens, as imagens médicas devem revelar como resultado um diagnóstico, documentando com precisão a situação que se encontra o paciente. O objetivo é que não haja perdas de informação após a descompressão, evitando o comprometimento na precisão do diagnóstico. Logo, compressão com perdas deve ser utilizada com muito critério, de modo que se obtenha um diagnóstico compatível com o caso clínico, podendo assim, obter resultados aceitáveis com taxas de compressão de 10:1 ou mais, dependendo da área médica e do tipo de compactação que se está utilizando [29].

A maioria das imagens digitais é destinada a observadores humanos, sendo que as pesquisas atuais estão sendo direcionadas à compressão com perdas, minimizando a distorção visual e possibilitando a obtenção de resultados visualmente iguais a da compressão sem perdas [30]. Porém, deve haver um nível aceitável de compressão com perda, de modo a não afetar o diagnóstico clínico.

O grau de compressão que se pode utilizar em imagens médicas possui grande variação dependendo do tipo de exame: ultra-sonografia, endoscopia, tomografia computadorizada, ressonância magnética, radiografia computadorizada, hemodinâmica, patologia, mamografia, vídeo-cirurgia, radioterapia e demais exames (Fig. 2.5).

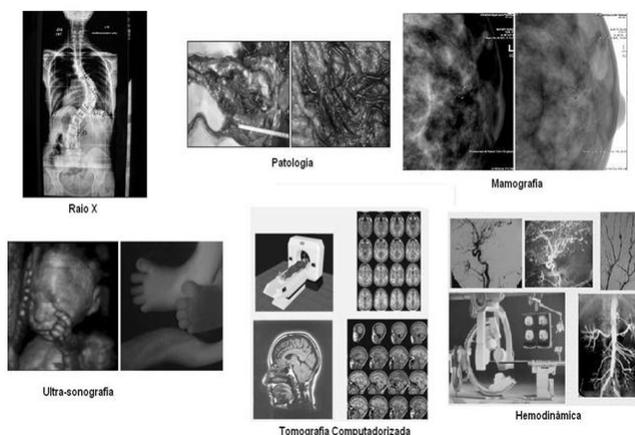


Figura 2.5: Tipos de imagens médicas.

Compressões sem perdas são possíveis com taxa de 2:1 ou 3:1. Compressões com perdas com fator de compressão 10:1 podem ser discutíveis, embora sejam adotadas em hospitais. Estudos mostram uso de compressões de até 40:1 sem que haja perda de informações com relevância clínica [31].

Existem vários formatos de arquivos para armazenar imagens digitais estáticas, podendo ser com ou sem compactação. Dentre os arquivos mais conhecidos, estão: RAW (informação não processada), PSD (*Photoshop Document*), BMP (*Bitmap*), GIF (*Graphics Interchange Format*), PNG (*Portable*

Network Graphics), TIFF (*Tagged Image File Format*), JPEG (*Joint Photographic Expert Group*) e JPEG2000.

Cada formato possui características quanto ao tamanho do arquivo gerado, quantidade de cores, grau de perda de qualidade de imagem, entre outras, tornando cada uma mais adequada para um tipo de aplicação. Caso todos os tipos de arquivos fossem aceitos em imagens médicas, a interoperabilidade dos equipamentos geradores de imagem seria difícil.

Neste contexto, a situação atual de grande parte dos hospitais do mundo, revela que nos diferentes equipamentos hospitalares produtores de imagens, a maioria não possui a capacidade de serem conectados entre si e nem a possibilidade de terem suas imagens visualizadas em computadores pessoais, mostrando dessa forma, a necessidade de obter a compatibilidade e interoperabilidade entre sistemas independentes de informação em saúde [29].

Com a necessidade de padrões internacionais específicos na área médica, o DICOM (*Digital Imaging and Communicating in Medicine*), que se tornou o padrão internacional para interconexão dos equipamentos de imagem na área da saúde e troca de imagens diagnósticas e terapêuticas. As interfaces DICOM estão presentes em quase todos os novos modelos de equipamentos de imagens, tornando-se possível o uso de equipamentos de diferentes fabricantes em uma mesma rede de imagens, desde que exista uma adequada comunicação entre eles [29].

A compressão de imagem utilizando TW no padrão DICOM já são linhas de pesquisas e já mostram resultados superiores ao JPEG, onde maiores detalhes podem ser encontrados em [30].

Inúmeros outros métodos têm sido testados para compactar diferentes tipos de imagens médicas, existindo a possibilidade que no futuro cada tipo de exame tenha um algoritmo de compactação que melhor funcione para ele [32].

2.5.2 Imagens de Satélite

O sensoriamento remoto pode ser definido de uma maneira ampla como a detecção da natureza de um objeto sem que haja contato físico com este, em que aviões e satélites são as plataformas mais comuns. O termo sensoriamento remoto é restrito aos métodos que se utilizam da energia eletromagnética na detecção e medida das características de objetos, incluindo-se aqui as energias relativas à luz, calor e ondas de rádio [33].

A fotografia aérea foi o primeiro sensor remoto a ser utilizado, e ainda é utilizado em mapas topográficos, sendo que esta tem sido usada intensivamente como um sensor remoto na identificação e mapeamento dos recursos naturais. Pelo potencial de utilização, as fotografias aéreas prestam-se como ferramenta de trabalho nos mais diferentes campos, tais como Geografia, Geologia, Hidrologia, Ecologia, Agronomia, Engenharia, entre outros. A Fig. 2.6 ilustra sua esquematização, sendo que as informações obtidas podem apresentar diferentes resoluções espaciais, temporais e espectrais [34].

Desta forma, uma alta resolução espacial é necessária para uma melhor caracterização da forma, estrutura e textura dos objetos, enquanto uma alta resolução espectral permite a discriminação e conseqüente classificação de diferentes regiões em uma imagem. Quanto maior a resolução da imagem, maior será seu tamanho em bits, conseqüentemente, maior a largura de banda de transmissão para essa informação, logo, maior o custo. Como se sabe, a alta compressão priorizaria a economia de banda de transmissão, porém, a informação obtida deve ser analisada para ver se está dentro das exigências de sua respectiva aplicação.

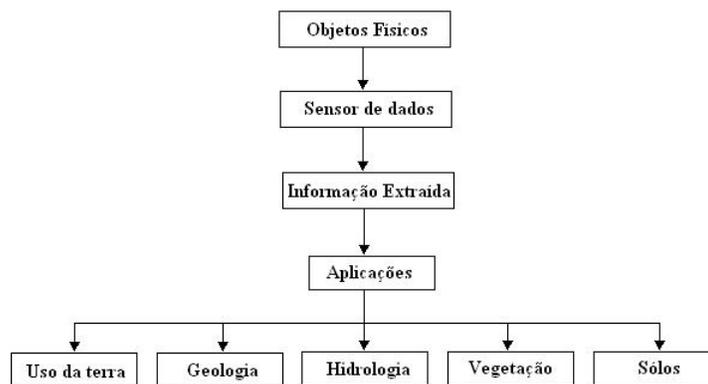


Figura 2.6: Esquemática do processo de sensoriamento remoto [34].

As bases de *wavelets* ortonormais são muito utilizadas em diversas aplicações, e mais recentemente em GPS (Sistema de Posicionamento Global - *Global Position System*). Algumas bases bi-ortogonais apesar de ainda não utilizadas, têm sido apontadas como possuindo características muito interessantes para esta aplicação [35].

As *wavelets* ortogonais e biortogonais são utilizadas para realizar uma análise de multiresolução e reduzir erros nos sinais de GPS, principalmente com relação ao erro do multicaminho. Multicaminhos são causados pela reflexão dos sinais em objetos nas proximidades do receptor provocando um deslocamento no sinal direto e, portanto, diversos erros que prejudicam a precisão e acurácia dos resultados obtidos nos sistemas de posicionamento por satélite. Trata-se de um dos erros mais difíceis de serem reduzidos, tendo sido alvo de muitas investigações. É nesse sentido, relativo à redução de erros dos sinais GPS, que as *wavelets* têm mostrado seu grande potencial.

Nesse contexto, o sinal GPS é decomposto utilizando um algoritmo de filtragem, conhecido como algoritmo de Mallat, iterativo e com sucessivas aproximações. Como conseqüência, o sinal é dividido em componentes de resolução cada vez mais baixa, separando as altas freqüências, onde se encontra o efeito de multicaminho. Após o processo de decomposição, modificando os coeficientes *wavelets* através de métodos de limiarização, o sinal pode ser reconstruído sem o efeito do multicaminho [36].

A partir da interpretação de diferentes tipos de imagens é possível fazer a previsão do tempo,

estudar fenômenos oceânicos, detectar e monitorar furacões, queimadas e desflorestamentos, gerar mapas geológicos, de solos e de uso da terra, mapear os recursos hídricos, as áreas agrícolas, as áreas urbanas e acompanhar sua transformação e expansão, entre outras aplicações.

Os satélites artificiais, equipados com sensores que captam imagens da superfície terrestre, localizam-se ficam em órbitas distantes da Terra. Quanto mais distante o satélite, maior será a área da superfície imageada pelo sensor. Em contrapartida, quanto mais próximo ele estiver da Terra, menor será a área imageada pelo sensor e maior será a riqueza de detalhes da imagem captada. No processo de interpretação de uma imagem, ou seja, de identificação dos objetos nela representados, utiliza-se as variações de cor, forma, tamanho, textura (impressão de rugosidade), localização e contexto.

2.5.3 Imagens Naturais

Diversas imagens podem ser analisadas, dentre estas, as imagens que são obtidas por câmeras (dispositivos de recepção de imagens), onde estas transformam a informação luminosa recebida em uma representação padrão. A imagem obtida pode-se referir a imagens estáticas ou vídeo.

No Processamento Digital de Imagens, seja para imagens estáticas ou em movimento, sempre é levado em consideração a satisfação do usuário final e, portanto, pesquisas contínuas estão em ascensão com o intuito de prover melhores representações das mesmas, através da máxima retirada na quantidade de informação redundante da imagem original, respeitando a percepção do SVH.

Neste sentido, hoje é possível prover recepção doméstica de diversos canais de televisão digitais, graças ao desenvolvimento de técnicas de transmissão bastante efetivas e à padronização de técnicas de compressão de áudio e vídeo. A importância da qualidade da imagem não deve ser menosprezada, já que o usuário certamente irá levá-la em consideração ao fazer sua escolha pelos novos serviços disponíveis.

Mostrando o interesse em preservar a qualidade de imagem para o usuário final, pode-se citar, como exemplo, que a avaliação de qualidade de vídeo digital tem sido objeto de estudo de diversos grupos de pesquisa, principalmente os pertencentes ao VQEG (*Video Quality Experts Group*) [18], que disputam atualmente o processo internacional de padronização da avaliação objetiva de TV Digital. Dentre os modelos para esta padronização, pode citar a segmentação de cenas naturais em regiões planas, de borda e de textura, e um conjunto de parâmetros objetivos é calculado para cada um destes contextos.

No que se refere à TV digital de alta definição, este é mais um ramo a se beneficiar imensamente com o estudo de Processamento Digital de Imagens. O Brasil se depara diante de um novo salto tecnológico na qualidade da televisão, sendo que suas transmissões digitais começaram no dia 02 de dezembro de 2007 em São Paulo. Muitas capitais já desfrutam de tal tecnologia dentre estas, São Paulo, Rio de Janeiro, Belo Horizonte e Goiânia. As demais capitais deverão receber o sinal para este

ano de 2009 e o mesmo deve se concretizar nos próximos anos nas demais cidades.

A TV digital surge como uma maneira de incorporar novas tecnologias a uma mídia já consagrada, aumentando a atratividade e o potencial competitivo da televisão ao oferecer uma melhoria considerável na qualidade da imagem e do som, pelo fato de eliminar ruídos de sinal, e oferecendo serviços diferenciados como transmissão de vários programas em um só canal, acesso à *internet*, interatividade e recepção móvel. Vale comentar que, para a TV digital, a avaliação da qualidade de imagem realizada de forma objetiva, a qual é adaptável para o tipo de aplicação e para as exigências do usuário, pode estar atrelada a um sistema de recepção de imagens, que ao detectar uma imagem insatisfatória, solicita o reenvio de informação. Maiores detalhes da TV digital no mercado brasileiro pode ser encontrado em [37].

O padrão brasileiro denominado de SBTVD (Sistema Brasileiro de Televisão Digital - *Brazilian Digital Television System*), proveniente de adaptação do padrão japonês ISDB (*Integrated Services Digital Broadcasting*), foi estabelecido para transmissão de sinais digitais de TV terrestre por emisoras de TV aberta no Brasil, decisão esta que o apontou como o mais flexível entre os padrões existentes, por responder melhor a necessidade de mobilidade e portabilidade. O padrão brasileiro, inicialmente denominado de ISBT-Tb, difere do padrão japonês por utilizar a codificação de vídeo no padrão H.264/MPEG-4 AVC (*Advanced Video Codec*). O Japão utiliza o padrão MPEG-2. O padrão de compressão de vídeo H.264 ou AVC, baseado no MPEG-4 Parte 10, faz uso de boa qualidade de vídeo com uma taxa de *bitrate* muito baixa em relação aos padrões já existentes, dentre estes o MPEG-2.

Até o momento, o padrão SBTVD parece ter dado mais importância para a transmissão terrestre, já que esta almeja incluir toda a população em um futuro próximo. Entretanto, visando igualmente a recepção móvel de TV em celulares, o padrão DVB-H (*Digital Video Broadcasting- Handheld*) se apresenta como um grande potencial.

Vale comentar que, as degradações em vídeo aparecem quando a quantidade de informação presente no vídeo é maior do que a capacidade de transmissão do canal [18]. As degradações no vídeo digital são dificilmente mensuráveis objetivamente, embora sejam quase sempre óbvias para observadores humanos. Esta dificuldade refere-se, em parte, ao fato dos algoritmos de compressão introduzir degradações numa escala que depende fortemente do conteúdo de detalhes e movimento das cenas, e a percepção dessas distorções também é altamente dependente deste contexto. Assim sendo, os métodos tradicionais de avaliação, baseados em sinais estáticos, são inadequados para quantificar a qualidade desses sistemas.

Apesar de imagens em movimento não ser o objetivo da pesquisa, tais comentários tornam necessários para mostrar o quanto é importante estudar as características do SVH, para que possa ser explorado o máximo possível na representação da imagem final. Com o foco voltado para imagens

estáticas e referente à imagens naturais, pode citar como exemplo a imagem de Lena, imagem referência utilizada na literatura científica.



Figura 2.7: Imagem Lena (256x256 pixels).

Como última observação para o SVH, destaca-se também que aspectos como oclusão de objetos, desordem visual e desorientação visual dificultam a interpretação das informações pelo usuário [38]. A oclusão de objetos se deve a existência de um grande volume de informações e alguns elementos sobrepõem-se a outros. A desordem visual é decorrente da dificuldade de reconhecimento e interpretação pelo usuário de muitos elementos (não necessariamente sobrepostos) presentes na visualização o que força uma sobrecarga no seu sistema cognitivo. Já o problema de desorientação visual, ocorre quando o usuário tem dificuldade na manutenção da atenção por ocasião de troca do ponto de vista e retorno à situações anteriores.

2.6 Medidas Objetivas Tradicionais de Avaliação de Imagens

Um importante descritor da qualidade de imagens digitais reconstruídas quando comparadas com uma imagem original, o PSNR (Razão Sinal Ruído de Pico - *Peak Signal Noise Ratio*), é medido em escala logarítmica e depende do MSE (Erro Médio Quadrático - *Mean Square Error*) entre as imagens ou quadros de vídeo originais. Esta medida é relativa ao valor $(2^n - 1)^2$, que representa o quadrado do maior valor possível de sinal na imagem, onde n é o número de bits de cada imagem. Nas simulações realizadas, o valor utilizado foi de 255 e, desta forma, o PSNR entre a imagem original (F_0) e a imagem reconstruída (F_R), ambas de dimensões $W \times H$, é expresso por:

$$PSNR(F_0, F_R) = 10 \cdot \log \left(\frac{255^2}{MSE(F_0, F_R)} \right), \text{ onde :} \quad (2.1)$$

$$MSE(F_0, F_R) = \frac{1}{W \cdot H} \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} (F_0(x, y) - F_R(x, y))^2 \quad (2.2)$$

Para garantir que as imagens reconstruídas, tanto por intermédio da utilização do método tradicional utilizando bancos de filtros *wavelets* como também pelo esquema *lifting*, apresentem a mesma imagem como resultado, esta medida torna-se fundamental na comparação.

2.7 Outras Medidas Objetivas de Avaliação de Imagens

Para o estudo em questão, apenas as medidas de PSNR das imagens reconstruídas já é o suficiente para corroborar com a perfeita implementação do *lifting*, dado que o mesmo não pode implicar em deterioração na imagem reconstruída. Entretanto, outras medidas objetivas mais recentes e extremamente interessantes são apresentadas a seguir para fins de documentação do tema.

2.7.1 SSIM

O SSIM (Método da Similaridade Estrutural - *Structural Similarity*) foi proposto por Zhou Wang e Alan C. Bovik em 2002 [39]-[41] e trata-se de uma medida de qualidade relativa entre duas imagens, assim como a PSNR já definida. O SSIM foi desenvolvido com o objetivo de refletir a percepção do SVH atingindo resultados mais fiéis do que as medidas feitas com PSNR, voltada para sinais genéricos. Inicialmente, apresenta-se o SSIM para uma amostra uni-dimensional, indico que pode ser facilmente estendido para um sinal bidimensional (imagens digitais).

Sejam duas amostras $x = \{x_i \mid i = 1, 2, \dots, N\}$ e $y = \{y_i \mid i = 1, 2, \dots, N\}$ as amostras original e em teste, respectivamente. O índice SSIM é definido por:

$$Q = \frac{4\sigma_{xy}\bar{x}\bar{y}}{(\sigma_x^2 + \sigma_y^2) [(\bar{x})^2 + (\bar{y})^2]} \quad (2.3)$$

A média do sinal x e y é expresso, respectivamente, por:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.4)$$

$$\bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (2.5)$$

No que diz respeito à variância do sinal x e y , as mesmas são expressas por:

$$\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (2.6)$$

$$\sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \bar{y})^2 \quad (2.7)$$

Em (2.8) tem-se a covariância entre os sinais x e y :

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y}) \quad (2.8)$$

O índice SSIM pode variar entre $[-1, 1]$. O melhor valor, 1, é alcançado se, e somente se, $y_i = x_i$ para todo $i = 1, 2, \dots, N$ e, em contrapartida, o menor valor, -1, ocorre quando $y_i = 2\bar{x} - x_i$ para todo $i = 1, 2, \dots, N$. Qualquer distorção pode ser modelada por esse índice como o produto de três fatores: perda de correlação, distorção na luminância, distorção no contraste. O índice SSIM definido em (2.3) pode ser decomposta em:

$$Q = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \cdot \frac{2\bar{x}\bar{y}}{(\bar{x})^2 + (\bar{y})^2} \cdot \frac{2\sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2} \quad (2.9)$$

A primeira componente é o coeficiente de correlação entre x e y , que mede a correlação linear entre eles, variando de $[-1, 1]$. O melhor valor é obtido quando $y_i = ax_i + b$ para todo $i = 1, 2, \dots, N$, onde a e b são constantes e $a > 0$. Mesmo quando x e y estão linearmente relacionados, ainda existem distorções entre eles, que são avaliadas pela segunda e terceira componentes. A segunda componente, com variação de $[0, 1]$, mede a proximidade entre a luminância média de x e y . Será igual a 1 se, e somente se, $\bar{x} = \bar{y}$. σ_x e σ_y podem ser vistos como uma medida de contraste entre x e y , de forma que a terceira componente mede a proximidade entre o contraste das duas imagens. A variação se dá entre $[0, 1]$, atingindo o melhor valor, 1, se, e somente se, $\sigma_x = \sigma_y$.

2.7.2 Filtro de Contraste

A propriedade do contraste é muito mais importante na caracterização da imagem do que o brilho (intensidade média de luz). O SVH apresenta uma sensibilidade maior a alto contraste do que a baixo contraste. O contraste é uma propriedade da imagem definida como a razão entre o ponto mais

brilhante (mais próximo do branco) e o mais escuro (mais próximo do preto) da imagem. O trabalho de Manos e Sakrison [42] apresenta uma nova abordagem para o contraste. Sabe-se que o SVH é mais sensível a baixas frequências espaciais do que a altas. Por isso, define-se uma CSF (Função de Sensibilidade a Contraste - *Contrast Sensitivity Function*), em termos da frequência espacial (f), como mostrado na relação:

$$A(f) = 2,6 \cdot (0,0192 + 0,114 \cdot f) \cdot e^{-(0,114 \cdot f)^{1,1}} \quad (2.10)$$

Frequência espacial é o número de ciclos por unidade de espaço, que em percepção visual, denominado de ciclo por grau de ângulo visual. A curva CSF da Fig. 2.8 apresenta um máximo na frequência de 8 ciclos/grau e possui valores desprezíveis para frequências acima de 60 ciclos/grau. Essa função é de grande utilidade na análise da qualidade de um sinal de imagem processado. Através de filtros, é possível privilegiar as frequências espaciais de maior relevância para o SVH.

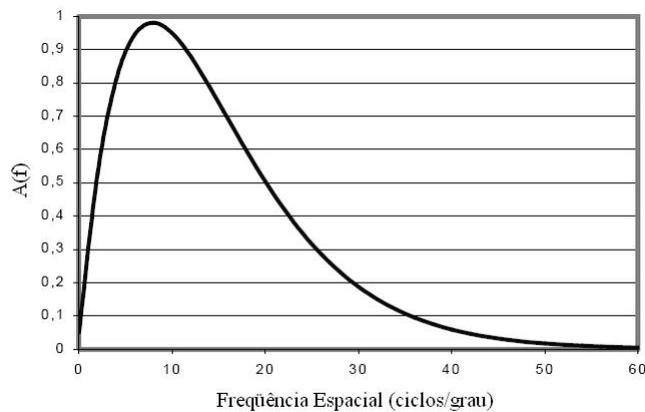


Figura 2.8: Função de sensibilidade à contraste [42].

2.8 Comentários

O capítulo abordou as principais características do SVH (Sistema Visual Humano - *Human Visual System*) através de um apanhado bibliográfico, abordando as principais características que o compõe, assim como a aplicabilidade das *wavelets* em diversas áreas do processamento de sinais. Como o campo de estudo de imagens é bastante vasto para análise de imagens médicas, de satélites e imagens naturais, as principais características e o estado da arte fez-se necessário, sendo apresentado a importância da TW nestas aplicações. Medidas objetivas tradicionais e novas medidas objetivas finalizam a apresentação deste capítulo.

Capítulo 3

TRANSFORMADA *WAVELET* CONTÍNUA (TWC)

A teoria sobre *wavelets* tem despertado atenção de várias áreas da ciência por sua origem interdisciplinar e a sua maneira simples e eficaz de implementação. No sentido de eliminar problemas inerentes a outras transformadas, contribuições de pesquisadores das mais diversas áreas, tais como a matemática, estatística, física, engenharia e computação gráfica permitem para que a mesma continue em ascensão.

Sob o ponto de vista de processamento de sinais, a transformada *wavelet* (TW) surgiu para sanar alguns problemas inerentes às outras transformadas como a análise de Fourier e a "*Short Time Fourier Transform*", objetivando um melhor desempenho nas diversas aplicações.

Tal interesse pela transformada *wavelet*, se deve por apresentar algoritmos de implementação rápida, boa capacidade de concentrar a energia do sinal em poucos coeficientes, além de não introduzir os efeitos de blocagem. Além do mais, importantes padrões de compressão de imagem e vídeo tais como o JPEG 2000, MPEG-4 e MPEG-7 fazem uso desta transformada [3].

O objetivo deste capítulo será realizar uma sucinta comparação entre as principais ferramentas no que diz respeito à análise de sinais estacionários e não-estacionários, assim como realizar uma abordagem sobre os princípios básicos da Transformada *Wavelet* Contínua 1D, sua interpretação através do conceito de bancos de filtros e a expansão em série da *wavelet* 1D e 2D. O estudo a seguir se restringe a funções quadraticamente integráveis 1D e 2D, uma vez que estas englobam os sinais e imagens que são de interesse.

3.1 Da Série de Fourier à TW

Muitas semelhanças são presenciadas entre a análise de Fourier e análise *wavelet*, tendo em ambos os casos a análise de sinais por expansões em termos de funções-base.

No que se refere à base de Fourier, esta é formada por ondas puras com as frequências variando sobre todo o espectro onde seus respectivos coeficientes, por sua vez, medem puramente o conteúdo frequencial do sistema sem identificar quando tal frequência ocorre. Em outras palavras, seu resultado refere-se ao conteúdo de frequência de um processo ou sinal, mas não fornece informações sobre a localização no domínio do tempo, apresentando assim resolução infinita em frequência e nula no tempo. Na transformada *wavelet*, as bases são localizadas tanto no domínio da frequência quanto no domínio do tempo, promovendo um balanceamento das resoluções no tempo e em frequência, de forma que ambas sejam finitas e não-nulas.

Há muitas décadas, cientistas têm procurado funções mais apropriadas do que as senóides e cossenóides (que compõem as bases da análise de Fourier) a fim de aproximar sinais com descontinuidades. Dennis Gabor em 1946 foi quem primeiro adaptou a TF (*Transform Fourier* - Transformada de Fourier) para analisar somente uma pequena seção do sinal no tempo, técnica esta conhecida como *Short Time Fourier Transform* (STFT), podendo ser interpretada como Transformada de Fourier de Curta Duração ou Transformada de Fourier Janelada. Este método propõe a divisão do sinal não-estacionário em intervalos pequenos o suficiente, de modo que se possa assumir estacionaridade em cada um deles. Assim sendo, a TF pode ser aplicada sem restrições. A totalidade dos espectros de cada intervalo mostra como o conteúdo de frequência do sinal está variando com o tempo.

Matematicamente, faz-se o uso de uma janela de observação, $h(x)$, centrada em x , para enfatizar o sinal apenas nas proximidades do instante de tempo que se deseja analisar:

$$s(\tau) = s(\tau) h(x - \tau) \quad (3.1)$$

Como o comprimento da janela $h(x)$ permanece fixo, esta transformada não é uma técnica adequada para analisar sinais que apresentem componentes em altas e baixas frequências ao mesmo tempo, pois esta janela continua a mesma para todas as frequências. Calcular a STFT de um sinal, corresponde ao cálculo de várias TF, uma para cada porção do sinal.

Contudo, a maioria dos sinais necessita de uma abordagem mais flexível, o que em outras palavras, refere-se a uma realização da análise do sinal através de uma janela de comprimento variável, obtendo assim uma localização mais precisa de um determinado evento tanto no tempo quanto em frequência.

Neste contexto, a TW foi desenvolvida como uma alternativa à STFT cuja resolução no tempo e frequência permanecem constantes em todo plano, conforme evidenciado na Fig. 3.1(a). Visando oferecer uma solução para esta peculiaridade, a *wavelet* $\psi(x)$ possui função semelhante ao da janela

apresentada na STFT, porém apresenta regiões de dimensão variável, permitindo desta forma, o uso de intervalos longos de tempo onde existe conteúdo de baixa frequência e intervalos menores de tempo para informação de alta frequência, como ilustrado na Fig. 3.1(b).

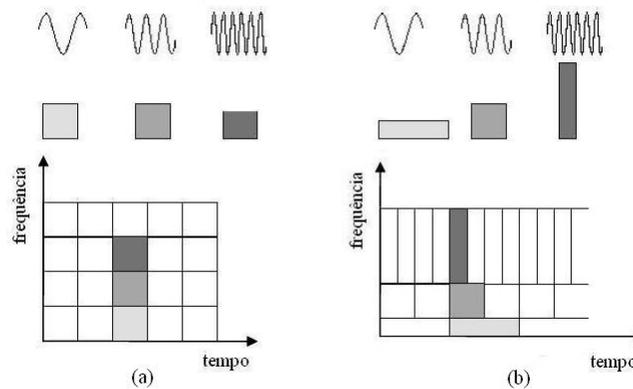


Figura 3.1: Tipos de transformadas (a) Transformada de Fourier Janelada (b) Transformada *Wavelet* [43].

Na análise *wavelet*, ao invés de interpretar os parâmetros nos domínios tempo e frequência, costuma-se utilizar os domínios escala e deslocamento, respectivamente, conforme abordado nas seções seguintes.

3.2 Transformada *Wavelet* Contínua 1D

A TWC (Transformada *Wavelet* Contínua - *Continuous Wavelet Transform*) foi introduzida por Grossman e Morlet [44]. Seja a *wavelet* $\psi(x)$ uma função oscilatória e de curta duração, a TWC é uma transformada real sendo definida pela seguinte expressão:

$$TWC(a, b) = \langle f, \psi_{a,b} \rangle \triangleq \int_{-\infty}^{\infty} f(x) \cdot \psi_{a,b}(x) dx \quad (3.2)$$

As funções-base $\psi_{a,b}(x)$ são comumente chamadas de *wavelets*-filhotes, sendo versões deslocadas e dilatadas ou comprimidas da *wavelet*-mãe $\psi(x)$ e capazes de gerar todo espaço $L^2(R)$. Para $a > 0$ e b números reais tem-se:

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x-b}{a}\right) \quad (3.3)$$

Em (3.2) pode ser observado que a TW é expressa em termos dos parâmetros a e b , onde o parâmetro b é responsável pelo deslocamento de $\psi(x)$ ao longo do eixo x , e o parâmetro a denominado escala, é responsável pela compressão ou dilatação, permitindo flexibilidade à $\psi(x)$ e possibilitando

umentar ou diminuir a largura conforme desejado. Como a *wavelet*-mãe possui média nula, todos os escalonamentos e translações de (3.3) também possuem média nula.

A constante $a^{-1/2}$ presente em (3.3) e consequentemente em (3.2), é o termo de normalização da energia da função com relação ao parâmetro a e assegura que as normas das funções-base sejam iguais de modo que:

$$\left\| f\left(\frac{x-b}{a}\right) \right\| = \sqrt{\int_{-\infty}^{\infty} \left| f\left(\frac{x-b}{a}\right) \right|^2 dx} = \sqrt{a} \|f(x)\| \quad (3.4)$$

Para reconstruir $f(x)$ a partir de seus coeficientes *wavelets*, é necessário que a condição de admissibilidade seja satisfeita para obtenção da TWC inversa, sendo esta definida por [45] [46]:

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\psi(s)|^2}{|s|} ds < \infty \quad (3.5)$$

Para definir se uma função pode ser uma *wavelet*, esta função tem que ser oscilatória, ou seja, seu valor médio no domínio temporal é nulo. A condição de admissibilidade restringe a classe de funções que podem ser utilizadas como *wavelet*-mãe, onde $\psi(s)$ deve se anular na origem e é o espectro da *wavelet*-básica real $\psi(x)$. Como s é denominador da integral, a exigência é que:

$$\psi(0) = 0 = \int_{-\infty}^{\infty} \psi(x) dx = 0 \quad (3.6)$$

Pode se dizer que, o espectro de amplitude da *wavelet*-mãe é similar à função de transferência de um filtro passa-faixas, já que $\psi(\infty) = 0$.

De forma resumida, a *wavelet* é uma função oscilatória com média nula e que tem decaimento rápido para zero.

A condição para existência da TWC inversa exige que a constante C_ψ seja finita, podendo assim a TWC inversa ser expressa por:

$$f(x) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty W(a,b) \psi_{a,b}(x) db \frac{da}{a^2} \quad (3.7)$$

3.3 Interpretação da TWC por Banco de Filtros

Como visto, na TWC todas as respostas ao impulso no banco de filtros são versões escalonadas (expandidas ou comprimidas) de uma mesma *wavelet*-básica $\psi(t)$. A *wavelet*-mãe escalonada de a e normalizada por $a^{-1/2}$, pode ser escrita como:

$$\psi_a(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x}{a}\right) \quad (3.8)$$

Definindo-se seu complexo conjugado refletido, tem-se:

$$\tilde{\psi}_a(x) = \psi_a^*(-x) = \frac{1}{\sqrt{a}}\psi^*\left(\frac{-x}{a}\right) \quad (3.9)$$

De modo análogo, a *wavelet*-mãe deslocada por b , pode ser expressa por:

$$\psi_b(x) = \psi(t - b) \quad (3.10)$$

Em (3.3) pode-se perceber o deslocamento com escalonamento. É de importância frisar que, se $\psi(x)$ for real e par, como freqüentemente ocorre, a reflexão e a conjugação não tem efeito. A TWC e sua inversa podem então serem reescritas respectivamente como:

$$W(a, b) = \int_{-\infty}^{\infty} f(x) \psi_a(x - b) dx = f * \tilde{\psi}_a \quad (3.11)$$

$$f(x) = \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} [f * \tilde{\psi}_a](b) \psi_a(b - x) db \frac{da}{a^2} = \frac{1}{C_\psi} \int_0^{\infty} [f * \tilde{\psi}_a * \psi_a](x) \frac{da}{a^2} \quad (3.12)$$

Nota-se que para um valor fixo de a , $W(a, b)$ é a convolução do sinal $f(x)$ com o conjugado refletido da *wavelet*-mãe na escala a . Para a condição de a fixo, cada $W(a, b)$ é um conjunto de coeficientes de mesma freqüência, logo, a TWC de um sinal corresponde à um banco de filtros lineares atuando sobre $f(x)$, onde cada valor de a define diferentes filtros passa-faixa. As saídas de todos juntos compõe a TWC, conforme visto na Fig. 3.2.

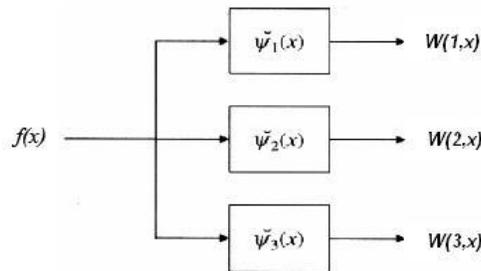


Figura 3.2: Interpretação da TWC por analogia de um banco de filtros [3].

Vale destacar que, para a interpretação da TWC visualizada na Fig. 3.2, existe uma única *wavelet*-mãe, onde cada bloco representado por $\tilde{\psi}_a(x)$ consiste nesta *wavelet*-mãe escalonada com uma freqüência distinta e pelo fato de estarem deslocadas pelo parâmetro b , cada uma delas torna-se uma família de *wavelets*. Outro aspecto a frisar é que, cada uma apresenta um conjunto de funções-base de mesma freqüência.

Um banco de filtros é um arranjo de filtros passa-faixa que decompõe o sinal em diversas componentes, cada uma carregando apenas uma sub-banda de frequência do sinal original. É desejável que o projeto do banco de filtros seja tal que permita a recombinação das sub-bandas para se recuperar o sinal original obedecendo (3.11). O primeiro processo, consistindo na separação das sub-bandas, é chamado de análise e o segundo de síntese. A saída da análise é chamada de sinal de sub-bandas, com tantas sub-bandas quantos forem os filtros.

O banco de filtros serve para isolar as diferentes componentes de frequência de um sinal, priorizando determinadas frequências de acordo com o objetivo. Por exemplo, as frequências mais importantes podem ser codificadas com uma resolução maior e por outro lado, as frequências menos importantes não precisam de reconstrução exata e uma codificação mais esparsa poderia ser adotada, mesmo que se percam detalhes na codificação.

Conforme dito, cada valor de a define diferentes filtros passa-faixa. Isto significa que, para valores altos de a , a *wavelet* definida em (3.8) irá sofrer dilatação no tempo e, por conseguinte, contração em frequência, deslocando o centro da banda passante para valores mais baixos em frequência e realizando a análise através de filtro estreito, conforme ilustrado pela Fig. 3.3. Para este caso, tem-se um maior número de componentes do sinal original na formação de cada coeficiente transformado.

De modo análogo, filtros mais longos podem ser utilizados quando se estiver realizando a análise das altas frequências do sinal, que ao contrário, resulta na utilização de um número menor de componentes do sinal original na formação de cada coeficiente transformado durante o processo de convolução.

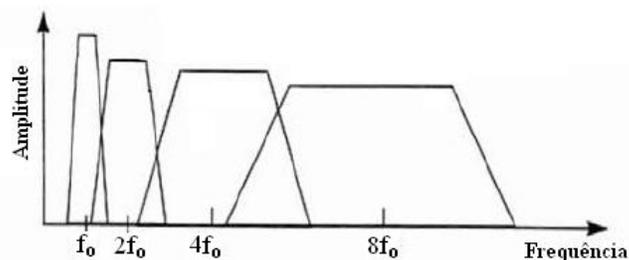


Figura 3.3: Análise espectral com banco de filtros [43].

Esta relação de tempo e frequência é regida pelo princípio de incerteza de Heisenberg (3.13), e como consequência, um acréscimo de $\Delta(f)$ resulta em uma redução do parâmetro $\Delta(t)$ e vice-versa [47]. Pode-se perceber que, a medida que se aumenta a precisão na determinação de uma resolução, perde-se precisão na determinação da outra. Caso o interesse seja em analisar altas frequências, menores valores de a podem ser escolhidos, obtendo deste modo uma boa localização no tempo.

$$\Delta f \times \Delta t \geq \frac{1}{4\pi} \quad (3.13)$$

Deste modo, percebe-se que escalas maiores permitem analisar uma maior parte do sinal enquanto escalas menores permitem analisar trechos mais curtos de forma detalhada. Fazendo uma analogia com uma floresta, escalas maiores podem ter utilidade para situar uma determinada visão geral da mesma e uma escala menor para localizar uma árvore em questão.

3.4 TWC Bidimensional

A TWC-2D de $f(x, y)$ utilizando uma *wavelet*-mãe bidimensional, $\psi(x, y)$, apresentando função oscilatória e de curta duração, pode ser representada por [17]:

$$W(a; b_x, b_y) = \langle f(x, y), \psi_{a; b_x, b_y}(x, y) \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \psi_{a; b_x, b_y}(x, y) dx dy \quad (3.14)$$

Em (3.14), os parâmetros b_x e b_y especificam a translação em duas dimensões. As *wavelets*-filhotes bidimensionais, $\psi_{a; b_x, b_y}(x, y)$, geradas a partir de $\psi(x, y)$, podem ser definida por:

$$\psi_{a; b_x, b_y}(x, y) = \frac{1}{a} \psi\left(\frac{x - b_x}{a}, \frac{y - b_y}{a}\right) \quad (3.15)$$

A TWC-2D inversa pode ser obtida, desde que a condição de admissibilidade seja satisfeita [48]. Deste modo, considerando $\psi(s_x, s_y)$ como a TF-2D de $\psi(x, y)$, essa condição de admissibilidade pode ser caracterizada por:

$$C_\psi = \int_0^{\infty} \int_0^{2\pi} \frac{|\psi(r \cos \theta, r \sin \theta)|^2}{r} dr d\theta < \infty \quad (3.16)$$

Se a *wavelet*-mãe for esfericamente simétrica, sua TF também será esfericamente simétrica [3]. De modo simplificado, (3.16) pode ser reescrita como:

$$C_\psi = \int_0^{\infty} \frac{|\psi(a^{-1}s_x, a^{-1}s_y)|^2}{a} da < \infty, \forall (s_x, s_y) \in R^2 \quad (3.17)$$

Caso a condição de admissibilidade for finita, implicará na existência da TWC-2D inversa definida como:

$$f(x, y) = \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{+\infty} W(a; b_x, b_y) \psi_{a; b_x, b_y}(x, y) db_x db_y \frac{da}{a^3} \quad (3.18)$$

A TWC, $W(a, b)$, de uma função $f(x)$ é uma função de duas variáveis e para funções de mais de uma variável, a TWC também aumenta em um a dimensionalidade, mas todos equacionamentos apresentados são válidos.

A TWC é considerada sobrecompleta, devido ao fato de representar um aumento considerável no volume de informação e no volume necessário para armazenamento de dados. Por conseqüência, o valor potencial da TWC não reside em representação compacta, mas sim na decomposição e análise [3].

3.5 Série de Coeficientes *Wavelets* (SCW)

A discretização da TW ocorre no domínio dos parâmetros referente às variáveis de escala (a) e translação (b). Para o parâmetro a , toma-se valores inteiros (positivos e negativos), potências de um parâmetro fixo a_0 , em que sofre discretização exponencial:

$$a = a_0^j \quad (3.19)$$

A discretização do parâmetro b depende de j tal que *wavelets* estreitas (de alta frequência) sejam deslocadas por passos menores e *wavelets* largas (de baixa frequência) sejam deslocadas por passos maiores, em que este parâmetro b sofre uma discretização proporcional à a , de modo a evitar a geração de redundância, sendo representado por:

$$b = k \cdot b_0 \cdot a_0^j \quad (3.20)$$

É importante salientar que, as constantes a_0 e b_0 são os comprimentos dos passos discretos de escala e deslocamento, respectivamente.

Deste modo, a *wavelet* discreta pode ser representada da forma:

$$\psi_{j,k}(x) = a_0^{-j/2} \psi(a_0^{-j} \cdot x - kb_0) \quad (3.21)$$

Dentro desse contexto, os coeficientes *wavelets* $W(j, n)$ podem ser caracterizados como:

$$W(j, n) = a_0^{-j/2} \int_{-\infty}^{\infty} f(x) \cdot \psi(a_0^{-j} \cdot x - kb_0) dx \quad (3.22)$$

O fator a_0 não pode ser arbitrário, pois diferentes valores de a_0 resultam em *wavelets* diferentes e bases ortonormais de *wavelets* só são conhecidas para valores racionais de a_0 [45]. Percebe-se que, escolher valores de a maiores resulta em um maior número de componentes do sinal original para formar cada coeficiente transformado, implicando em passos de deslocamentos *wavelets* maiores, de

modo a evitar redundância. Analogamente, tomando valores de a menores, passos de deslocamentos menores são realizados. A proporção em que os passos são realizados, de acordo com seus respectivos valores de a , torna-se importante para evitar o excesso de redundância durante a convolução do sinal $f(x)$.

3.5.1 Wavelets Diádicas

O tamanho da escala discreta a_0 geralmente é escolhido como sendo $1/2$ em (3.21), de forma a facilitar a implementação, pois em outras palavras, dilatar por um fator de 2 equivale a tomar uma amostra sim e outra não [17]. Deste modo, as *wavelets* diádicas apresentam escalonamentos binários (2^j) e deslocamentos diádicos ($k/2^j$), onde (3.21) pode ser representado por:

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j \cdot x - kb_0) \quad j, n \in Z \quad (3.23)$$

O plano deslocamento-escala pode ser amostrado por uma grade diádica, onde cada $W(j, n)$ associado a $\psi_{j,n}(k)$ é ilustrado com um ponto na Fig. 3.4.

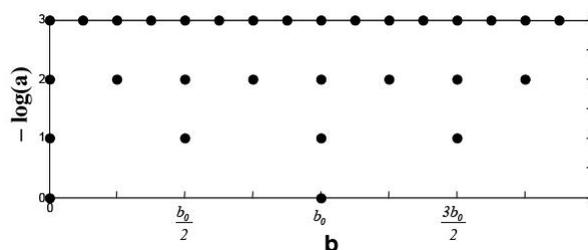


Figura 3.4: Discretização deslocamento-escala para *wavelets* diádicas [3].

Wavelets diádicas constituem bases ortonormais e permitem a caracterização de um sinal $f(x)$ sem redundância, pois, utilizando a análise multiresolução, pode ser mostrado que, qualquer função $f(x)$ pode ser aproximada numa precisão arbitrária por combinações lineares de *wavelets* ortonormais [45].

Vale comentar que, caso a *wavelet*-básica apresente norma unitária e suporte compacto, todos os elementos do sistema formado apresentarão a mesma propriedade, ou seja, as *wavelets*-filhotes também apresentarão as mesmas características assegurando a conservação de energia de $f(x)$ em (3.22).

A função $f(x)$ pertencente ao espaço $L^2(R)$ pode ser reconstruída a partir dos coeficientes *wavelets*, $W_{j,n}$, conforme visto em (3.24), em que a soma convergirá em média quadrática para $f(x)$ [46]. Vale ressaltar que, $W_{j,n}$ são os coeficientes da Transformada *Wavelet* Discreta (TWD) e se assemelham aos coeficientes da análise de Fourier.

$$f(x) = \sum_{j=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} W_{j,n} \cdot \psi_{j,n}(x) \quad (3.24)$$

Se $f(x)$ possuir suporte compacto e se $\psi(x)$ possuir a característica de decair rapidamente a zero conforme se distanciar da origem, tem-se que, muitos dos coeficientes com altos valores de $|n|$ apresentarão pequenas amplitudes, sendo omitidos dependendo da aplicação. Para os altos valores de $|j|$, por sua vez, a área das funções-base tendem a zero e, portanto, os coeficientes relacionados a ela podem ser negligenciados [3]. Isto reflete diretamente na redução do esforço computacional.

No caso da TWD, a precaução no que se refere à escolha da *wavelet*-mãe e quanto à densidade da grade torna-se necessária de modo que a reconstrução não se torne instável [49]. No que se refere à densidade da grade, caso esta seja densa, as *wavelets* são redundantes e a reconstrução é obtida com maior fidelidade, porém com baixa eficiência. Para o caso de uma densidade esparsa a representação *wavelet* se torna menos redundante, porém, a equação de reconstrução pode não ser mais válida devido à perda de informação durante o processo de filtragem. Logo, um compromisso entre tipos de densidade da grade torna-se necessário, onde esta deve ser densa apenas o suficiente para que a reconstrução seja obtida com o mínimo de perda e conseqüentemente, com o máximo de eficiência.

Se a família *wavelet* $\{\psi_{j,k}\}$ é definida em (3.23) e forma uma base ortonormal em $L^2(R)$, a *wavelet* é considerada ortonormal, onde:

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l} \cdot \delta_{k,m} \quad j, k, l, m \in Z \quad (3.25)$$

Vale destacar que, os parâmetros l e m são inteiros e a única imposição em (3.25) para se obter como resposta o valor 1, diz respeito à condição de $j = k$ e $l = m$. Os coeficientes podem ser calculados por:

$$W_{j,n} = \langle f, \psi_{j,n} \rangle \quad (3.26)$$

Deste modo, a série em 3.24 converge em $L^2(R)$, isto é:

$$\lim_{M_1, N_1, M_2, N_2 \rightarrow \infty} \left\| f(x) - \sum_{j=-M_2}^{N_2} \sum_{n=-M_1}^{N_1} W_{j,n} \cdot \psi_{j,n}(x) \right\| = 0 \quad (3.27)$$

3.5.2 SCW Bidimensional

O embasamento teórico aplicado ao caso unidimensional pode-se estender também à SW-2D. Deste modo, fixando os valores a_0 e b_0 e assumindo $a = a_0^j$; $b_x = n_x \cdot a_0^j$ e $b_y = n_y \cdot a_0^j$. Sendo assim, a SCW - 2D pode ser definida por:

$$W_{j;n_x,n_y} = a^{-j} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot \psi(a_0^{-j}x - n_x, a_0^{-j}y - n_y) dx dy \quad (3.28)$$

e, assumindo $a_0 = 1/2$ e $b_0 = 1$, para o caso das *wavelets* diádicas bidimensionais, tem-se que:

$$\psi_{j;n_x,n_y}(x, y) = 2^j \psi(2^j x - n_x, 2^j y - n_y) \quad (3.29)$$

3.6 Comentários

Um breve apanhado histórico foi realizado, de modo a relatar as ferramentas que antecederam o surgimento da TW, assim como suas principais características. Deste modo, um dos assuntos abordados e que merece destaque, refere-se à interpretação da TW por intermédio da implementação por banco de filtros, já que as aplicações da transformada *wavelet* residem nesse conceito. Esta abrangência teórica torna-se de grande importância, pois com esta breve introdução à TWD (*Discret Wavelet Transform* - Transformada *Wavelet* Discreta), um melhor embasamento para o capítulo seguinte pode ser alcançado.

Capítulo 4

INTRODUÇÃO AOS PRINCÍPIOS DA TRANSFORMADA *WAVELET* DISCRETA

A TWC, conforme visto, mapeia um sinal unidimensional (no tempo) em uma representação bidimensional (tempo, escala) altamente redundante, em que realiza translações e escalonamentos contínuos de uma função sobre um sinal, calculando uma correlação entre eles. Esta transformada não é muito adequada, pois seriam requeridas infinitas translações e escalonamentos, requerendo muito tempo e recursos computacionais, ainda assim, gerando muita redundância.

Deste modo, as transformadas contínuas podem ser aplicadas em análise de sinais, cabendo ao modo discreto a aplicabilidade no que diz respeito à implementação do ponto de vista computacional. Conforme visto, essa discretização da TW ocorre no domínio dos parâmetros referente às variáveis de escala e translação.

Com o propósito de analisar a implementação da TWD no decorrer dos capítulos, um embasamento teórico torna-se necessário no que se refere à codificação por sub-banda e ao algoritmo rápido de Mallat [24]. A codificação por sub-banda, técnica que levou ao desenvolvimento da implementação da TWD, possibilita a codificação compacta de sinais digitais, através da decomposição em componentes limitados em faixas de frequências, representando-os sem redundância e de modo que seja alcançado a reconstrução do sinal sem erros [17][50].

4.1 Codificação por sub-banda

A codificação por sub-bandas refere-se a uma técnica tempo-frequência inicialmente desenvolvida para o processamento de voz [51], que posteriormente foi estendida também para o caso de imagens. Essa técnica é baseada na decomposição do sinal original em sub-bandas que correspondem às componentes do sinal em diferentes faixas de frequências. Com o foco voltado apenas para imagens, esta

transformação passa a representar determinada imagem de acordo com os coeficientes distribuídos dentro destas diversas sub-bandas.

Após a decomposição, cada sub-banda do sinal tem seus coeficientes quantizados e, posteriormente, codicados. Já que estas quantizações são projetadas para corresponderem a bandas de diferentes percepções de ruído, uma quantização diferenciada pode ser utilizada com um maior número de níveis de quantização para bandas nas frequências de maior sensibilidade de visão, enquanto as bandas de menor sensibilidade podem ser quantizadas de forma mais grosseira. Esta adequação dos níveis de quantização às diversas sub-bandas é conhecida por modelamento do ruído (*noise shaping*).

Um modo de codificar um sinal discreto por sub-banda é através de filtragem digital. Filtrar um sinal nada mais é que aplicar ganhos específicos em frequências ou bandas de frequências escolhidas. Nesse contexto, a frequência de corte define o limite entre as frequências que passarão pelo filtro e aquelas que serão atenuadas. A banda de frequências que passa pelo filtro sem sofrer alterações é designada banda de passagem e a banda de frequências que é influenciada pelo filtro tem a designação de banda de atenuação.

Estes filtros utilizados no processo são de fundamental importância. Cada sub-banda resultante da decomposição tem de corresponder a uma faixa espectral do sinal original e o objetivo é que este seja recuperado sem perdas (filtros de reconstrução perfeita). Quando se deseja utilizar a decomposição de imagens em sub-bandas torna-se necessário a utilização de filtros bidimensionais. Também há casos de aplicações de filtros 3D para segmentações de vídeo.

Várias alternativas de utilização de filtros de reconstrução perfeita são empregadas na literatura, porém o mais usual refere-se a um par de filtros unidimensionais com respostas espectrais complementares: um passa-baixa e outro passa-alta. Neste caso, a sub-amostragem obtida (decimação) é realizada por um fator de 2 implicando em tomar uma amostra sim e outra não do sinal. Em [52] é realizada uma decomposição de imagens estáticas com sub-amostragem apenas das componentes de baixa-frequências, de modo que o número total de coeficientes para representar as imagens aumenta por um fator aproximadamente igual a $4/3$, o que torna indesejável.

No processo de filtragem, o sinal original $f(x)$ é decomposto em uma versão de baixa resolução ("borrada") do sinal original e em uma versão que contém basicamente os detalhes do sinal original. Quando a decomposição requer um número maior de bandas, o filtro é utilizado em cascata, em que a saída de uma etapa do filtro ingressa em outra, conforme ilustra a Fig. 4.1.

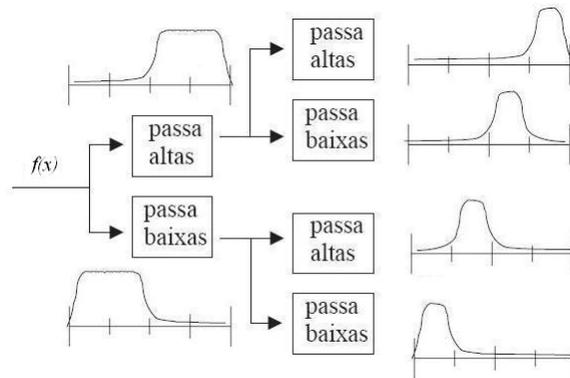


Figura 4.1: Decomposição em mais de duas sub-bandas.

A decomposição por filtros assimétricos torna-se uma alternativa à abordagem de filtros unidimensionais simétricos [53]. Através desta, as larguras de faixas dos filtros passa-baixas e passa-altas são diferentes correspondendo, respectivamente, às seguintes frações da largura de faixa do sinal original: $1/K_l$ e $1/K_h$. As decimações, para este caso, também têm de ser diferenciadas, apresentando um fator K_l para as baixas frequências e K_h para as altas. Os fatores K_l e K_h devem ser racionais para que possa haver uma reconstituição perfeita do sinal original.

A decomposição em sub-bandas pode ser feita através de filtros não-separáveis. Alguns trabalhos procuram obter filtros não-separáveis que operem de maneira fiel ao processamento da visão do que filtros separáveis. Dentre estes, destacam-se os trabalhos de Mahesh e Pearlman [54] e Simoncelli e Adelson [55], baseados em amostragem hexagonal de imagens, e o trabalho de Bamberger e Smith [56], baseado em amostragem convencional.

Regida pela teoria de análise multiresolução, outra decomposição em canais, introduzida por Mallat [57], fornece um caminho alternativo para se obter filtros de reconstrução perfeita, onde sinais são divididos em sub-bandas com larguras iguais em escala logarítmica, divisão esta ilustrada pela Fig. 4.2 e detalhada na seção seguinte. Vale destacar que, na Fig. 4.2, os blocos referentes aos filtros passa-baixa e passa-alta incluem os processos de decimação.

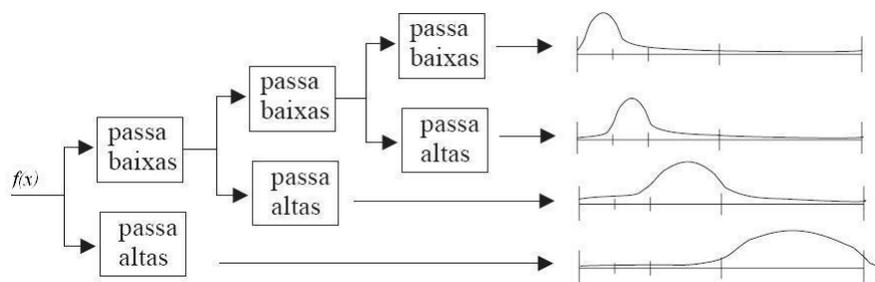


Figura 4.2: Decomposição da imagem em diferentes escalas.

4.2 Introdução ao Algoritmo Rápido para TWD

Uma vez determinada a *wavelet*-básica do sistema, Mallat [24] definiu um algoritmo mais eficiente, e, que é efetivamente utilizado para a implementação da TWD, tratando-se de um caminho alternativo para se obter filtros de reconstrução perfeita. Este é conhecido simplesmente por algoritmo de Mallat ou também chamado de FWT (*Fast Wavelet Transform*) ou *Mallat's Herringbone Algorithm*. A Fig. 4.3, esquematiza a realização da decomposição do sinal.

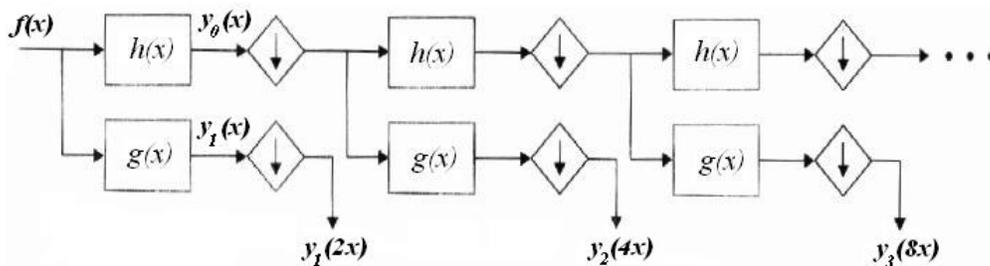


Figura 4.3: Algoritmo de Mallat (TWD direta) [24].

Neste contexto, a codificação por sub-banda começa com a partição do eixo de frequências em intervalos disjuntos de forma a permitir a implementação de filtros passa-faixas nos quais o sinal será submetido. Deste modo, considerando que o sinal a ser analisado seja $f(x)$, seu espectro é submetido a dois filtros sendo estes passa-baixas meia banda $h(x)$ e passa-faixas meia banda $g(x)$.

O sinal $y_0(x)$ refere-se à saída do filtro passa-baixas $h(x)$ e sofre a seguir, sub-amostragem ou decimação, sendo geralmente representado com a metade das amostras sem perda de informação [24]. Já $y_1(x)$ refere-se à saída do filtro $g(x)$ e contém exatamente a informação de alta frequência que foi eliminada de $f(x)$ durante a geração de $y_0(x)$, podendo dizer que, $y_0(x)$ e $y_1(x)$ contém juntos toda a informação presente no sinal original $f(x)$ [3] [17]. Em outras palavras, tem-se que:

$$f(x) = y_0(x) + y_1(x) = f(x) * h(x) + f(x) * g(x) \quad (4.1)$$

O processo é iterativo sobre o sinal resultante do filtro passa-baixas obtido do passo anterior, sendo este novamente submetido aos dois próximos filtros até que seja obtido um único coeficiente. Vale frisar que, as saídas dos filtros $g(x)$ são retidas a cada processo de iteração, correspondendo às características de detalhe do sinal. Os coeficientes transformados correspondem ao coeficiente banda-baixa e ao conjunto de coeficientes meia-banda alta codificados, totalizando N coeficientes.

Cada conjunto de coeficientes transformados é obtido através da convolução de $f(x)$ repetidamente com $h(x)$ e somente uma vez com $g(x)$. Daí as funções-base da TWD, $\psi_{j,n}(x)$, serem a reflexão de $g(x)$ e de funções derivadas da convolução de $g(x)$ repetidamente com $h(x)$ [3].

O processo de reconstrução do sinal original $f(x)$, realiza o processo inverso do que foi descrito. Por intermédio de super-amostragem dos dois sinais codificados em sub-banda, seguido de uma interpolação com $2h(x)$ e $2g(x)$ e efetuando a soma destes, tem-se como resultado o sinal analisado $f(x)$, conforme visto pela Fig. 4.4.

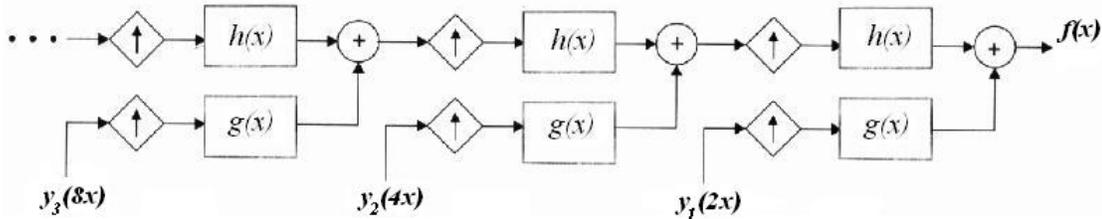


Figura 4.4: Algoritmo de Mallat (TWD inversa) [24].

4.3 Comentários

O embasamento teórico deste capítulo envolvendo a codificação por sub-banda, assim como a visão sobre o algoritmo de Mallat, possibilita compreender de forma mais simples o estudo da implementação deste algoritmo através da análise multiresolução detalhada no decorrer dos capítulos, especialmente no Capítulo 5, onde se introduz o conceito de multiresolução.

Capítulo 5

ANÁLISE MULTIRESOLUÇÃO E PROJETO DA TWD

A análise multiresolução é uma técnica que permite analisar o sinal através da decomposição de seu conteúdo em um conjunto de detalhes em diferentes resoluções, levando naturalmente a um esquema rápido e hierárquico para a obtenção dos coeficientes *wavelets*. A resolução de um sinal $f(x) \in L^2(R)$, definido no intervalo $0 < x < x_0$, é a quantidade de pontos para os quais ele é definido [24] [48]. Ao alterar a resolução do sinal $f(x)$ original, o sinal resultante dessa mudança é chamado de uma *aproximação* de $f(x)$. Dada uma seqüência crescente de resoluções $\{r_j, j \in Z\}$ adotadas para o sinal, os detalhes com uma dada resolução r_j do sinal são definidos como a diferença de informação entre a aproximação com uma resolução r_j e a aproximação com resolução r_{j-1} , mas baixa [24].

Uma análise multiresolução consiste numa seqüência de espaços de aproximações sucessivas V_{2^j} , em que cada subespaço $V_{2^{j+1}}$ contém o subespaço V_{2^j} , tendo em mente que o sinal apresenta resolução cada vez menor no decorrer das aproximações, numa seqüência de estágios de processamentos consecutivos. Considere $f(x) \in L^2(R)$ como o sinal original e que o operador A_{2^j} seja responsável pela projeção de $f(x)$ no subespaço vetorial $V_{2^j} \subset L^2(R)$, gerando sua aproximação com resolução 2^j . O operador A_{2^j} é um operador de projeção no subespaço vetorial $V_{2^j} \subset L^2(R)$. O subespaço vetorial V_{2^j} pode ser interpretado como o conjunto de todas as possíveis aproximações de todas as funções $L^2(R)$ no nível de resolução 2^j .

Este capítulo destina-se também a estender a análise multiresolução em $L^2(R^2)$, para o caso 2D, formada por uma seqüência de subespaços $L^2(R^2)$, a partir de *wavelets* ortogonais, mas pode ser facilmente estendido para caso não-ortogonais, como por exemplo, para *wavelets* biortogonais conforme visto em [58] [59]. A compressão de imagens estáticas e vídeo apresentam excelentes resultados com a aplicação da TWD, e, dentro do contexto, a aproximação de uma imagem $f(x, y)$ no

nível de resolução 2^j é a sua projeção no subespaço vetorial $V_{2^j}^2$.

5.1 Propriedades da Multiresolução em $L^2(R)$

O operador A_{2^j} é responsável pela projeção de $f(x)$ gerando sua aproximação com resolução 2^j , ressaltando que o sinal original $f(x)$ é mensurável e tem energia finita: $f(x) \in L(R)$. Esta aproximação do sinal $f(x) \in L^2(R)$ com resolução 2^j é denotada por $A_{2^j} \cdot f(x)$ e caracteriza-se pelas seguintes propriedades principais [24]:

1. Se $A_{2^j} \cdot f(x)$ é a aproximação com resolução de 2^j da função $f(x)$ e se a operação for repetida no nível de resolução 2^j , tem-se que, tal aproximação não é modificada, que em outras palavras, implica que $A_{2^j} \cdot A_{2^j} = A_{2^j}$. O conjunto dos subespaços vetoriais V_{2^j} , representado como o conjunto de todas as possíveis aproximações de todas as funções $L^2(R)$ no nível de resolução 2^j , formam o espaço vetorial $\{V_{2^j}, j \in Z\}$ que será representado por $\{V_{2^j}\}$.
2. Dentre todas as aproximações da função $f(x)$ no nível de resolução 2^j , $A_{2^j} \cdot f(x)$ é a função que mais se aproxima de $f(x)$, ou seja:

$$\forall y(x) \in V_{2^j}, |y(x) - f(x)| \geq |A_{2^j} \cdot f(x) - f(x)| \quad (5.1)$$

O operador A_{2^j} calcula a projeção ortogonal em V_{2^j} .

3. A operação aproximação independe do nível de resolução. Os subespaços das aproximações podem ser obtidos a partir de outros subespaços, através de dilatação ou compressão das aproximações $f(x)$, implicando que:

$$\forall j \in Z, A_{2^j} \cdot f(x) \in V_{2^j} \rightarrow A_{2^j} \cdot f\left(\frac{x}{2}\right) \in V_{2^{j+1}} \text{ ou } A_{2^j} \cdot f(2x) \in V_{2^{j-1}} \quad (5.2)$$

4. Definindo que A_{2^j} é um operador de projeção em V_{2^j} , tem-se a seguinte hierarquia de subespaços:

$$\dots \subset V_{2^{-1}} \subset V_{2^0} \subset V_{2^1} \subset V_{2^2} \subset \dots \quad (5.3)$$

Implicando que a aproximação de $f(x)$ no nível de resolução 2^{j+1} contém toda a informação necessária para calcular a sua aproximação no nível de resolução 2^j .

5. A aproximação $A_{2^j} \cdot f(x)$ pode ser caracterizada por 2^j amostras por unidade de comprimento. Quando $f(x)$ é deslocada por um valor proporcional a 2^j , $A_{2^j} \cdot f(x)$ é deslocada do mesmo valor, implicando em uma invariância com o deslocamento.

6. Quando se determina uma aproximação de $f(x)$ no nível de resolução de 2^j , alguma informação da função original $f(x)$ é perdida, perda esta que é inversamente proporcional à resolução. Se a resolução aumentar em direção a $+\infty$, o sinal aproximação converge para o sinal original. Em contrapartida, à medida que a resolução diminui, o sinal aproximado passa a conter cada vez menos informação, convergindo para zero. Desta forma:

$$\lim_{j \rightarrow \infty} V_{2^j} = \bigcup_{j=-\infty}^{+\infty} V_{2^j} = L_2(R) \quad (5.4)$$

$$\lim_{j \rightarrow -\infty} V_{2^j} = \bigcap_{j=-\infty}^{+\infty} V_{2^j} = \{0\} \quad (5.5)$$

5.2 Análise Multiresolução Ortogonal

Definição 1

Denomina-se de aproximação por multiresolução qualquer conjunto de subespaços vetoriais $\{V_{2^j}\}$ que satisfaçam as propriedades 1-6 listadas na seção anterior [24]. O conjunto de operadores $\{A_{2^j}\}$ associados a este conjunto de espaços vetoriais gera as aproximações de qualquer função $f(x) \in L^2(R)$ em todos os níveis de resolução 2^j , para $j \in Z$.

Teorema 1

Seja $\{V_{2^j}, j \in Z\}$ uma aproximação por multiresolução em $L^2(R)$. Dado j , há um único operador A_{2^j} que faz a projeção ortogonal de $f(x)$ na base ortonormal de V_{2^j} . Existe uma única função $\phi(x) \in L^2(R)$, denominada função-escala, responsável pela formação de todas as bases ortonormais de todos os subespaços, tendo assim que:

$$\phi_{j,n}(x) = 2^{j/2} \phi(2^j x - n), n \in Z \quad (5.6)$$

Essa família será referenciada por $\{\phi_{j,n}\}$ e através da variação de n , tem-se uma base ortonormal para cada V_{2^j} .

5.2.1 O Subespaço Vetorial $\{V_{2^j}\}$ na Análise Multiresolução 1D

De acordo com o Teorema 1, dado j , existe um único operador A_{2^j} que realiza a projeção ortogonal de $f(x)$ na base ortonormal de V_{2^j} . Dentro deste contexto, a relação entre $\{\phi_{j,n}\}$ e V_{2^j} pode ser melhor

compreendida através da Fig. 5.1, a qual é exemplificada para $j \in \{-1, 0, 1\}$, onde está representada somente a base ortonormal de $V_{2^{-1}}$.

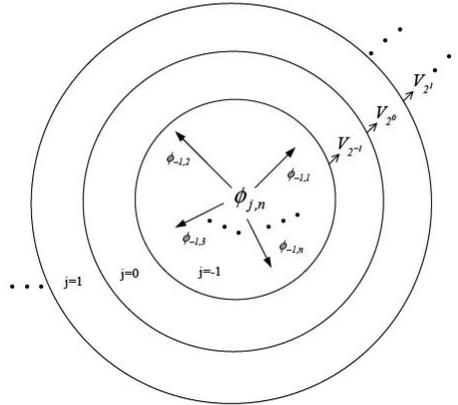


Figura 5.1: Esquema ilustrativo da relação entre $\phi_{j,n}(x)$ e V_{2^j} para $j \in [-1, 1]$ [3].

Com base neste exemplo, vale ressaltar que as bases ortonormais, embora diferentes, são geradas pela mesma função-escala $\phi(x)$ e referem-se a seus respectivos subespaços vetoriais.

Percebe-se que a aproximação por multiresolução $\{V_{2^j}, j \in \mathbb{Z}\}$ é completamente caracterizada pela função-escala $\phi(x)$, conforme visto pela Teorema 1. A função $\phi(x)$ deve ser continuamente diferenciável. O decaimento assintótico de $\phi(x)$ e de sua derivada $\phi'(x)$ no infinito deve satisfazer as seguintes exigências:

$$\phi(x) = O(x^{-2}) \text{ e } |\phi'(x)| = O(x^{-2}) \quad (5.7)$$

A projeção ortogonal de $f(x)$ no subespaço vetorial V_{2^j} pode ser obtida decompondo esta função na base ortonormal, definida pelo Teorema 1, de modo que:

$$\forall f(x) \in L^2(\mathbb{R}), A_{2^j} \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f, \phi_{j,n} \rangle \cdot \phi_{j,n}(x) \quad (5.8)$$

Em (5.9), $A_{2^j} \cdot f(x)$ refere-se à aproximação discreta de $f(x)$ na resolução 2^j .

$$A_{2^j}^d \cdot f = \{\langle f, \phi_{j,n} \rangle, n \in \mathbb{Z}\} \quad (5.9)$$

5.2.2 Implementação da Transformada Multiresolução Ortogonal: Decomposição (Análise)

É possível obter aproximações do sinal discreto original a partir de outras aproximações e, com o intuito de se obter a projeção em 2^j a partir da projeção em 2^{j+1} , leva-se em consideração que todas as aproximações do sinal discreto podem ser obtidas através de diferentes taxas de subamostragens do mesmo, sendo esta a base da análise multiresolução.

Estes sinais utilizados na prática são discretos e, por questão de normalização, possuem resolução máxima finita, igual a 1. Conseqüentemente, a aproximação discreta de $f(x)$ é representada para este caso como $A_1^d \cdot f$.

No caso da análise, o sinal de entrada é decomposto em uma aproximação e detalhes. A aproximação obtida será a próxima entrada da Fig. 5.2 resultando em nova aproximação e detalhes e assim sucessivamente. Cada aproximação $A_{2^j}^d \cdot f$ corresponde a um subespaço vetorial $\{V_{2^j}, j \in Z\}$, e os detalhes $D_{2^j}^d \cdot f$ correspondem aos seus respectivos subespaços vetoriais $\{W_{2^j}, j \in Z\}$. Cada nível de resolução é representado pelo nível j .

A partir de uma aproximação do sinal, supondo para o caso inicial a representação $A_1^d \cdot f$, todas as demais aproximações discretas $A_{2^j}^d \cdot f$ para $j < 0$ podem ser calculadas de acordo com o princípio da Propriedade 3, apresentada na seção 5.1. Esta seção destina-se a descrever algoritmos simples e iterativos para o cálculo destas aproximações discretas [24] [48].

A Fig. 5.2 ilustra como é realizada a decomposição do sinal de um determinado nível de resolução, resultando em um nível de resolução inferior a este, em que esta diferença de informação é denominada de detalhe.

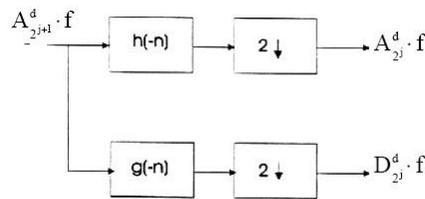


Figura 5.2: Diagrama em blocos do esquema de decomposição [24].

Seja $\{V_{2^j}, j \in Z\}$ uma aproximação por multiresolução e $\phi(x)$ a função-escala correspondente. Segundo o Teorema 1, a família $\{\phi_{j,n}\}$ é uma base ortonormal para cada V_{2^j} [24] [57]. Para qualquer $n \in Z$, a função $\{\phi_{j,n}\}$ pertence ao subespaço V_{2^j} e, que por sua vez, está contido em $V_{2^{j+1}}$ podendo resultar em uma expansão da função $\{\phi_{j,n}\}$ na base ortonormal de $V_{2^{j+1}}$, expressa por:

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{j,n}, \phi_{j+1,k} \rangle \phi_{j+1,k}(x) \quad (5.10)$$

Realizando uma mudança de variáveis no produto interno de (5.10), tem-se:

$$\begin{aligned}
\langle \phi_{j,n}, \phi_{j+1,k} \rangle &= \langle 2^{j/2} \phi(2^j u - n), 2^{(j+1)/2} \phi(2^{j+1} u - k) \rangle \\
&= 2^{j/2+(j+1)/2} \int_{-\infty}^{\infty} \phi(2^{-1} y) \cdot \phi(y - (k - 2n)) 2^{-j-1} dy \\
&= 2^{-1/2} \langle \phi(2^{-1} u), \phi(u - (k - 2n)) \rangle
\end{aligned} \tag{5.11}$$

obtendo deste modo:

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \phi_{j+1,k}(u) \tag{5.12}$$

Calculando o produto interno de $f(x)$ com ambos os lados de (5.12), obtém-se:

$$\langle f, \phi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} \langle \phi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \langle f(u), \phi_{j+1,k}(u) \rangle \tag{5.13}$$

A expressão (5.13) pode ser associada a um processo de filtragem, em que pode determinar o filtro de resposta impulsiva $h(m)$ como:

$$h(m) = \langle \phi_{-1,0}(u), \phi_{0,m}(u) \rangle, \forall n \in Z \tag{5.14}$$

Desta forma, pode-se escrever (5.12) como:

$$\langle f, \phi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} h(k - 2n) \cdot \langle f, \phi_{j+1,k} \rangle \tag{5.15}$$

Fica evidenciado, desta forma, que a aproximação $A_{2^j}^d \cdot f$ pode ser obtida realizando a convolução de $A_{2^{j+1}}^d \cdot f$ com $h(-n)$, onde, em seguida, deve ser realizada a dizimação por um fator de 2 em cada etapa que o sinal é decomposto. Em outras palavras, tendo como ponto de partida o sinal de maior resolução $A_1^d \cdot f$, todas as aproximações $A_{2^j}^d \cdot f$, para $j < 0$, podem ser obtidas por intermédio da aplicação recursiva, procedimento este denominado de codificação piramidal [60].

Utilizando (5.14), a equação (5.12) pode ser reescrita como:

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} h(k - 2n) \cdot \phi_{j+1,k} \tag{5.16}$$

E assim, fazendo $j = 0$ e $n = 0$ em (5.16), obtém-se:

$$\phi(x) = \sum_{k=-\infty}^{\infty} h(k) \cdot 2^{1/2} \phi(2x - k) \quad (5.17)$$

De modo recursivo, a TF de (5.17) pode ser obtida, utilizando:

$$\phi(w) = \sum_{k=-\infty}^{\infty} h(k) \frac{\phi(0, 5w)}{\sqrt{2}} e^{-i0,5wk} = \frac{\phi(0, 5w)}{\sqrt{2}} H(0, 5w) \quad (5.18)$$

O filtro $h(n)$ possui a seguinte TF:

$$H(w) = \sum_{n=-\infty}^{\infty} h(n) e^{-inw} \quad (5.19)$$

Este filtro $H(w)$ deve satisfazer aos seguintes equacionamentos [57]:

$$H(0) = \sum_{k=-\infty}^{\infty} h(k) = \sqrt{2} \text{ e } h(n) = O(x^{-2}), \quad x \rightarrow \infty \quad (5.20)$$

$$|H(w)|^2 + |H(w + \pi)|^2 = 2, \quad \forall w \quad (5.21)$$

$$|H(w)| \neq 0, \text{ para } w \in [0, \pi/2] \quad (5.22)$$

E, a partir de (5.18), a TF da função-escala pode ser definida como:

$$\phi(w) = \phi(0) \cdot \prod_{m=1}^{\infty} \frac{H(w \cdot 2^{-m})}{\sqrt{2}} \quad (5.23)$$

Segundo Porat [61], para qualquer análise multiresolução, tem-se que $\phi(0) = 1$, o que implica em dizer que a família de *wavelets* de funções $\{\phi_{j,n}\}$ não são *wavelets*, por não obedecer à condição de admissibilidade. Tais propriedades evidenciam que o filtro $H(w)$ refere-se a um passa-baixas, em que, este conjunto de restrições apresentado a $h(k)$ possibilita determinar $\phi(x)$ para que seja bem localizado em x .

5.2.3 O Subespaço Vetorial $\{W_{2^j}\}$ na Análise Multiresolução 1D

De posse da apresentação referente ao conceito do subespaço vetorial V_{2^j} , vale destacar que o sinal pertencente ao subespaço vetorial $V_{2^{j+1}}$ possui maior resolução do que no subespaço vetorial V_{2^j} e assim sucessivamente. A diferença de informação (sinal de detalhamento) entre estes subespaços

vetoriais pode ser compreendida analisando o subespaço vetorial que é o complemento ortogonal do subespaço vetorial V_{2^j} , W_{2^j} :

$$V_{2^j} \oplus W_{2^j} = V_{2^{j+1}} \quad (5.24)$$

Conforme percebido em (5.24), este sinal de detalhamento consiste na diferença de informação entre os níveis de resolução 2^j e 2^{j+1} . A Fig. 5.3 ilustra a hierarquia de relações entre os subespaços vetoriais V_{2^j} e W_{2^j} .

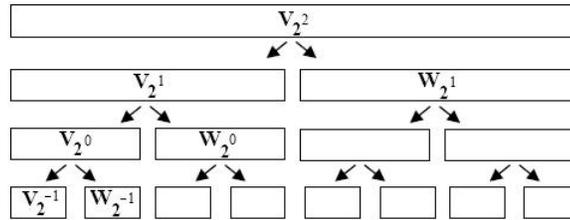


Figura 5.3: Espaço vetorial W_{2^j} .

A função $\psi(x)$ está relacionada com W_{2^j} da mesma forma que $\phi(x)$ está com V_{2^j} e, de forma análoga, uma base ortonormal para gerar o subespaço W_{2^j} torna-se necessária para calcular a projeção de $f(x)$ em W_{2^j} , implicando desta forma em encontrar uma função $\psi(x)$ capaz de gerar o subespaço W_{2^j} .

Deseja-se que a família presente em (5.25), que será referenciada apenas como $\{\psi_{j,n}\}$ no decorrer do texto, seja uma base ortonormal para W_{2^j} [48][24].

$$\{\psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n), n \in Z\} \quad (5.25)$$

Assumindo a existência de $\psi(x)$, para qualquer $n \in Z$, a função (5.25) pertence a $W_{2^j} \subset V_{2^{j+1}}$ e, deste modo, pode-se expandir $\psi_{j,n}$ na base ortonormal de $V_{2^{j+1}}$:

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \psi_{j,n}, \phi_{j+1,k} \rangle \cdot \phi_{j+1,k}(x) \quad (5.26)$$

De modo análogo à seção anterior, realizando-se a mudança de variáveis no produto interno de (5.26), tem-se:

$$\begin{aligned} \langle \psi_{j,n}, \phi_{j+1,k} \rangle &= 2^{j/2+(j+1)/2} \int_{-\infty}^{\infty} \psi(2^{-1}y) \cdot \phi(y - (k - 2n)) 2^{-j-1} dy \\ &= 2^{-1/2} \langle \psi(2^{-1}u), \phi(u - (k - 2n)) \rangle \end{aligned} \quad (5.27)$$

Obtendo-se assim, com a substituição de (5.27) em (5.26):

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \psi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \phi_{j+1,k}(x) \quad (5.28)$$

A equação (5.28) pode ser encarada como uma filtragem. O filtro discreto $g(n)$, de resposta impulsiva, pode ser determinado para qualquer $n \in Z$ como:

$$g(m) = \langle \psi_{-1,0}(u), \phi_{0,m}(u) \rangle \quad (5.29)$$

E, desta forma, $\psi_{j,n}(x)$ presente em (5.28) pode ser reescrito como:

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} g(k-2n) \cdot \phi_{j+1,k}(x) \quad (5.30)$$

Calculando-se cada coeficiente resultante da projeção de $f(x)$, através da aplicação do produto interno de $f(x)$, com ambos os lados de (5.26) e utilizando (5.29), tem-se:

$$\langle f, \psi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} g(k-2n) \cdot \langle f, \phi_{j+1,k} \rangle \quad (5.31)$$

Tomando $j = 0$ e $n = 0$ em (5.30), obtém-se a expressão que determina a função-escala:

$$\psi(x) = \sum_{k=-\infty}^{\infty} g(k) \cdot 2^{1/2} \phi(2x - k) \quad (5.32)$$

Em (5.32), $g(k)$ refere-se a uma seqüência quadraticamente somável. Realizando a TF de ambos os lados de (5.32), pode-se obter:

$$\psi(w) = \sum_{k=-\infty}^{\infty} g(k) \frac{\phi(0, 5w)}{2^{1/2}} e^{-j0,5wn} = \frac{\phi(0, 5w)}{2^{1/2}} G(0, 5w) \quad (5.33)$$

Lema 1

A família $\{\psi_{j,n}\}_{n \in Z}$ é uma base ortonormal de W_{2^j} se, e, somente se [61]:

$$|G(w)|^2 + |G(w + \pi)|^2 = 2, \quad \forall w \quad (5.34)$$

$$H(w)G^*(w) + H(w + \pi)G^*(w + \pi) = 0 \quad (5.35)$$

A fim de tornar evidente que $\{\psi_{j,k}\}_{k \in Z}$ é uma base ortonormal, a condição de admissibilidade

para $\psi(x)$ deve ser testada. Substituindo (5.20) em (5.21), tem-se:

$$\begin{aligned} |H(0)|^2 + |H(\pi)|^2 &= |(\sqrt{2})|^2 + |H(\pi)|^2 = 2 \\ |H(\pi)|^2 &= 0 \end{aligned} \quad (5.36)$$

Substituindo (5.20) e (5.36) em (5.35), tem-se:

$$G(0) = 0 \text{ e } \psi(0) = \frac{\phi(0)}{\sqrt{2}} \cdot G(0) = 0 \quad (5.37)$$

Deste modo, pode se comprovar que $\psi(w)$ satisfaz a condição de admissibilidade e conseqüentemente, a família $\{\psi_{j,n}\}$ é uma base *wavelet*.

Similarmente ao que foi realizado para a aproximação do sinal, o sinal de detalhamento de $f(x)$, $D_{2^j} \cdot f(x)$, apresentando resolução 2^j pode ser representado por:

$$D_{2^j} \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f(u), \psi_{j,n}(u) \rangle \cdot \psi_{j,n}(x) \quad (5.38)$$

em que o conjunto de produtos internos determinam os coeficientes deste sinal de detalhamento, dado por:

$$D_{2^j}^d \cdot f = \{ \langle f, \psi_{j,n} \rangle, n \in Z \} \quad (5.39)$$

Conforme já enfocado, o sinal de detalhamento contém a diferença de informação entre $A_{2^{j+1}}^d \cdot f$ e $A_{2^j}^d \cdot f$ e a mesma é representada por $D_{2^j}^d \cdot f$.

5.2.3.1 Relação entre os Filtros

A relação entre os filtros $g(n)$ e $h(n)$ é a base da análise multiresolução e, deste modo, merecem destaque nesta seção.

Uma vez fixado $H(w)$, $G(w)$ não é mais arbitrário e visando provar esta afirmação pode-se definir que:

$$H_1 = H(w) \text{ e } H_2 = H(w + \pi) \quad (5.40)$$

$$G_1 = G(w) \text{ e } G_2 = G(w + \pi) \quad (5.41)$$

o que resulta em:

$$\frac{G_1}{G_2} = - \left(\frac{H_2}{H_1} \right)^* \quad (5.42)$$

Dividindo ambos os lados de (5.42) por $\left[1 + \left|\frac{G_1}{G_2}\right|^2\right]^{1/2}$ e $\left[1 + \left|\frac{H_2}{H_1}\right|^2\right]^{1/2}$, respectivamente, tem-se:

$$\frac{\frac{G_1}{G_2}}{\left[1 + \left|\frac{G_1}{G_2}\right|^2\right]^{1/2}} = - \frac{\left(\frac{H_2}{H_1}\right)^*}{\left[1 + \left|\frac{H_2}{H_1}\right|^2\right]^{1/2}} \quad (5.43)$$

Substituindo (5.21) e (5.34) em (5.43), obtém-se uma importante relação entre os filtros expressa por:

$$G_1 = \frac{|H_1| G_2}{|G_2| H_1^*} \cdot H_2^* = -A(w) \cdot H_2^* \quad (5.44)$$

Como $|A(w)| = 1$ se refere a um filtro passa-tudo, pode-se afirmar que $|G_1| = |H_2|$, implicando assim que a magnitude do espectro de $g(n)$ é igual ao espectro de $h(n)$ deslocado de π .

Substituindo (5.44) em (5.35), verifica-se que:

$$A(w) + A(w + \pi) = 0 \quad (5.45)$$

Freqüentemente a escolha de $A(w) = 1$ é bastante usual, já que esta escolha implica em $|A(w)| = 1$. Substituindo em (5.44), obtém-se:

$$G(w) = e^{-jw} \cdot H(w + \pi) \quad (5.46)$$

que no domínio temporal, pode ser representado por:

$$g(n) = (-1)^{n-1} \cdot h^*(1-n) \quad (5.47)$$

Com base nestas expressões apresentadas, qualquer par de filtros $h(n)$ e $g(n)$, que obedecer tais imposições, podem ser utilizados no processo de decomposição e síntese da TW. Usualmente $h(n)$ é escolhido como filtro passa-baixa, o que implica conseqüentemente em dizer que $g(n)$ refere-se a um filtro passa-alta. As funções $\psi(x)$ e $\phi(x)$ são determinadas após tais filtros serem determinados.

Deste modo, pode-se ser enunciado o seguinte teorema:

Teorema 2

Considerando $\{V_{2^j}\}$ como o conjunto de subespaços vetoriais de uma análise multiresolução, $\phi(x)$ referindo à função-escala e $h(n)$ ao filtro em questão, pode-se expressar que a função $\psi(x)$

apresenta a seguinte TF:

$$\psi(w) = G(0, 5w) \phi(0, 5w) 2^{-1/2} \quad (5.48)$$

em que:

$$G(w) = e^{-iw} \cdot H^*(w + \pi) \text{ e } \psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n) \quad (5.49)$$

Desta forma, implica que:

$$\psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n), \quad n \in Z \quad (5.50)$$

Em (5.50), $\{\psi_{j,n}\}$ é uma base ortonormal em W_{2^j} [48].

Pode ser observado pelo Teorema 2 a relação existente entre a *wavelet* e a função-escala. A partir de um par de filtros $h(n)$ e $g(n)$ que satisfaçam as condições (5.20) - (5.22) pode-se encontrar a função-escala e a *wavelet*-básica utilizando as expressões (5.23) e (5.48), respectivamente.

5.2.4 Implementação da *Wavelet* Multiresolução Ortogonal: Reconstrução (Síntese)

O sinal discreto original pode ser reconstruído e este é o objetivo em questão. O espaço vetorial definido para a representação *wavelet* ortogonal é completo. Como W_{2^j} é o complemento ortogonal de V_{2^j} em $V_{2^{j+1}}$, $(\{\phi_{j,n}(x)\}, \{\psi_{j,n}(x)\})_{n \in Z}$ é uma base ortonormal em V_2^{j+1} e, deste modo, $\phi_{j+1,n}(x)$ pode ser representado por:

$$\phi_{j+1,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \phi_{j,k} \rangle \cdot \phi_{j,k}(x) + \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \psi_{j,k} \rangle \cdot \psi_{j,k}(x) \quad (5.51)$$

Calculando-se o produto interno de ambos os lados de (5.51) com a função $f(x)$, tem-se:

$$\langle f(x), \phi_{j+1,n}(x) \rangle = \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \phi_{j,k} \rangle \cdot \langle f(x), \phi_{j,k}(x) \rangle + \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \psi_{j,k} \rangle \cdot \langle f(x), \psi_{j,k}(x) \rangle \quad (5.52)$$

Realizando uma mudança de variáveis nos produtos internos de (5.52), obtém-se:

$$\langle \phi_{j+1,n}(u), \phi_{j,k}(u) \rangle = \langle \phi_{-1,0}(u), \phi_{0,n-2k}(u) \rangle = h((n - 2k)) \quad (5.53)$$

$$\langle \phi_{j+1,n}(u), \psi_{j,k}(u) \rangle = \langle \psi_{-1,0}(u), \phi_{0,n-2k}(u) \rangle = g(n - 2k) \quad (5.54)$$

Substituindo (5.53) e (5.54) em (5.52), a mesma reescrita em função dos filtros $h(n)$ e $g(n)$, obtendo:

$$\langle f(x), \phi_{j+1,n}(x) \rangle = \sum_{k=-\infty}^{\infty} h(n-2k) \cdot \langle f(x), \phi_{j,k}(x) \rangle + \sum_{k=-\infty}^{\infty} g(n-2k) \cdot \langle f(x), \psi_{j,k}(x) \rangle \quad (5.55)$$

A equação (5.55) mostra que $A_{2^{j+1}}^d \cdot f$ pode ser construída por super-amostragem, ou seja, com a inserção de um zero entre cada amostra $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$ e convoluindo os resultados com $h(n)$ e $g(n)$, respectivamente. Cada coeficiente de $A_{2^{j+1}}^d \cdot f$ é formado por todos os coeficientes de $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$.

Na Fig. 5.4, encontra-se o diagrama em blocos da reconstrução do sinal. No algoritmo de decomposição utiliza-se $h(-n)$ e $g(-n)$, enquanto na síntese utiliza-se filtros de resposta impulsiva $h(n)$ e $g(n)$. Contudo, a escolha arbitrária pode levar a *wavelets* sem o comportamento desejado. Nesse contexto, diversos pesquisadores se aprofundaram na construção de *wavelets* que obedecessem a determinadas características importantes para as mais diversas aplicações, elaborando algoritmos partindo de $h(n)$ e encontrando $\phi(x)$, $g(n)$ e $\psi(x)$.

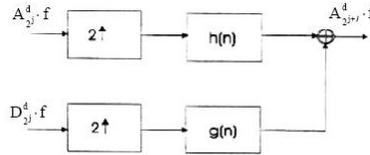


Figura 5.4: Diagrama em blocos do esquema de reconstrução da representação *wavelet* [24].

5.3 Análise Multiresolução Bidimensional

A análise multiresolução para o caso 2D segue os mesmos princípios da seção anterior. Porém, o subespaço vetorial $\{V_{2^j}^1\}$ é agora representado por $\{V_{2^j}^2, j \in Z\}$ e, qualquer conjunto desses subespaços vetoriais que satisfazer as propriedades da análise multiresolução contidas na Seção 5.1. Para o caso bidimensional, são denominados de aproximação por multiresolução em $L^2(R^2)$. O subespaço vetorial $V_{2^j}^2$ é definido como o produto tensorial dado por:

$$V_{2^j}^2 = V_{2^j}^1 \otimes V_{2^j}^1 \quad (5.56)$$

Estendendo o Teorema 1 para o caso bidimensional, pode-se mostrar que existe uma única função-escala bidimensional $\phi(x, y) \in L^2(R)$ responsável pela formação de todas as bases ortogonais de todos os subespaços em questão [24]. Deste modo, a família $\{\phi_{j;n_x,n_y}(x, y)\}$ responsável pela formação de uma base ortonormal para cada $V_{2^j}^2$ é representada por:

$$\{\phi_{j;n_x,n_y}(x,y) = 2^j \phi(2^j x - n_x, 2^j y - n_y), (n_x, n_y) \in \mathbb{Z}^2\} \quad (5.57)$$

Foi visto na seção anterior, que $\phi(x)$ é a função-escala unidimensional referente a $\{V_{2^j}^1\}$. A função-escala bidimensional $\phi(x,y)$ ortogonal se separável, pode ser expressa por [62]:

$$\phi(x,y) = \phi(x) \cdot \phi(y) \quad (5.58)$$

Para a base ortogonal em $\{V_{2^j}^2\}$, tem-se:

$$\phi_{j;n_x,n_y}(x,y) = \phi_{j;n_x}(x) \cdot \phi_{j;n_y}(y), (n_x, n_y) \in \mathbb{Z}^2 \quad (5.59)$$

A aproximação de uma imagem $f(x,y)$ no subespaço vetorial $\{V_{2^j}^2\}$ é obtida de forma análoga à seção anterior, sendo representada por:

$$A_{2^j} \cdot f(x,y) = \sum_{n_x=-\infty}^{\infty} \sum_{n_y=-\infty}^{\infty} \langle f(u,v), \phi_{j;n_x,n_y}(u,v) \rangle \cdot \phi_{j;n_x,n_y}(x,y) \quad (5.60)$$

Logo, a aproximação discreta de $f(x,y)$ no nível de resolução 2^j é caracterizada pelo seguinte conjunto de produtos internos:

$$A_{2^j}^d \cdot f = \{\langle f(u,v), \phi_{j;n_x}(u) \cdot \phi_{j;n_y}(v) \rangle, (n_x, n_y) \in \mathbb{Z}^2\} \quad (5.61)$$

5.3.1 Implementação da Transformada *Wavelet* Bidimensional: Decomposição

Os sinais bidimensionais $f(x,y)$, para o caso em questão, são voltados para imagens discretas de resolução finita. No caso 2D, e, de modo análogo ao caso 1D, a resolução máxima finita igual a 1 se mantém por questão de normalização. A aproximação discreta de uma imagem $f(x,y)$ para esta resolução será $A_1^d \cdot f(x,y)$.

A família $\{\phi_{j;n_x,n_y}(x,y)\}$ é uma base ortonormal para cada $V_{2^j}^2$, em que $\{\phi_{j;n_x,n_y}(x,y)\} \in V_{2^j}^2 \subset V_{2^{j+1}}^2$. Portanto, expandir $\{\phi_{j;n_x,n_y}(x,y)\}$ na base ortonormal $\{\phi_{j+1;k_x,k_y}(x,y)\}$ de $V_{2^{j+1}}^2$ resulta em:

$$\phi_{j;n_x,n_y}(x,y) = \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j;n_x,n_y}(u,v), \phi_{j+1;k_x,k_y}(u,v) \rangle \cdot \phi_{j+1;k_x,k_y}(x,y) \quad (5.62)$$

Deve-se notar que essa expansão é análoga ao caso 1D e, realizando uma pequena mudança de variáveis no produto interno de (5.62), tem-se que cada coeficiente resultante da projeção de $f(x, y)$ pode ser obtido por:

$$\begin{aligned} \langle f(x, y) \cdot \phi_{j;n_x,n_y}(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle f(x, y) \cdot \phi_{j+1;k_x,k_y}(x, y) \rangle \cdot \\ &\cdot \langle \phi_{-1;0,0}(u, v), \phi_{0;(k_x-2n_x),(k_y-2n_y)}(u, v) \rangle \end{aligned} \quad (5.63)$$

A equação (5.63) pode ser interpretada como um processo de filtragem, onde pode ser definido $h(l, m)$ como um filtro discreto bidimensional de resposta impulsiva:

$$h(l, m) = \langle \phi_{-1;0,0}(u, v), \phi_{0;l,m}(u, v) \rangle = \langle \phi_{-1,0}(u), \phi_{0,l}(u) \rangle \cdot \langle \phi_{-1,0}(v), \phi_{0,m}(v) \rangle \quad (5.64)$$

Para o processamento de imagens, a utilização de uma aproximação por multiresolução separável dá ênfase às direções verticais e horizontais, representadas pelas variáveis x e y . Nesse contexto, o filtro $h(l, m)$, por apresentar esta característica, pode ser decomposto no produto de dois filtros digitais unidimensionais definidos em (5.14).

Substituindo (5.64) em (5.63) e convoluindo o resultado com $f(x, y)$, obtém-se:

$$\begin{aligned} \langle f(x, y) \cdot \phi_{j;n_x,n_y}(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} h(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} h(k_x - 2n_x) \cdot \\ &\cdot \langle f(x, y) \cdot \phi_{j+1;k_x,k_y}(x, y) \rangle \end{aligned} \quad (5.65)$$

Realizando uma comparação entre (5.61) e (5.65), pode-se comprovar que $A_{2^j}^2 \cdot f$ pode ser calculada através de filtragens sucessivas de $A_{2^{j+1}}^2 \cdot f$ em ambas as direções. O primeiro passo consiste em filtrar $A_{2^{j+1}}^2 \cdot f$ na direção horizontal, que corresponde à direção x (linhas) com o filtro unidimensional $h(-n)$, dizimando o resultado por um fator de 2 nessa direção, que em outras palavras, consiste em descartar uma linha e outra não. No passo seguinte, o resultado é filtrado na direção vertical, que por sua vez, consiste à direção y (colunas) com o mesmo $h(-n)$, sendo aplicado novamente o descarte de uma amostra sim e outra não nessa direção. No ramo superior da Fig. 5.5 ilustra-se a esquematização deste processo.

Se $A_1^d \cdot f(x, y)$ possuir N pixels, para $j < 0$, a sua aproximação discreta com resolução 2^j representada por $A_{2^{j+1}}^2 \cdot f$ irá possuir $2^j \cdot N$ pixels. Todas as aproximações $A_{2^j}^d \cdot f$ são obtidas recursivamente tendo como ponto de partida $A_1^d \cdot f$, em que pode ser realizado primeiramente o processamento

por colunas e, em seguida, por linhas, à semelhança do que ocorre com as demais transformadas bidimensionais.

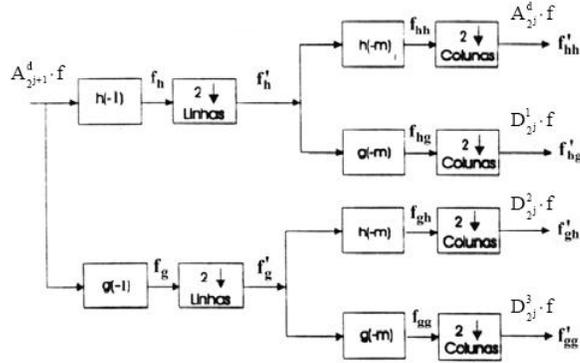


Figura 5.5: Diagrama de blocos de um estágio de decomposição *wavelet* de uma imagem [24].

Nesse contexto, e de modo análogo ao caso 1D, o complemento ortogonal de $V_{2^j}^2$ em $V_{2^{j+1}}^2$ refere-se ao sinal de detalhamento que, no nível de resolução 2^j , corresponde à projeção ortogonal do sinal no subespaço $W_{2^j}^2$. Vale comentar que, na Fig. 5.5 também podem ser observados os componentes de detalhamento.

Teorema 3

Seja $\{V_{2^j}^2\}$ uma aproximação por multiresolução separável em $L^2(\mathbb{R}^2)$. Considerando (5.58) e sendo $\psi(x)$ a *wavelet* unidimensional associada a $\phi(x)$ tem-se, deste modo, 3 *wavelets* bidimensionais responsáveis pela formação das bases ortonormais [17] [24] representadas por:

$$\begin{aligned}\psi^1(x, y) &= \phi(x) \cdot \psi(y) \\ \psi^2(x, y) &= \psi(x) \cdot \phi(y) \\ \psi^3(x, y) &= \psi(x) \cdot \psi(y)\end{aligned}\tag{5.66}$$

Baseado no Teorema 3, a diferença de informação entre $A_{2^{j+1}}^d \cdot f$ e $A_{2^j}^d \cdot f$ é igual à projeção ortogonal de $f(x, y)$ no subespaço $W_{2^j}^2$, em que a diferença de informação é dada pelas 3 imagens de detalhes:

$$\begin{aligned}
D_{2^j}^1 \cdot f &= \left\{ \langle f(x, y), 2^j \psi^1(2^j x - n_x, 2^j y - n_y) \rangle \right\}, (n_x, n_y) \in Z^2 \\
D_{2^j}^2 \cdot f &= \left\{ \langle f(x, y), 2^j \psi^2(2^j x - n_x, 2^j y - n_y) \rangle \right\}, (n_x, n_y) \in Z^2 \\
D_{2^j}^3 \cdot f &= \left\{ \langle f(x, y), 2^j \psi^3(2^j x - n_x, 2^j y - n_y) \rangle \right\}, (n_x, n_y) \in Z^2
\end{aligned} \tag{5.67}$$

5.3.2 Representação da Transformada Wavelet Ortogonal Bidimensional

Considerando que o Teorema 3 seja válido, existe um conjunto de *wavelets* separáveis $\left\{ \psi_{j;n_x,n_y}^i(x, y) \right\}$, onde i indica o índice da *wavelet* utilizada e, para o caso em questão, está limitado em $1 \leq i \leq 3$, sendo que cada *wavelet* pertence ao mesmo espaço $W_{2^j}^2 \subset V_{2^{j+1}}^2$. Assim, pode-se expandir $\left\{ \psi_{j;n_x,n_y}^i(x, y), 1 \leq i \leq 3 \right\}$ na base ortonormal de $V_{2^{j+1}}^2$:

$$\psi_{j;n_x,n_y}^i(x, y) = \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \left\langle \psi_{j;n_x,n_y}^i(u, v), \phi_{j+1;k_x,k_y}(u, v) \right\rangle \cdot \phi_{j+1;k_x,k_y}(x, y) \tag{5.68}$$

Realizando uma mudança de variáveis em (5.68), tem-se:

$$\psi_{j;n_x,n_y}^i(x, y) = \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \left\langle \psi_{-1;0,0}^i(u, v), \phi_{0;(k_x-2n_x),(k_y-2n_y)}(u, v) \right\rangle \cdot \phi_{j+1;k_x,k_y}(x, y) \tag{5.69}$$

A equação (5.69) pode ser interpretada como um processo de filtragem, em que pode-se definir o filtro discreto bidimensional de resposta impulsiva, $g^i(l, m)$, como:

$$g^i(l, m) = \left\langle \psi_{-1;0,0}^i(u, v), \phi_{0;l,m}(u, v) \right\rangle \tag{5.70}$$

Para cada *wavelet* i existe um filtro separável correspondente que pode ser decomposto no produto de dois filtros 1D, $h(m)$ e $g(m)$. Para o caso particular de limitar as *wavelets* a $1 \leq i \leq 3$, pode-se

definir os seguintes filtros bidimensionais:

$$\begin{aligned}
 g^1(l, m) &= \langle \phi_{-1,0}(u) \psi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \rangle \\
 &= \langle \phi_{-1,0}(u), \phi_{0,l}(u) \rangle \cdot \langle \psi_{-1,0}(v), \phi_{0,m}(v) \rangle \\
 &= h(l) \cdot g(m)
 \end{aligned} \tag{5.71}$$

$$\begin{aligned}
 g^2(l, m) &= \langle \psi_{-1,0}(u) \phi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \rangle \\
 &= \langle \psi_{-1,0}(u), \phi_{0,l}(u) \rangle \cdot \langle \phi_{-1,0}(v), \phi_{0,m}(v) \rangle \\
 &= g(l) \cdot h(m)
 \end{aligned} \tag{5.72}$$

$$\begin{aligned}
 g^3(l, m) &= \langle \psi_{-1,0}(u) \psi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \rangle \\
 &= \langle \psi_{-1,0}(u), \phi_{0,l}(u) \rangle \cdot \langle \psi_{-1,0}(v), \phi_{0,m}(v) \rangle \\
 &= g(l) \cdot g(m)
 \end{aligned} \tag{5.73}$$

As equações (5.71)-(5.73) evidenciam que esses filtros bidimensionais são separáveis e resultam em produtos de filtros unidimensionais que operam em duas direções (x, y) distintas. Com a obtenção de tais filtros, pode-se obter o processo de filtragem por:

$$\begin{aligned}
 \langle f(x, y) \cdot \psi_{j;n_x,n_y}^1(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} g(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} h(k_x - 2n_x) \cdot \\
 &\quad \cdot \langle f(x, y) \cdot \phi_{j+1;n_x,n_y}(x, y) \rangle
 \end{aligned} \tag{5.74}$$

$$\begin{aligned}
 \langle f(x, y) \cdot \psi_{j;n_x,n_y}^2(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} h(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} g(k_x - 2n_x) \cdot \\
 &\quad \cdot \langle f(x, y) \cdot \phi_{j+1;n_x,n_y}(x, y) \rangle
 \end{aligned} \tag{5.75}$$

$$\begin{aligned}
 \langle f(x, y) \cdot \psi_{j;n_x,n_y}^3(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} g(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} g(k_x - 2n_x) \cdot \\
 &\quad \cdot \langle f(x, y) \cdot \phi_{j+1;n_x,n_y}(x, y) \rangle
 \end{aligned} \tag{5.76}$$

deixando evidente que as diferentes imagens de detalhes $D_{2^j}^d \cdot f$ podem ser obtidas por meio de filtragens sucessivas de $A_{2^{j+1}}^d \cdot f$ em ambas as direções. O filtro utilizado em cada direção depende do índice i da *wavelet* a ser utilizada e, conforme já enfocado, deve-se filtrar $A_{2^{j+1}}^d \cdot f$ na direção das linhas com um filtro 1D, dizimando o resultado por uma fator de 2 e, em seguida, filtrar o resultado na direção das colunas com outro filtro 1D, seguido novamente por uma dizimação por um fator de 2.

A representação *wavelet* ortogonal 2D de uma imagem original $A_1^d \cdot f$ pode ser obtida por intermédio de sucessivas decomposições de $A_{2^{j+1}}^d \cdot f$ em $A_{2^j}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$ e $D_{2^j}^3 \cdot f$, para $-J \leq j \leq -1$.

A Fig. 5.5 ilustra o estágio completo deste algoritmo. A representação *wavelet* bidimensional pode, desta forma, ser calculada através de dois algoritmos piramidais unidimensionais, sendo um aplicado às linhas e outro às colunas da imagem.

Uma imagem $A_1^d \cdot f(x, y)$, para $J > 0$, é completamente representada pelas $3J + 1$ imagens discretas:

$$\left(A_{2^{-J}}^d \cdot f, \{D_{2^j}^1 \cdot f\}_{-J \leq j \leq -1}, \{D_{2^j}^2 \cdot f\}_{-J \leq j \leq -1}, \{D_{2^j}^3 \cdot f\}_{-J \leq j \leq -1} \right) \quad (5.77)$$

A imagem $A_{2^{-J}}^d \cdot f$ refere-se às frequências mais baixas de $f(x, y)$ e corresponde à aproximação de $A_1^d \cdot f$ com nível de resolução 2^{-J} . Adotando três estágios ($j = 3$), conforme ilustrado pela Fig. 5.6, o parâmetro $D_{2^j}^1 \cdot f$ corresponde às frequências verticais altas (linhas horizontais), assim como $D_{2^j}^2 \cdot f$ corresponde às frequências horizontais altas (linhas verticais) e, finalmente, $D_{2^j}^3 \cdot f$ corresponde às frequências altas em ambas as direções.

$A_{2^{-3}}^d \cdot f$	$D_{2^{-3}}^1 \cdot f$	$D_{2^{-2}}^1 \cdot f$	$D_{2^{-1}}^1 \cdot f$
$D_{2^{-3}}^2 \cdot f$	$D_{2^{-3}}^3 \cdot f$		
$D_{2^{-2}}^2 \cdot f$	$D_{2^{-2}}^3 \cdot f$		
$D_{2^{-1}}^2 \cdot f$		$D_{2^{-1}}^3 \cdot f$	

Figura 5.6: Imagens resultantes da decomposição *wavelet* ortogonal bidimensional para 3 níveis [24].

Para melhor demonstrar o efeito deste algoritmo em uma imagem, a imagem Mandrill utilizada nas simulações foi decomposta em 3 níveis ($j = 3$), sendo a imagem original visualizada na Fig. 5.7 e sua respectiva representação *wavelet* ortogonal bidimensional ilustrada na Fig. 5.8.

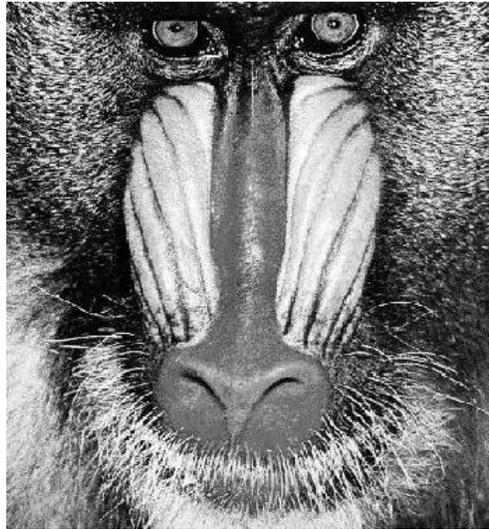


Figura 5.7: Imagem original Mandrill.

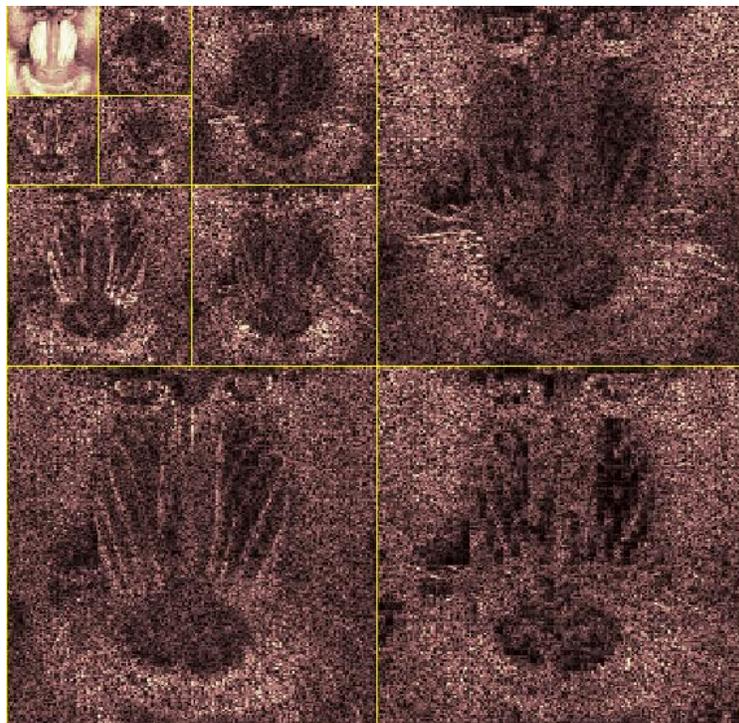


Figura 5.8: Representação *wavelet* ortogonal bidimensional para $j=3$.

Naturalmente, portanto, a maior concentração de energia da imagem situa-se na subimagem aproximação, decaindo à medida em que eleva-se o nível de resolução 2^{-j} . No presente trabalho foram realizadas simulações com imagens-teste comumente utilizadas na literatura a título ilustrativo para evidenciar estas características.

	Aprox.	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Db1	98,4640%	0,8848%	0,4375%	0,2137%
Db2	99,0775%	0,5814%	0,2407%	0,1004%
Db3	99,2059%	0,5289%	0,1930%	0,0722%
Db4	99,3469%	0,4222%	0,1739%	0,0570%
Db5	99,4219%	0,3812%	0,1473%	0,0497%
Db6	99,4973%	0,3260%	0,1325%	0,0441%
Db7	99,5353%	0,3022%	0,1224%	0,0401%
Db8	99,5696%	0,2940%	0,1003%	0,0362%

Tabela 5.1: Distribuição de energia dos coeficientes para Imagem Bird 256x256.

A porcentagem da concentração de energia dos componentes de aproximação e detalhes são apresentadas na Tabela 5.1 para a imagem Bird 256x256, considerando todas as *wavelets* de Daubechies testadas. Note-se que a porcentagem da concentração de energia na subimagem de aproximação, por exemplo, foi considerada como sendo uma variável aleatória (v.a) que varia com a TWD, obtendo-se para ela um valor médio de 99,2647% considerando todas as *wavelets* de Daubechies testadas. Deste modo, para esta v.a. obtém-se um desvio padrão de 0,3412% (medido em porcentagem dada a natureza da v.a.). Procedimento similar foi realizado para as componentes de detalhes de 1º, 2º, e 3º nível.

O mesmo procedimento também foi adotado para as demais *wavelets* e para as demais imagens-teste. Para a família de *wavelets* de Daubechies foram utilizadas as de ordens 1 à 8. Para as famílias Symlets e Biortogonais, todas as *wavelets* destas famílias foram utilizadas na construção das tabelas.

Assim, a porcentagem de concentração de energia dos componentes de aproximação e detalhes são apresentadas, respectivamente, nas Tabelas 5.2-5.4 para um conjunto de imagens testadas 256x256 em diversas implementações de famílias *wavelets* (Daubechies, Symlets e Biortogonais).

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	99,2647%	0,3412%	0,4651%	0,1935%	0,0767%
Bridge	96,6097%	0,7214%	1,5454%	1,0568%	0,7881%
Lena	96,8312%	0,7919%	1,8112%	0,8907%	0,4669%
Goldhill	97,9880%	0,8284%	0,9001%	0,5720%	0,5399%

Tabela 5.2: *Wavelets* de Daubechies: energia dos coeficientes para as imagens 256x256 testadas.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	99,2928%	0,1142%	0,4723%	0,1726%	0,0623%
Bridge	96,4054%	0,2088%	1,7358%	1,0944%	0,7644%
Lena	97,0122%	0,2841%	1,7844%	0,8075%	0,3959%
Goldhill	97,9347%	0,3380%	0,9710%	0,5617%	0,5326%

Tabela 5.3: *Wavelets* de Symlets: energia dos coeficientes para as imagens 256x256 testadas.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	98,7260%	0,5571%	0,9301%	0,2739%	0,0700%
Bridge	94,0953%	2,5438%	3,5034%	1,7161%	0,6852%
Lena	94,9231%	1,9757%	3,4051%	1,2877%	0,3841%
Goldhill	96,4617%	1,9107%	2,0443%	1,0042%	0,4898%

Tabela 5.4: *Wavelets* Biortogonais: energia dos coeficientes para as imagens 256x256 testadas.

As Tabelas 5.5-5.7 apresentam informações análogas para um conjunto de imagens 512x512.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Barb	97,2341%	0,3494%	0,8091%	0,7547%	1,2021%
Boat	98,7269%	0,2302%	0,6859%	0,3992%	0,1880%
Goldhill	98,6755%	0,3987%	0,6333%	0,4168%	0,2744%
Mandrill	96,5018%	0,2247%	1,0489%	1,2368%	1,2125%
Lena	98,5614%	0,3624%	0,8416%	0,4251%	0,1719%
Peppers	98,3430%	0,3273%	0,9721%	0,4456%	0,2393%
Washsat	93,4706%	0,4994%	2,2083%	2,4942%	1,8269%
Zelda	99,3209%	0,2180%	0,3893%	0,1913%	0,0985%

Tabela 5.5: *Wavelets* de Daubechies: energia dos coeficientes para as imagens 512x512 testadas.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Barb	97,3349%	0,1440%	0,7566%	0,7178%	1,1907%
Boat	98,7605%	0,1045%	0,6907%	0,3811%	0,1677%
Goldhill	98,6752%	0,1490%	0,6485%	0,4122%	0,2641%
Mandrill	96,3997%	0,1256%	1,1462%	1,2774%	1,1767%
Lena	98,6677%	0,1422%	0,8034%	0,3866%	0,1423%
Peppers	98,4024%	0,0781%	0,9025%	0,4686%	0,2265%
Washsat	93,7334%	0,3600%	2,1518%	2,4236%	1,6912%
Zelda	99,3807%	0,0768%	0,3615%	0,1717%	0,0861%

Tabela 5.6: *Wavelets* de Symlets: energia dos coeficientes para as imagens 512x512 testadas

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Barb	95,2360%	2,6315%	2,0353%	1,7303%	0,9984%
Boat	97,8092%	0,9380%	1,4714%	0,5644%	0,1550%
Goldhill	97,6463%	1,1010%	1,4613%	0,6493%	0,2431%
Mandrill	93,3417%	3,4285%	3,1620%	2,4733%	1,0230%
Lena	97,4274%	1,1343%	1,7994%	0,6208%	0,1524%
Peppers	97,0330%	1,1502%	1,9804%	0,7746%	0,2120%
Washesat	88,6993%	5,2218%	5,7881%	4,0826%	1,4300%
Zelda	98,8533%	0,4779%	0,7897%	0,2692%	0,0878%

Tabela 5.7: Wavelets Biortogonais: energia dos coeficientes para as imagens 512x512 testadas.

Para a análise de cada imagem-teste, a energia dos coeficientes obtida para cada *wavelet* que compõe cada família (Daubechies, Symlets e Biortogonais) não apresentou variação significativa e, deste modo, foi representada pela média tomada por todos os valores dos coeficientes das *wavelets* que as compõem, seguida da medida de desvio padrão (σ) para cada caso. As tabelas referentes a estes valores assumem três níveis de detalhes, sendo que para cada nível foi realizado a soma dos detalhes horizontais, verticais e diagonais da imagem.

Pode perceber-se, a concentração de energia na aproximação e o decaimento conforme aumenta-se a quantidade de níveis de decomposição da imagem. Este estudo pode ser estendido visando medir a eficiência da compactação da energia de cada família, característica interessante para possibilitar ganho de compressão. Através de uma análise estatística pode-se analisar qual a melhor *wavelet* sob o ponto de vista da compactação. Por fugir ao escopo da presente pesquisa, tal avaliação não foi abordada, estando separada para os trabalhos futuros do pesquisador na seqüência desta dissertação de mestrado.

5.3.3 Implementação da Transformada Wavelet Ortogonal Bidimensional: Reconstrução

A imagem decomposta pode ser reconstruída uma vez que $W_{2^j}^2$ é o complemento ortogonal de $V_{2^j}^2$ em $V_{2^{j+1}}^2$. O conjunto de funções $\left(\left\{ \phi_{j;n_x,n_y}(x,y) \right\} \left\{ \psi_{j;n_x,n_y}^i(x,y), 1 \leq i \leq 3 \right\} \right)_{(n_x,n_y) \in Z}$ é a base ortonormal em $V_{2^{j+1}}^2$ e a função $\phi_{j+1;n_x,n_y}(x,y)$ pode ser decomposta da seguinte forma:

$$\begin{aligned} \phi_{j+1;n_x,n_y}(x,y) &= \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \phi_{j;k_x,k_y}(u,v) \rangle \cdot \phi_{j;k_x,k_y}(x,y) + \\ &+ \sum_{i=1}^3 \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \psi_{j;k_x,k_y}^i(u,v) \rangle \cdot \psi_{j;k_x,k_y}^i(x,y) \end{aligned} \quad (5.78)$$

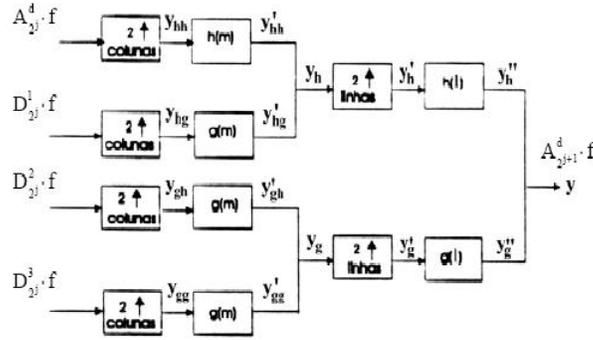


Figura 5.9: Diagrama em blocos de um estágio da reconstrução da imagem [24].

Realizando algumas substituições em (5.78) e convoluindo o resultado com $f(x, y)$, obtém-se:

$$\begin{aligned}
 \langle f(x, y) \cdot \phi_{j+1; n_x, n_y}(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} h(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} h(n_x - 2k_x) \langle f(x, y) \cdot \phi_{j; k_x, k_y}(x, y) \rangle + \\
 &+ \sum_{k_y=-\infty}^{\infty} g(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} h(k_x - 2k_x) \langle f(x, y) \cdot \psi_{j; k_x, k_y}^1(x, y) \rangle + \\
 &+ \sum_{k_y=-\infty}^{\infty} h(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} g(n_x - 2k_x) \langle f(x, y) \cdot \psi_{j; k_x, k_y}^2(x, y) \rangle + \\
 &+ \sum_{k_y=-\infty}^{\infty} g(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} g(n_x - 2k_x) \langle f(x, y) \cdot \psi_{j; k_x, k_y}^3(x, y) \rangle \quad (5.79)
 \end{aligned}$$

Este algoritmo é melhor detalhado pela Fig. 5.9, que o ilustra por diagrama de blocos, em que a imagem é reconstruída a partir da representação *wavelet* repetindo-se este processo para $-1 \leq j \leq -J$. Pode-se concluir por (5.79) que, $A_{2^{j+1}}^d \cdot f$ é reconstruída por filtragens unidimensionais das imagens $A_{2^j}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$ e $D_{2^j}^3 \cdot f$. O primeiro passo, consiste na inserção de uma coluna de zeros entre cada coluna $A_{2^j}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$ e $D_{2^j}^3 \cdot f$, onde é realizado a convolução das linhas resultantes com um filtro unidimensional. O passo seguinte, consiste em realizar a inserção de uma linha de zeros entre cada linha, seguida pela convolução das colunas resultantes com outro filtro unidimensional. Os filtros unidimensionais utilizados referem-se à $h(m)$ e $g(m)$.

5.4 Comentários

Este capítulo apresentou a implementação da TWD (decomposição e síntese) para o caso ortogonal, tanto para o modo unidimensional, quanto para o modo bidimensional. A geração dos sinais provenientes dos filtros passa-baixa e passa-alta são de fundamental importância no processamento de sinais. Conceitos estes que fornecem um ótimo embasamento teórico para o entendimento da análise multiresolução.

Apresentada esta teoria ortogonal e visando sanar deficiências do caso ortogonal, a biortogonalidade torna-se o próximo assunto a ser abordado, a qual consagrou a utilização da TWD em esquemas de compressão de imagens com excelente desempenho.

Capítulo 6

BIORTOGONALIDADE

Apesar da ortogonalidade apresentar muitas vantagens, a mesma não oferece suporte às exigências impostas para que a transformada utilizada apresente, entre outras vantagens, suporte compacto, simetria e regularidade. A única *wavelet* ortogonal que obedece a tais exigências se trata da *wavelet* de Haar, porém a mesma não é contínua, refletindo má localização em frequência não sendo empregada em aplicações práticas. Os filtros empregados deveriam garantir a reconstrução do sinal sem erros, entretanto, Daubechies *et.al* [4] afirma que não existe filtro FIR ortonormal e simétrico trivial com reconstrução perfeita.

As exigências citadas para a transformada em questão são de grande relevância em diversas aplicações, dentre estas, no processamento digital de imagens em que se concentra o trabalho em questão. A regularidade está relacionada diretamente com o número de funções possíveis de serem representados com maior fidelidade. Os filtros simétricos ou filtros de fase linear, por sua vez, tornam possível o cascadeamento de banco de filtros sem a necessidade de compensação de fase, já que as imagens exigem a utilização de tais filtros. Finalmente, *wavelets* de suporte compacto implicam que os filtros apresentem resposta impulsiva de duração finita (FIR) e conseqüentemente, algoritmos de rápida execução.

Para oferecer suporte a essas questões conflitantes, a condição de ortogonalidade é relaxada, gerando a biortogonalidade. Os filtros utilizados para a implementação das *wavelets* biortogonais são bem mais flexíveis e mais fáceis de serem projetados [63]. Vale ressaltar que, as *wavelets* biortogonais são uma generalização das *wavelets* ortogonais [59] [64] [65]. Neste contexto, a análise multiresolução para *wavelets* biortogonais é similar à apresentada para o caso ortogonal, porém há dois espaços vetoriais V_{2^j} e \tilde{V}_{2^j} , um par de funções-escala duais $\phi(x)$ e $\tilde{\phi}(x)$, dois espaços vetoriais de detalhamento W_{2^j} e \tilde{W}_{2^j} , assim como um par de *wavelets* duais $\psi(x)$ e $\tilde{\psi}(x)$ [20], conforme detalhado nas seções seguintes.

6.1 Teoria de *Frames*

Para se obter uma reconstrução numericamente estável de $f(x)$ a partir dos coeficientes $W_{j,n}$ é importante que os vetores utilizados para a expansão sejam constituídos por *frames* [47]. Deste modo, considerando $\{g_k, k \in K\}$ um conjunto de funções em $L^2(R)$, este conjunto será um *frame* se existirem números $0 < A \leq B < \infty$, tais que:

$$A \|f\|^2 \leq \sum_{k \in K} |g_k, f|^2 \leq B \|f\|^2, \forall f \in L^2(R) \quad (6.1)$$

Em (6.1), A e B são denominados de limitantes do *frame* [17]. Um *frame* é um conjunto de funções, em que qualquer função não-nula deve ter uma projeção não-nula em pelo menos uma destas

Um *frame* é necessariamente um conjunto completo em $L^2(R)$. Nesse contexto, as funções de um *frame* podem ser linearmente dependentes (implicando em redundância), onde o *frame* não constitui uma base para $L^2(R)$, ou podem ser linearmente independentes (sem redundância) onde o *frame* constitui uma base para $L^2(R)$ (não necessariamente uma base ortonormal) sendo denominado *frame* exato. Será considerado um *frame* estreito se seus limitantes (A e B) forem iguais, refletindo que a energia após a transformação é a energia de $f(x)$ escalonada.

Lema 1

Se os limitantes (A e B) do *frame* forem iguais, o *frame* é denominado de estreito indicando que a energia após a transformação é a energia de $f(x)$ escalonada. Nesse caso, tem-se que $A = B = 1$, obtendo desta forma o caso de amostragem crítico e o *frame* é uma base ortonormal [45].

Qualquer *frame* $\{g_k, k \in K\}$ pode ser associado a um "operador *frame*" $T : L^2(R) \rightarrow L^2(R)$ que realize a expansão em série de $f(x)$, onde o *frame* $\{g_k\}$ é utilizado como base e os coeficientes são dados por $\langle f, g_k \rangle$, tendo assim:

$$T.f(x) = \sum_{k \in K} \langle f, g_k \rangle \cdot g_k(x) \quad (6.2)$$

Lema 2

Em [61] demonstra-se que o operador T é inversível e, nesse contexto, o *frame* dual de $\{g_k, k \in K\}$ pode ser definido como um conjunto de funções $\{\gamma_k, k \in K\}$ obtidas a partir de $\{g_k\}$.

Define-se o conjunto de funções $\{\gamma_k, k \in K\}$ como sendo o *frame* dual de $\{g_k, k \in K\}$ obtido da seguinte forma [61]:

$$\gamma_k = T^{-1}g_k \quad (6.3)$$

Os *frames* $\{\gamma_k\}$ e $\{g_k\}$ são biortogonais quando o *frame* for exato e assim, tem-se:

$$\langle g_k, \gamma_l \rangle = \delta(k - l) \quad (6.4)$$

Teorema 4

A partir das equações (6.2) e (6.3), tem-se que qualquer função $f(x) \in L^2(R)$ pode ser expressa por:

$$f(x) = \sum_{k \in K} \langle f, \gamma_k \rangle \cdot g_k(x) \quad (6.5)$$

ou

$$f(x) = \sum_{k \in K} \langle f, g_k \rangle \cdot \gamma_k(x) \quad (6.6)$$

Em (6.5) e (6.6) mostram que a função $f(x)$ pode ser expressa como uma combinação linear das funções componentes do *frame*, podendo ser representada tanto pelo *frame* $\{g_k\}$ quanto pelo seu dual $\{\gamma_k\}$. Vale ressaltar que, para se obter $f(x)$ dado um *frame* $\{g_k\}$ é preciso simplesmente calcular o seu dual $\{\gamma_k\}$, por intermédio de (6.3) [46].

Daubechies [45] afirma que se uma *wavelet* $\psi(x)$ gera um *frame*, implica a existência da inversa, em que $\psi(x)$ deve satisfazer a relação [45]:

$$A \|f\|^2 \leq \sum_{j,k \in Z} |\langle f, \psi_{j,n} \rangle|^2 \leq B \|f\|^2 \quad (6.7)$$

Em outras palavras, uma função $f(x) \in L^2(R)$ pode ser recuperada a partir dos coeficientes de sua TW.

Considerando $\{\psi_{j,n}\}$ um *frame* e substituindo $\{g_k\} = \{\psi_{j,n}\}$ em (6.2), obtém-se:

$$T \cdot f(x) = \sum_{j,n \in Z} |\langle f, \psi_{j,n} \rangle| \cdot \psi_{j,n}(x) \quad (6.8)$$

De acordo com o Teorema 4, a função $f(x)$ pode ser reescrita da forma:

$$f(x) = T^{-1} \cdot T \cdot f(x) = \sum_{j,n \in Z} \langle f, \psi_{j,n} \rangle \cdot T^{-1} \cdot \psi_{j,n}(x) \quad (6.9)$$

O *frame* dual de $\psi_{j,n}$, representado por $\tilde{\psi}_{j,n}$, é definido por:

$$\tilde{\psi}_{j,n}(x) = T^{-1} \cdot \psi_{j,n}(x) \quad (6.10)$$

Isto conduz à fórmula de reconstrução da função $f(x)$ que pode ser reescrita como:

$$f(x) = \sum_{j,n \in \mathbb{Z}} \langle f, \psi_{j,n} \rangle \cdot \tilde{\psi}_{j,n}(x) \triangleq \sum_{j,n \in \mathbb{Z}} \langle f, \tilde{\psi}_{j,n} \rangle \cdot \psi_{j,n}(x) \quad (6.11)$$

Para o caso de $\{\psi_{j,n}\}$ referir a *frames* estreitos, implica em dizer que o *frame* dual é composto de versões escalonadas e deslocadas da *wavelet*-básica, multiplicadas por uma constante, ou seja:

$$\tilde{\psi}_{j,n} = A^{-1} \psi_{j,n} \quad (6.12)$$

Podendo reescrever (6.9) como:

$$f(x) = A^{-1} \sum_{j,n \in \mathbb{Z}} \langle f, \psi_{j,n} \rangle \cdot \psi_{j,n}(x) \quad (6.13)$$

Para os *frames* que condizerem com o Lema 1, estes formam bases ortonormais e (6.13) pode ser reduzida em:

$$f(x) = \sum_{j,n \in \mathbb{Z}} \langle f, \psi_{j,n} \rangle \cdot \psi_{j,n}(x) \quad (6.14)$$

6.2 Análise de Multiresolução para *Wavelets* Biortogonais

A multiresolução birtogonal consiste basicamente em dois espaços vetoriais de aproximação $\{V_{2^j}\}_{j \in \mathbb{Z}}$ e $\{\tilde{V}_{2^j}\}_{j \in \mathbb{Z}}$, um par de funções-escala duais $\phi(x)$ e $\tilde{\phi}(x)$, assim como dois espaços de detalhamentos $\{W_{2^j}\}_{j \in \mathbb{Z}}$ e $\{\tilde{W}_{2^j}\}_{j \in \mathbb{Z}}$ e finalmente, um par de *wavelets* duais ψ e $\tilde{\psi}$. Pode-se perceber, desta forma, um aumento da complexidade quando comparado ao caso ortogonal.

Porém, muitos conceitos são semelhantes ao caso ortogonal como, por exemplo, a seguinte hierarquia para os subespaços V_{2^j} e \tilde{V}_{2^j} :

$$\begin{aligned} \dots \subset V_{2^{-2}} \subset V_{2^{-1}} \subset V_{2^0} \subset V_{2^1} \subset V_{2^2} \subset \dots \\ \dots \subset \tilde{V}_{2^{-2}} \subset \tilde{V}_{2^{-1}} \subset \tilde{V}_{2^0} \subset \tilde{V}_{2^1} \subset \tilde{V}_{2^2} \subset \dots \end{aligned} \quad (6.15)$$

Os subespaços V_{2^j} e \widetilde{V}_{2^j} são gerados pelas bases não-ortogonais $\{\phi_{j,n}\}$ e $\{\widetilde{\phi}_{j,n}\}$, em que tais bases são obtidas a partir de deslocamentos, dilatações e compressões das funções-escala $\phi(x)$ e $\widetilde{\phi}(x)$, respectivamente. As funções-escala são regidas por:

$$\langle \phi(t), \widetilde{\phi}(t-m) \rangle = \delta(m) \quad (6.16)$$

Os espaços de detalhamentos W_{2^j} e \widetilde{W}_{2^j} são complementos não ortogonais de V_{2^j} em $V_{2^{j+1}}$ e de \widetilde{V}_{2^j} em $\widetilde{V}_{2^{j+1}}$, respectivamente, tendo assim:

$$V_{2^{j+1}} = V_{2^j} \oplus W_{2^j}$$

$$\widetilde{V}_{2^{j+1}} = \widetilde{V}_{2^j} \oplus \widetilde{W}_{2^j} \quad (6.17)$$

Vale ressaltar que V_{2^j} não é ortogonal a W_{2^j} em que \widetilde{V}_{2^j} também não é ortogonal a \widetilde{W}_{2^j} . Portanto, os subespaços vetoriais de detalhamentos obedecem a seguinte hierarquia:

$$\dots \subset W_{2^{-2}} \subset W_{2^{-1}} \subset W \subset W_{2^1} \subset W_{2^2} \subset \dots$$

$$\dots \subset \widetilde{W}_{2^{-2}} \subset \widetilde{W}_{2^{-1}} \subset \widetilde{W}_{2^0} \subset \widetilde{W}_{2^1} \subset \widetilde{W}_{2^2} \subset \dots \quad (6.18)$$

Utilizando um raciocínio análogo, os subespaços W_{2^j} e \widetilde{W}_{2^j} podem ser obtidos a partir das bases $\{\psi_{j,n}\}$ e $\{\widetilde{\psi}_{j,n}\}$, em que são bases não ortogonais obtidas por deslocamentos, dilatações e compressões das *wavelets*-básicas $\psi(x)$ e $\widetilde{\psi}(x)$, respectivamente. Essas *wavelets*, por sua vez, obedecem à seguinte relação:

$$\langle \psi(t), \widetilde{\psi}(t-m) \rangle = \delta(m) \quad (6.19)$$

Em análise de multiresolução biortogonal, além das propriedades já vistas, tem-se que [45]:

$$V_{2^j} \perp \widetilde{W}_{2^j} \text{ e}$$

$$\widetilde{V}_{2^j} \perp W_{2^j} \quad (6.20)$$

A equação implica na seguinte relação:

$$\langle \phi(t), \widetilde{\psi}(t-m) \rangle = \langle \widetilde{\phi}(t), \psi(t-m) \rangle = \delta(m) \quad (6.21)$$

Sejam $\{\phi_{j^n}\}$ e $\{\tilde{\phi}_{j^n}\}$ os dois *frames* duais geradores dos subespaços V_{2^j} e \tilde{V}_{2^j} . A projeção da função $f(x) \in L^2(R)$ na base $\{\phi_{j,n}\}$ no nível de resolução 2^j , segundo a teoria de *frames*, para o caso biortogonal pode ser expressa por:

$$A_{2^j} \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f, \tilde{\phi}_{j,n} \rangle \phi_{j,n}(x) \quad (6.22)$$

A equação (6.22), em termos de sua aproximação discreta, pode ser obtida por:

$$A_{2^j}^d \cdot f = \left\{ \langle f, \tilde{\phi}_{j,n} \rangle, n \in Z \right\} \quad (6.23)$$

De forma análoga, a projeção da função $f(x) \in L^2(R)$ na base $\{\psi_{j,n}\}$ no nível de resolução 2^j , sendo $\{\psi_{j,n}\}$ e $\{\tilde{\psi}_{j,n}\}$ os dois *frames* duais geradores dos subespaços W_{2^j} e \tilde{W}_{2^j} , pode ser obtida, segundo a teoria de *frames*, para o caso biortogonal por:

$$D_{2^j}^d \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f, \tilde{\psi}_{j,n} \rangle \cdot \psi_{j,n}(x) \quad (6.24)$$

este sinal de detalhamento em sua forma discreta, pode ser representado por:

$$D_{2^j}^d \cdot f = \left\{ \langle f, \tilde{\psi}_{j,n} \rangle, n \in Z \right\} \quad (6.25)$$

Tendo em mente que $\{\tilde{\phi}_{j,n}\} \in \tilde{V}_{2^j} \subset \tilde{V}_{2^{j+1}}$, esta pode ser decomposta na base de $\tilde{V}_{2^{j+1}}$, obtendo:

$$\begin{aligned} \tilde{\phi}_{j,n}(x) &= \sum_{k=-\infty}^{\infty} \langle \tilde{\phi}_{j,n}, \phi_{j+1,k} \rangle \cdot \tilde{\phi}_{j+1,k}(x) \\ &= \sum_{k=-\infty}^{\infty} \langle \tilde{\phi}_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \tilde{\phi}_{j+1,k}(x) \\ &= \sum_{k=-\infty}^{\infty} \tilde{h}(k-2n) \cdot \tilde{\phi}_{j+1,k}(x) \end{aligned} \quad (6.26)$$

A equação (6.26) pode ser encarada como uma filtragem e, desta forma, pode-se definir o filtro discreto de resposta impulsiva $\tilde{h}(m)$ como:

$$\tilde{h}(m) = \langle \tilde{\phi}_{-1,0}(u), \phi_{0,m}(u) \rangle \quad (6.27)$$

De forma análoga, como $\{\tilde{\psi}_{j,n}\} \in \tilde{W}_{2^j} \subset \tilde{W}_{2^{j+1}}$, esta pode ser decomposta na base de $\tilde{W}_{2^{j+1}}$:

$$\begin{aligned}
\tilde{\psi}(x) &= \sum_{k=-\infty}^{\infty} \langle \tilde{\psi}_{j,n}, \phi_{j+1,k} \rangle \cdot \tilde{\phi}_{j+1,k}(x) \\
&= \sum_{k=-\infty}^{\infty} \langle \tilde{\psi}_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \tilde{\phi}_{j+1,k}(x) \\
&= \sum_{k=-\infty}^{\infty} \tilde{g}(k-2n) \cdot \tilde{\phi}_{j+1,k}(x)
\end{aligned} \tag{6.28}$$

A equação (6.28) pode ser encarada como uma filtragem e, desta forma, pode-se definir o filtro discreto de resposta impulsiva $\tilde{g}(m)$ como:

$$\tilde{g}(m) = \langle \tilde{\psi}_{-1,0}(u), \phi_{0,m}(u) \rangle \tag{6.29}$$

Convolvindo (6.26) e (6.28) com a função $f(x)$, o processo de decomposição da análise multiresolução biortogonal é caracterizado por:

$$\langle f, \tilde{\phi}_{j,n} \rangle = \sum_{k=-\infty}^{\infty} \tilde{h}(k-2n) \cdot \langle f, \tilde{\phi}_{j+1,k} \rangle \tag{6.30}$$

$$\langle f, \tilde{\psi}_{j,n} \rangle = \sum_{k=-\infty}^{\infty} \tilde{g}(k-2n) \cdot \langle f, \tilde{\phi}_{j+1,k} \rangle \tag{6.31}$$

6.3 Reconstrução a partir da Representação Wavelet Biortogonal

A aproximação da função $f(x)$ no nível de resolução 2^{j+1} , em que se tem um nível acima do que se obtém em (6.22), pode ser representada por:

$$A_{2^{j+1}}^d f(x) = \sum_{n=-\infty}^{\infty} \langle f(x), \tilde{\phi}_{j+1,n} \rangle \phi_{j+1,n} \tag{6.32}$$

O espaço $V_{2^{j+1}}$ pode ser gerado pela base $(\{\phi_{j,n}\}, \{\psi_{j,n}\})_{n \in \mathbb{Z}}$, sabendo que o subespaço W_{2^j} é o complemento não-ortogonal de V_{2^j} em $V_{2^{j+1}}$.

Os dois filtros digitais de resposta impulsiva utilizados no processo são representados por:

$$h(m) = \langle \phi_{-1,0}(u), \tilde{\phi}_{0,m}(u) \rangle \tag{6.33}$$

$$g(m) = \langle \psi_{-1,0}(u), \tilde{\phi}_{0,m}(u) \rangle \tag{6.34}$$

Deste modo, $\tilde{\phi}_{j+1,n}$ é representada em (6.35) e pode ser interpretada como a soma de duas filtrações:

$$\tilde{\phi}_{j+1,n}(x) = \sum_{k=-\infty}^{\infty} h(n-2k) \cdot \tilde{\phi}_{j,k}(x) + \sum_{k=-\infty}^{\infty} g(n-2k) \cdot \tilde{\psi}_{j,k}(x) \quad (6.35)$$

A equação (6.35) convoluída com $f(x)$, conduz à:

$$\begin{aligned} \langle f(x), \tilde{\phi}_{j+1,n}(x) \rangle &= \sum_{k=-\infty}^{\infty} h(n-2k) \cdot \langle f(x), \tilde{\phi}_{j,k}(x) \rangle \\ &+ \sum_{k=-\infty}^{\infty} g(n-2k) \cdot \langle f(x), \tilde{\psi}_{j,k}(x) \rangle \end{aligned} \quad (6.36)$$

A reconstrução do sinal, ilustrada pela Fig. 6.1, a partir da representação *wavelet* biortogonal é semelhante à síntese multiresolução ortogonal, resultando em:

$$A_{2^{j+1}}^d \cdot f(x) = A_{2^j}^d \cdot f(x) + D_{2^j}^d \cdot f(x) \quad (6.37)$$

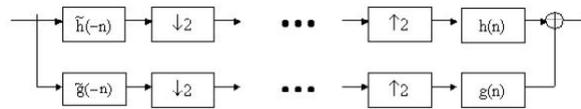


Figura 6.1: Diagrama em blocos do algoritmo de decomposição e síntese da TWD biortogonal [24].

Nota-se que cada coeficiente de $A_{2^{j+1}}^d \cdot f$ é formado por todos os coeficientes de $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$, em que $A_{2^{j+1}}^d \cdot f$ pode ser reconstruída pela inserção de um zero entre cada amostra de $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$ convoluindo os resultados com os filtros $h(n)$ e $g(n)$, respectivamente.

Conforme visto na Fig. 6.1, diferentemente do caso ortogonal na decomposição, os filtros são simbolizados por $\tilde{h}(-n)$ e $\tilde{g}(-n)$, enquanto que na síntese utiliza-se filtros de resposta impulsiva $h(n)$ e $g(n)$. Além disso, os filtros correspondentes à *wavelet* biortogonal são mais flexíveis, fáceis de projetar e podem ser simétricos [4].

6.4 Comentários

Com a biortogonalidade torna-se possível preservar a simetria e a reconstrução exata e ainda obter alta regularidade, sendo a alternativa encontrada para superar os limites impostos pela condição de or-

togonalidade. Para melhor compreensão da biortogonalidade, a teoria de frames foi enfatizada assim como a análise multiresolução para decomposição e síntese, com o foco voltado para as principais formulações matemáticas que a caracterizam.

Capítulo 7

REGULARIDADE E SELEÇÃO DAS *WAVELETS*

A regularidade da TW está diretamente relacionada com uma boa representação da função, pois componentes de alta frequência gerados resultam em erros e perdas de qualidade visual proveniente da compressão. No caso particular de sinais referente a imagens em que se tem mudança abrupta de contraste ou de outras descontinuidades, a análise e síntese de Fourier não são muito apropriadas para acomodar os termos de alta frequência.

Dentro desse contexto, uma abordagem sobre as características desejáveis para a TW na compressão de imagens, com destaque para a regularidade, é enfocada neste capítulo, além de enfatizar a importância da biortogonalidade para esta questão. O enfoque será dado às *wavelets* que foram empregadas nas implementações, com o intuito de possibilitar uma melhor compreensão das mesmas.

7.1 Seleção das *Wavelets* e *Wavelets* Utilizadas

A TW e, conseqüentemente, seus filtros de implementação, são regidos por determinadas características, as quais possibilitam, deste modo, que cada *wavelet* apresente um comportamento específico [3] [17] [48] [45].

A regularidade das *wavelets* está relacionada com o tipo de função que a transformada pode apresentar, em que, quanto maior for a regularidade, maior o número de sinais possíveis de serem representados com maior fidelidade. Em outras palavras, quanto maior o comprimento do filtro, mais regular torna-se a *wavelet*, em contrapartida, pior o aspecto do ponto de vista da velocidade em realizar a implementação. Percebe-se assim que, suporte compacto é conflitante com a regularidade.

Outro fator de relevância refere-se à simetria das *wavelets*, já que as imagens carecem da utilização de filtros de fase linear (filtros simétricos), o que possibilita o cascadeamento de bancos, além de

evitar possíveis problemas nas bordas naturais das imagens, devido ao fato de não se preocupar com a necessidade de compensação de fase. Visando uma menor introdução de erros, adota-se filtros com coeficientes racionais diádicos, evitando assim as operações de ponto flutuante.

O suporte compacto desejado, por sua vez, refere-se a uma *wavelet*-mãe que deveria apresentar uma função oscilatória de curta duração, em que as translações diádicas de escalonamentos binários da função fossem ortonormais. Se o suporte compacto estiver presente tanto na função escala quanto na *wavelet*, algoritmos de rápida implementação podem ser almejados, já que as respostas impulsivas dos filtros correspondentes possuem comprimento finito.

Diante disso, encontrar um ponto de equilíbrio que satisfizesse esta condição seria uma boa solução para o caso, porém em [4], Daubechies afirma que independente da regularidade, não existe filtro de comprimento finito ortonormal e simétrico trivial com propriedade de reconstrução exata à exceção de Haar. A biortogonalidade, por sua vez, permite preservar a simetria e a reconstrução exata, além de obter alta regularidade e bom desempenho na compressão de imagens [4], podendo ser vista sob o aspecto de suprir as necessidades impostas pela ortogonalidade.

Os filtros utilizados no esquema de decomposição e reconstrução da TW devem garantir a reconstrução perfeita do sinal, ou seja, nenhuma informação deve ser perdida. Nesse contexto, um grande número de funções podem ser eleitas como *wavelets*-mãe, sendo estas escolhidas de acordo com o sinal em que se deseja analisar. As *wavelets* de Daubechies, Symlets e as Biortogonais merecem maior destaque por apresentarem fundamental importância nas implementações realizadas.

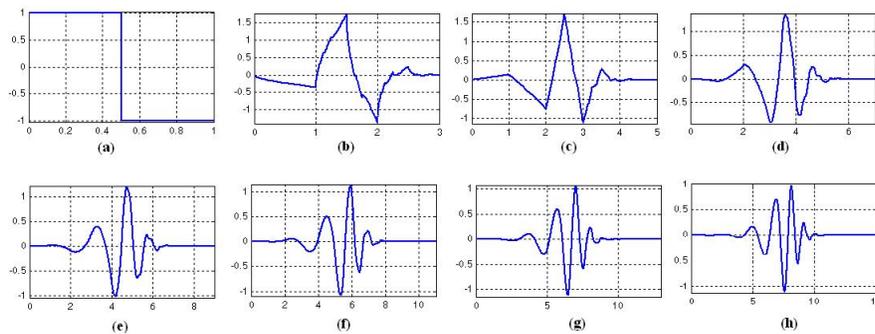
7.1.1 Wavelets de Daubechies

Ingrid Daubechies realizou a construção de um conjunto de *wavelets*-mãe com características especiais, dentre estas a de apresentar suporte compacto. As *wavelets* que compõem esta família, são simbolizadas por DbN , onde N é o parâmetro que controla a regularidade de $\phi(x)$ e $\psi(x)$. Vale destacar que, para $N = 1$ obtém-se a *wavelet* de Haar e, que as *wavelets* tornam-se mais regulares conforme aumenta-se o valor de N .

O algoritmo de Daubechies para a construção de filtros FIR que geram *wavelets* regulares, ortogonais e de suporte compacto pode ser encontrado em [63]. Sua abordagem não foi aqui descrita por fugir do escopo da dissertação.

Essa família apresenta muitos tipos de *wavelets* podendo apresentar ordens bem elevadas. Porém, essa família foi analisada apenas para o intervalo $1 \leq N \leq 8$, onde a Tabela 7.1 ilustra o comprimento dos filtros e seus respectivos coeficientes referentes a estas *wavelets*-mãe $\psi(x) |_{N}$, de acordo com o parâmetro de regularidade estabelecido por cada *wavelet*. O comportamento das *wavelets*-mãe $\psi(x) |_{1 \leq N \leq 8}$ é mostrado na Fig. 7.1.

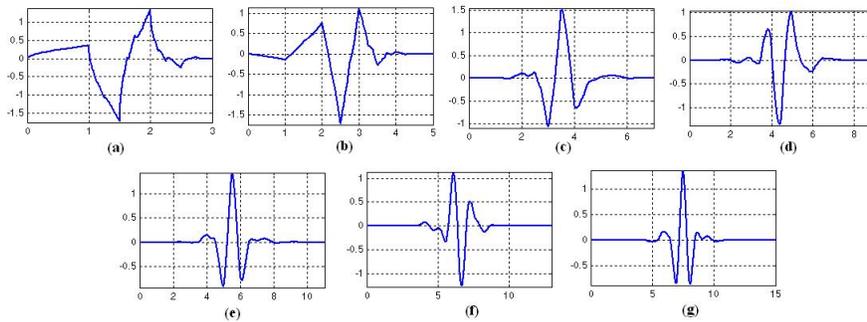
N	Coeficientes								
1	$h_0 - h_1$	0,7071	0,7071						
2	$h_0 - h_3$	0,4830	0,8365	0,2241	-0,1294				
3	$h_0 - h_5$	0,3327	0,8069	0,4599	-0,1350	-0,0854	0,0352		
4	$h_0 - h_7$	0,2304	0,7148	0,6309	-0,0280	-0,1870	0,0308	0,0329	-0,0106
5	$h_0 - h_4$	0,1601	0,6038	0,7243	0,1384	-0,2423			
	$h_5 - h_9$	-0,0322	0,0776	-0,0062	-0,0126	0,0033			
6	$h_0 - h_5$	0,1115	0,4946	0,7511	0,3153	-0,2263	-0,1298		
	$h_0 - h_{11}$	0,0975	0,0275	-0,0316	0,0006	0,0048	-0,0011		
7	$h_0 - h_6$	0,779	0,3965	0,7291	0,4698	-0,1439	0,2240	0,0713	
	$h_7 - h_{13}$	0,0806	-0,0380	-0,0166	0,126	0,0004	-0,0018	0,0004	
8	$h_0 - h_7$	0,0544	0,3129	0,6756	0,5854	-0,0158	-0,2840	0,0005	0,1287
	$h_8 - h_{15}$	-0,0174	-0,0441	0,0140	0,0087	-0,0049	-0,0004	0,0007	-0,0001

Tabela 7.1: Coeficientes dos filtros das *wavelets* de Daubechies.Figura 7.1: *Wavelets* de Daubechies para: (a)N=1; (b)N=2; (c)N=3; (d)N=4; (e)N=5; (f)N=6; (g)N=7; (h)N=8.

7.1.2 Wavelet de Symlets

As *wavelets* de Symlets, também construídas por Daubechies, são mais simétricas e foram projetadas para garantir momentos nulos tanto na função-escala $\phi(x)$ quanto na *wavelet*-mãe $\psi(x)$. Também apresentam suporte compacto e podem ser caracterizadas de acordo com o número de momentos nulos N , sendo representadas por $symN$. Para cada valor de N , as *wavelets* apresentam um determinado comprimento com seus respectivos coeficientes dos filtros, conforme visto para $2 \leq N \leq 8$ na Tabela 7.2. As *wavelets*-mãe $\psi(x) |_{2 \leq N \leq 8}$ encontram-se na Fig. 7.2.

N	Coeficientes								
2	$h_0 - h_3$	0,4830	0,8365	0,2241	-0,1294				
3	$h_0 - h_5$	0,3327	0,8069	0,4599	-0,1350	-0,0854	0,0352		
4	$h_0 - h_7$	0,0322	-0,0126	-0,0992	0,2979	0,8037	0,4976	-0,0296	-0,0758
5	$h_0 - h_4$	0,0195	-0,0211	-0,1753	0,0166	0,6340			
	$h_5 - h_9$	0,7234	0,1994	-0,0391	0,0295	0,0273			
6	$h_0 - h_5$	-0,0078	0,0018	0,0447	-0,0211	-0,0726	0,3379		
	$h_6 - h_{11}$	0,7876	0,4911	-0,0483	-0,1180	0,0035	0,0154		
7	$h_0 - h_6$	0,0103	0,0040	-0,1078	-0,1400	0,2886	0,7678	0,5361	
	$h_7 - h_{13}$	0,0174	-0,0496	0,0679	0,0305	-0,0126	-0,0010	0,0027	
8	$h_0 - h_7$	0,0019	-0,0003	-0,0150	0,0038	0,0491	-0,0272	-0,0519	0,3644
	$h_7 - h_{15}$	0,7772	0,4814	-0,0613	-0,1433	0,0076	0,0317	-0,0005	-0,0034

Tabela 7.2: Coeficientes dos filtros das *wavelets* de Symlets.Figura 7.2: *Wavelets* de Symlets para: (a)N=2; (b)N=3; (c)N=4; (d)N=5; (e)N=6; (f)N=7; (g)N=8.

7.1.3 *Wavelets* Biortogonais

No que concerne às *wavelets* biortogonais, conforme enfocado no capítulo 6, têm-se na análise multiresolução dois espaços vetoriais V_{2j} e \tilde{V}_{2j} , um par de funções-escala duais $\phi(x)$ e $\tilde{\phi}(x)$, dois espaços vetoriais de detalhamento W_{2j} e \tilde{W}_{2j} , assim como um par de *wavelets* duais $\psi(x)$ e $\tilde{\psi}(x)$. Tanto para ψ quanto para $\tilde{\psi}$, uma alta regularidade pode ser obtida, contanto que os filtros apresentem comprimentos suficientemente longos [4]. Para o caso das funções ψ e $\tilde{\psi}$ serem diferenciáveis por $(r-1)$ e $(\tilde{r}-1)$ vezes no tempo, respectivamente, tem-se assim que $H(w)$ e $\tilde{H}(w)$ são divisíveis, respectivamente, por $(1 + e^{-jw})^r$ e $(1 + e^{-jw})^{\tilde{r}}$, em que, desta forma, os filtros $h(n)$ e $\tilde{h}(n)$ irão apresentar um comprimento maior que r e \tilde{r} , respectivamente [66].

Daubechies em [45] afirma que a divisibilidade de $\tilde{H}(w)$ por $(1 + e^{-jw})^{\tilde{r}}$ resulta que ψ pode ter \tilde{r} momentos nulos consecutivos, em que os coeficientes $\langle f, \psi_{m,n} \rangle$ significariam que as funções f são \tilde{r} vezes diferenciáveis com um alto potencial de compressão. Dentro desse contexto e dentro dos

limites impostos pelo suporte compacto, procure-se escolher \tilde{r} o maior possível para obter bom nível de regularidade [4]. Quando se diz que ψ pode ter \tilde{r} momentos nulos consecutivos, significa que:

$$\int x^l \psi(x) dx = 0, \text{ para } l = 0, 1, \dots, \tilde{r} - 1 \quad (7.1)$$

De [4], a expressão de reconstrução perfeita pode ser definida por:

$$H(w) \tilde{H}(w) + H(w + \pi) \tilde{H}(w + \pi) = 2 \quad (7.2)$$

A fim de poder construir diferentes exemplos de filtros, em (7.3) mostra-se a equação geral para este fim. Diante desta equação, deve-se escolher um polinômio $R(w)$ ímpar em $\cos(w)$ e, em seguida, fatorar $H(w)$ e $\tilde{H}(w)$ gerando-se uma família. Finalmente, escolher r e \tilde{r} obtendo assim filtros diferentes, já que $2l = r + \tilde{r}$. Dada a extensão e complexidade para se chegar a este resultado e por fugir do escopo da dissertação, a mesma pode ser definida por [4]:

$$H(w) \tilde{H}(w) = \cos^{2l} \left(\frac{w}{2} \right) \cdot \left[\sum_{q=0}^{l-1} \left(\binom{l-1+q}{q} \sin^{2q} \left(\frac{w}{2} \right) + \sin^{2l} \left(\frac{w}{2} \right) R(w) \right) \right] \quad (7.3)$$

Algumas *wavelets* biortogonais e suas duais utilizadas nas implementações são ilustradas na Fig. 7.3.

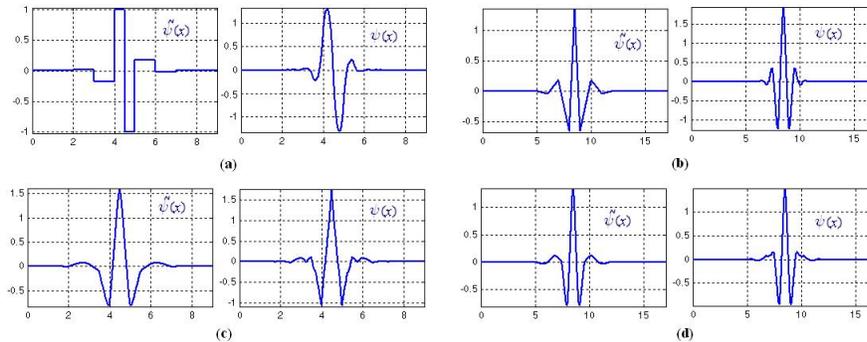


Figura 7.3: *Wavelets* Biortogonais $\psi(x)$ e suas duais $\tilde{\psi}(x)$: (a) bior1.5; (b) bior2.8; (c) bior4.4; (d) bior6.8.

7.1.3.1 *Wavelets Spline* Biortogonais Cohen - Daubechies - Feauveau

Se for atribuído em (7.3) que o parâmetro $R(w) = 0$ e, se apresentar a resposta do filtro passa-baixas dual definido por (7.4), obtém-se desta forma a família de filtros conhecida como filtros *spline*

n	0	± 1	± 2	± 3	± 4
$2^{-l/2} \cdot h_n$	0,7031	0,2969	-0,125	-0,0469	0,0234
$2^{-l/2} \cdot \tilde{h}_n$	0,5	0,25	0	0	0

Tabela 7.3: Coeficientes dos filtros *spline* $l = 3$ e $\tilde{q} = 2$.

n	0	± 1	± 2	± 3	± 4
$2^{-l/2} \cdot h_n$	0,6029	0,2668	-0,0782	-0,0168	0,0267
$2^{-l/2} \cdot \tilde{h}_n$	0,5575	0,2956	-0,0287	-0,0456	0

Tabela 7.4: Coeficientes dos filtros variante *spline* com comprimentos semelhantes $l = q = 4$.

[4], já que a função $\tilde{\phi}$ correspondente é B-*spline* ou filtro binomial, onde os \tilde{h} são coeficientes binomiais.

$$H(w) = \cos^{\tilde{r}}\left(\frac{w}{2}\right) \cdot e^{-j\varepsilon w}, \text{ onde } \begin{cases} \varepsilon = 0, \text{ se } \tilde{r} \text{ for par} \\ \varepsilon = 1, \text{ se } \tilde{r} \text{ for impar} \end{cases} \quad (7.4)$$

Os filtros *spline* são simétricos, suaves e possuem coeficientes diádicos, onde a resposta em frequência do filtro passa-baixas é caracterizada por:

$$H(w) = \cos\left(\frac{w}{2}\right)^{2l-\tilde{r}} \left(\frac{w}{2}\right) e^{j\varepsilon w/2} \left[\sum_{q=0}^{l-1} \binom{l-1+q}{q} \sin^{2q}\left(\frac{w}{2}\right) \right] \quad (7.5)$$

Os coeficientes h_n e \tilde{h}_n dos filtros passa-baixas são apresentados na Tabela 7.3 para o caso em questão, compreendido pelo valores $l = 3$ e $\tilde{r} = 2$. Como característica dos filtros *spline*, pode ser observado a presença do comprimento ímpar e a simetria em relação a 0.

7.1.3.2 Variante de Wavelets Spline Biortogonais

Essa família de filtros é obtida tomando $R(w) = 0$ em (7.3) e fatorando o lado direito da expressão por intermédio de quebrar o polinômio de grau $l - 1$ em $\sin(w/2)$, que em outras palavras, resulta em um produto de dois polinômios em $\sin(w/2)$. Com o intuito de facilitar a implementação, o comprimento do filtro h deve ser mais próximo do comprimento do filtro \tilde{h} , onde o polinômio resultante que apresentar coeficientes reais é definido como H e o outro como \tilde{H} [66].

Os coeficientes h_n e \tilde{h}_n dos filtros passa-baixas são encontrados na Tabela 7.4. Trata-se de um filtro curto, que especificamente falando, o mais curto dessa família. Esta *wavelet* é largamente utilizada em compressão de imagens nos dias de hoje, por apresentar suporte compacto, boa regularidade, simetria, rapidez dos algoritmos, além de possibilitar reconstrução exata.

n	0	± 1	± 2	± 3	± 4
$2^{-l/2} \cdot h_n$	0,6	0,25	-0,05	0	0
$2^{-l/2} \cdot \tilde{h}_n$	0,6071	0,2607	-0,0535	-0,0107	0

Tabela 7.5: Coeficientes dos filtros variante *spline* com comprimentos muito semelhantes $l = q = 2$.

7.1.3.3 Wavelets Próximas às Ortonormais

Existe uma escolha de $R(w)$ para a qual os filtros de *wavelets* biortogonais em (7.3) estão muito próximos e conseqüentemente, muito próximos ao caso ortonormal. Tal situação pode ser ilustrada tomando $R(w) \neq 0$ e, como exemplo, pode ser citado o filtro piramidal laplaciano proposto em [60], tendo $l = q = 2$ e:

$$R(w) = 48\cos\left(\frac{w}{2}\right)/175 \quad (7.6)$$

A Tabela 7.5 ilustra os coeficientes para esse filtro em questão, em que os dois filtros biortogonais são muito próximos de um filtro *wavelet* ortonormal de comprimento 6, mais detalhado em [67], onde foram denominados de *coiflets*. Vale destacar que, para o caso de ser um filtro ortogonal, o *coiflet* não é simétrico.

7.2 Principais Diferenças entre Wavelets Ortogonais e Biortogonais

Os filtros ortogonais possuem necessariamente o mesmo comprimento, sendo este de número par que, em contrapartida, não é imposto para os filtros biortogonais.

Uma das principais vantagens da biortogonalidade refere-se ao fato de possibilitar que *wavelets* e funções escalas simétricas e de reconstrução exata sejam construídas. Porém, como desvantagem, pode ser citado que a norma do vetor dos coeficientes não é idêntica à norma da função, o que se resume em dizer que a identidade de Parseval não é válida para os sistemas biortogonais.

7.3 Tipos de Wavelets

- *Haar*: considerada como sendo uma D1 (Daubechies 1): a primeira *wavelet* com um único momento nulo;
- *Daubechies*: de suporte compacto e suavidade regulável [45] [46] [63] [68];
- *Coiflets*: projetada para satisfazer certos números de momentos nulos;

- *Coifman*: cujos filtros foram projetados para que tanto a *wavelet* quanto a função escaladora tenham momentos nulos [45] [69] [70];
- *Beylkin*: filtros que apresentam raízes na proximidade da frequência de Nyquist [70];
- *Vaidyanathan*: apresenta filtro que exhibe reconstrução exata, apesar de não satisfazer nenhuma condição de momentos [70];
- *Meyer*: derivam das chamadas *wavelets* de Shannon, que são suavemente enjaneladas na frequência tal que o decaimento no tempo (t) possa ser mais rápido que qualquer potência de t [68] [69].
- *Malvar*: *wavelets* ortonormais cuja descoberta se insere na estrutura geral de referência da análise enjanelada de Fourier, vindo a constituir um algoritmo de análise em tempo-frequência, em oposição à análise tempo-escala [68].
- *Biortogonais*: para as quais a condição de ortogonalidade é relaxada [68] [69];
- *Symlets ou simétricas*: apresenta suporte compacto e número de momentos nulos variando de 4 a 10 [70];
- *Ótimas*: fornecem a decomposição mais comprimida de um sinal [69] [70];
- *deOliveira*: nova família de *wavelets* ortogonais complexas, a qual é baseada no critério de Nyquist para eliminação de Interferência Intersimbólica em Sistemas de Comunicação Digital [71].

7.4 Comentários

As características desejáveis para a TWD foram apresentadas, dentre elas, regularidade, suporte compacto e simetria. Nesse contexto, alguns tipos de *wavelets* foram abordadas visando analisar o comportamento que cada uma apresenta, tais como suas características e os respectivos valores dos coeficientes dos filtros. Diante de tal análise, foi enfatizado que, com a biortogonalidade pode-se preservar a simetria e a reconstrução exata, além de obter alta regularidade contornando as questões conflitantes encontradas pela ortogonalidade.

Capítulo 8

LIFTING

A transformada *wavelet* é vista como uma poderosa ferramenta para a compressão de dados e a eficiência de sua compactação depende dos filtros *wavelets* [72]. Neste sentido, diferentes métodos podem ser escolhidos para se calcular a TWD de um sinal, dentre estes encontram-se o método tradicional embasado na teoria de bancos de filtros, conforme já apresentado nos capítulos 3, 4 e 5, e o esquema *lifting*.

Os dois métodos referem-se a diferentes alternativas para realizar a decomposição e reconstrução do sinal original, porém a principal diferença entre estes refere-se ao fato de que o esquema *lifting* não necessita da TF (*Fourier Transform* - Transformada de Fourier) para as operações e conseqüentemente, pode ser denominado como transformada *wavelet* de segunda geração [1] [73]. Ambos têm por objetivo obter o mesmo sinal reconstruído, com a menor diferença entre si e implicando conseqüentemente, na menor distorção entre os sinais originais e reconstruídos [73].

Wavelets de primeira geração são todas transladadas e dilatadas de uma ou algumas formas básicas e, deste modo, a TF é uma ferramenta crucial para construção de *wavelet* [1]. O esquema *lifting*, incorporado pelo padrão JPEG2000 [74], foi introduzido por Sweldens como uma ferramenta flexível e aceleradora, não necessariamente transladadas e dilatadas da *wavelet*-básica [75]-[78].

Deste modo, este capítulo destina-se a realizar uma abordagem sobre os conceitos básicos referentes as principais características em que se baseia este método. Os processos de decomposição e reconstrução são abordados de maneira simples e precisa, onde a decomposição via passos *lifting* para a *wavelet* de Haar é analisada. Conceitos sobre a transformada *wavelet* e representação polinomial auxiliam no embasamento desta teoria, além de apresentar conceitos sobre o algoritmo euclidiano e a fatoração em passos *lifting*. O capítulo é finalizado apresentando, brevemente, como o ganho aritmético teórico é obtido sob a interpretação de Daubechies e Sweldens e também via deduções matemáticas propostas por Reichel.

8.1 O Método *Lifting*

Este método é visto como uma boa solução para reduzir a complexidade da TWD, fazendo uso de menor número de operações aritméticas para que a transformada seja realizada [1] [2]. Neste sentido, provê um rápido processamento por fazer ótimo uso das similaridades entre os filtros passa-alta e passa-baixa, facilitando e acelerando a implementação e conseqüentemente, o cálculo dos coeficientes [75] [78].

O sinal a ser analisado, é processado de acordo com as seguintes etapas: *split*, *prediction* e *update*. A primeira etapa do algoritmo *lifting* consiste em separar as amostras do sinal de entrada em sub-seqüências pares e em sub-seqüências ímpares. Este passo, conhecido como *split*, refere-se à divisão das amostras do sinal original e, após essa divisão, pode-se realizar o passo *predict* que consiste em prever a sub-seqüência indexada com amostras ímpares por intermédio da sub-seqüência indexada com amostras pares e a predição do erro é calculada. Deste modo, a sub-seqüência indexada com as amostras ímpares serão substituídas com o valor encontrado pela predição do erro.

O passo seguinte, referenciado na literatura como *update*, consiste em atualizar a sub-seqüência indexada com amostras pares a partir da sub-seqüência que contém as amostras ímpares. Repetindo este processo, a TWD pode ser implementada.

A Fig. 8.1 ilustra estas etapas de construção do algoritmo *lifting*, em que $\lambda_{j,k}$ representa o sinal de entrada do *j*-ésimo nível de decomposição e representado por *k*-ésimas amostras da seqüência.

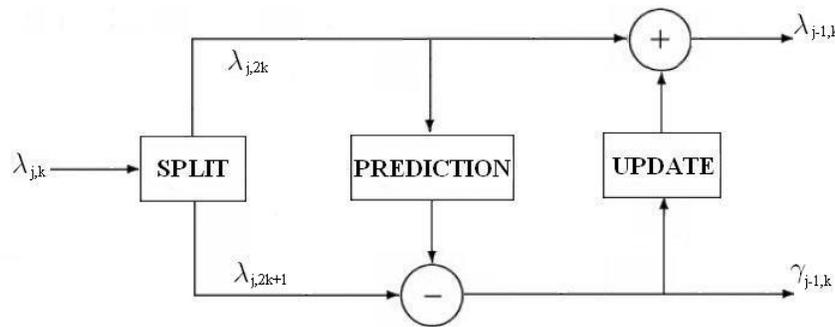


Figura 8.1: Decomposição do esquema *lifting* [79].

O sinal de entrada, $\lambda_{j,k}$, é dividido em um subconjunto de amostras pares e em um subconjunto de amostras ímpares, representados respectivamente por $\lambda_{j,2k}$ e por $\lambda_{j,2k+1}$. Os coeficientes *wavelets*, $\gamma_{j-1,k}$, podem ser calculados após as amostras do sinal de entrada passarem por esta divisão em subconjuntos e, desta forma, tem-se que:

$$\gamma_{j-1,k} = \lambda_{j,2k+1} - P(\lambda_{j,2k}) \quad (8.1)$$

Pode ser observado em (8.1) que P refere-se à função de predição e o sinal $\gamma_{j-1,k}$ refere-se à saída do filtro passa-alta. Em muitos casos, a função de predição (P) faz uso de interpolação polinomial de ordem $N-1$ [79].

De posse da obtenção dos coeficientes *wavelets*, pode-se agora atualizar o conjunto de amostras pares e assim, calcular os coeficientes da função escala, da forma:

$$\lambda_{j-1,k} = \lambda_{j,2k} + U(\lambda_{j,2k+1}) \quad (8.2)$$

O sinal $\lambda_{j-1,k}$ corresponde à saída do filtro passa-baixa.

Para um sinal não estacionário, usualmente não é possível realizar uma predição exata do sinal. Contudo, a técnica de predição possibilita uma diferença muito pequena entre o sinal original e o sinal obtido pela predição [79].

Se o sub-conjunto de amostras ímpares $\lambda_{j,2k+1}$ for substituído pela seqüência de predição do erro, $\gamma_{j-1,k}$, poucos bits podem ser usados para codificar essas amostras quando comparado às amostras indexadas como ímpares.

Vale comentar que o método *lifting* empregado na pesquisa, baseia-se na Fig. 8.1, fazendo uso de apenas duas bandas. Análises envolvendo mais bandas podem ser encontradas na literatura [6] [80]. A Fig. 8.2 mostra a esquematização para três bandas.

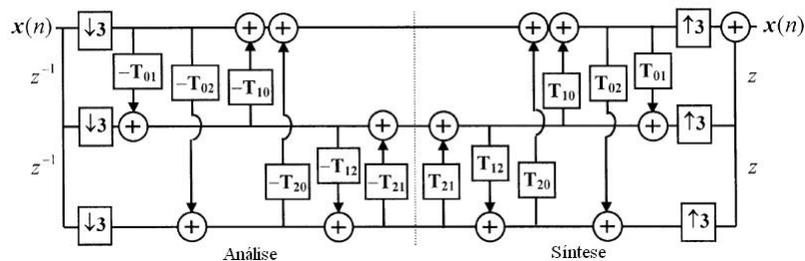


Figura 8.2: Método *lifting* representado por três bandas [6].

8.2 Decomposição da Wavelet de Haar Utilizando Lifting

A decomposição da transformada *wavelet* clássica de Haar consiste de sucessivos passos, em que cada passo corresponde a um nível de resolução. Cada passo da decomposição de Haar calcula médias e diferenças referentes ao sinal de entrada.

O sinal de entrada, $\lambda_{j,k}$, primeiramente é dividido em amostras pares $\lambda_{j,2k}$ e ímpares $\lambda_{j,2k+1}$ que, ao passar pelo primeiro nível de decomposição, resulta em uma média $\lambda_{j-1,k}$ e detalhes (diferenças) $\gamma_{j-1,k}$, correspondentes ao nível $j-1$, representados por:

$$\lambda_{j-1,k} = \frac{\lambda_{j,2k} + \lambda_{j,2k+1}}{2} \quad (8.3)$$

$$\gamma_{j-1,k} = \lambda_{j,2k+1} - \lambda_{j,2k} \quad (8.4)$$

A diferença é tomada entre as amostras adjacentes (pares e ímpares) e, durante o processo de decomposição, tais diferenças são armazenadas ao invés das amostras ímpares [81]. O passo seguinte da decomposição, repete o mesmo procedimento da transformada sob a média do sinal $\lambda_{j-1,k}$ e, este sinal torna-se a entrada do próximo nível de decomposição.

Os detalhes são fundamentais para reconstrução do sinal e as amostras pares e ímpares podem ser representadas como:

$$\lambda_{j,2k+1} = \lambda_{j-1,k} + \gamma_{j-1,k}/2 \quad (8.5)$$

$$\lambda_{j,2k} = \lambda_{j-1,k} - \gamma_{j-1,k}/2 \quad (8.6)$$

De posse dessa reconstrução, (8.6) pode ser reescrito em termos da média do sinal obtido na saída do primeiro nível de decomposição como:

$$\lambda_{j-1,k} = \lambda_{j,2k} + \gamma_{j-1,k}/2 \quad (8.7)$$

Em (8.7) é mostrado que ao invés de calcular médias e diferenças simultaneamente, primeiro calcula-se as diferenças e depois as médias [81]. A Fig. 8.3 ilustra a decomposição de Haar.

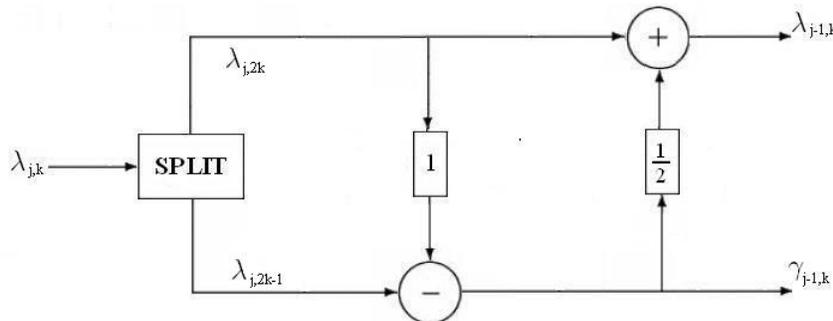


Figura 8.3: Implementação do esquema *lifting* da transformada de Haar [81].

Esse processo apresentado refere-se ao método *lifting* para a transformada de Haar. Tanto para a decomposição de Haar quanto para outras *wavelets*, o cálculo dos coeficientes de detalhes é denominado de passo *dual lifting*, enquanto o cálculo envolvendo as médias, por sua vez, é referenciado

como passo *primal lifting*.

Essa reorganização da transformada de Haar possui um impacto fundamental na interpretação dos coeficientes. Na implementação tradicional, as diferenças são calculadas junto com as médias e estas são interpretadas como a informação necessária para reconstruir a entrada a partir das médias, sendo que na implementação *lifting*, médias e diferenças são obtidas separadamente [81].

8.3 Implementação do Esquema *Lifting*: Decomposição e Reconstrução

A TWD pode ser obtida por meio de um determinado número de passos *lifting*. Considerando $\lambda_{j,k}$ como o sinal de entrada de comprimento 2^j , a transformada direta do sinal, que também pode ser referenciada como Análise, implica em uma decomposição do sinal que pode ser realizada j vezes até ser obtida apenas uma amostra do sinal.

Assumindo uma decomposição do sinal de entrada em 3 níveis, o sinal $\lambda_{j,k}$ após passar pelo primeiro nível de decomposição, implica no surgimento dos sinais $\gamma_{j-1,k}$ e $\lambda_{j-1,k}$, correspondentes às seqüências de diferenças e da média do sinal de entrada, respectivamente. Este processo é iterativo e realizado até o nível de decomposição desejado [82].

O sinal referente à média ou aproximação do sinal, $\lambda_{j-1,k}$, passa a ser o sinal de entrada a ser processado no próximo nível de decomposição e conseqüentemente, dando origem aos sinais $\lambda_{j-2,k}$ e $\gamma_{j-2,k}$. Os sinais $\lambda_{j-3,k}$ e $\gamma_{j-3,k}$ são obtidos e caracterizados de acordo com o sinal $\lambda_{j-2,k}$.

Considerando que o sinal de entrada possui comprimento 2^j , o sinal $\lambda_{j-1,k}$ passa a ter comprimento de 2^{j-1} e assim sucessivamente, implicando conseqüentemente nos comprimentos 2^{j-2} e 2^{j-3} referentes ao segundo e terceiro estágio de decomposição, respectivamente.

Na Fig. 8.4 pode ser mostrada a TWD do sinal em três níveis de decomposição.

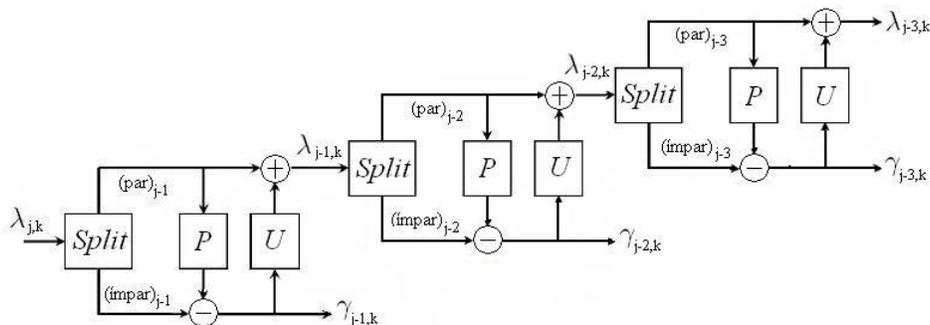


Figura 8.4: TWD em 3 níveis ($j=3$) [82].

A Tabela 8.1 ilustra a decomposição para o caso particular do sinal de entrada, $\lambda_{j,k}$, processado

mediante as imposições estabelecidas ($j=3$ e comprimento 2^j). Considera-se como sinal de entrada um sinal de 8 amostras representados por λ_3 , e a cada nível de decomposição do sinal obtém-se como saída a metade das amostras de entrada e o detalhamento referente à diferença entre estes níveis de decomposições. Como percebido pela Fig. 8.4, o sinal de detalhe referente a cada nível de decomposição é armazenado e apenas a média das amostras de entrada é processada pelo nível seguinte.

λ_3 [0]	λ_3 [1]	λ_3 [2]	λ_3 [3]	λ_3 [4]	λ_3 [5]	λ_3 [6]	λ_3 [7]
λ_2 [0]	λ_2 [1]	λ_2 [2]	λ_2 [3]	γ_2 [0]	γ_2 [1]	γ_2 [2]	γ_2 [3]
λ_1 [0]	λ_1 [1]	γ_1 [0]	γ_1 [1]	γ_2 [0]	γ_2 [1]	γ_2 [2]	γ_2 [3]
λ_0 [0]	γ_0 [0]	γ_1 [0]	γ_1 [1]	γ_2 [0]	γ_2 [1]	γ_2 [2]	γ_2 [3]

Tabela 8.1: Esquemática dos parâmetros na decomposição por esquema *lifting* [82].

Após realizar a decomposição do sinal, estes passos *lifting* podem ser facilmente desfeitos. Tomando como exemplo a Fig. 8.5, o sinal $\lambda_{j-2,k}$ pode ser obtido por intermédio da junção dos sinais $\lambda_{j-3,k}$ e $\gamma_{j-3,k}$. Os blocos *update* e *prediction* também estão presentes na reconstrução do sinal, porém com estes blocos desenvolvendo operações inversas ao modo de decomposição. Em vez de ser realizado a adição é feita a subtração no que se refere ao bloco *update* e, seguindo o mesmo princípio, ao invés de realizar a subtração no bloco *prediction* é realizado a adição.

A Fig. 8.5 complementa a Fig. 8.4 ilustrando a reconstrução do sinal original em 3 níveis.

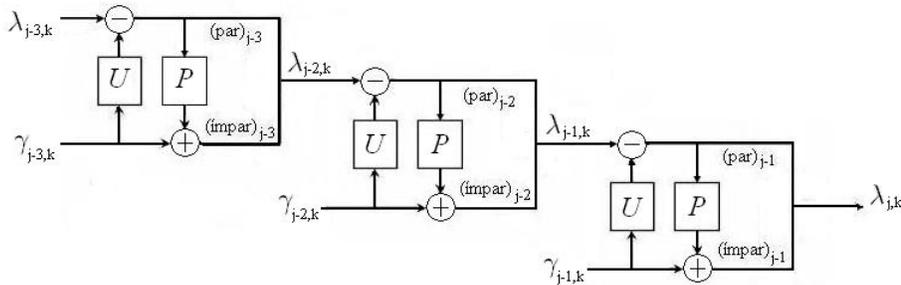


Figura 8.5: Reconstrução do sinal em 3 níveis [82].

8.4 Transformada Wavelet e o Esquema Lifting

A TWD direta consiste de dois filtros de análise, \tilde{h} (passa-baixa) e \tilde{g} (passa-alta), seguida por sub-amostragem de cada saída destes filtros. Na TWD inversa, primeiro realiza-se a super-amostragem do sinal e, então, utiliza-se dois filtros de síntese, h (passa-baixa) e g (passa-alta), para reconstrução do sinal original. A Fig. 8.6 ilustra o esquema da transformada *wavelet*.

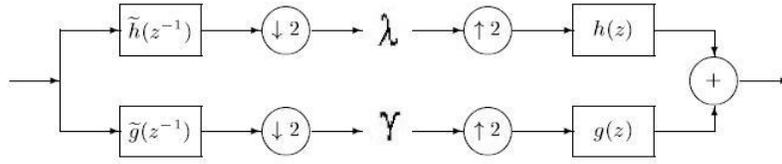


Figura 8.6: TWD direta e inversa do sinal [1].

Os filtros da transformada *wavelet* (h , g , \tilde{h} e \tilde{g}) empregados na decomposição e reconstrução do sinal são filtros FIR e a condição para reconstrução perfeita é dada por [1] [82]:

$$\begin{aligned} h(z) \tilde{h}(z^{-1}) + g(z) \tilde{g}(z^{-1}) &= 2 \\ h(z) \tilde{h}(-z^{-1}) + g(z) \tilde{g}(-z^{-1}) &= 0 \end{aligned} \quad (8.8)$$

Almejando reconstrução perfeita, as condições apresentadas em (8.8) implicam que os quatro filtros não podem ser escolhidos independentemente [82].

A representação polifásica de um filtro h é dada por:

$$h(z) = h_e(z^2) + z^{-1}h_o(z^2) \quad (8.9)$$

onde h_e representa os coeficientes pares e os coeficientes ímpares representados por h_o . Desta forma, tem-se que:

$$h_e(z) = \sum_k h_{2k} z^{-k} \text{ e } h_o(z) = \sum_k h_{2k+1} z^{-k} \quad (8.10)$$

ou podem ser representados como:

$$h_e(z^2) = \frac{h(z) + h(-z)}{2} \text{ e } h_o(z^2) = \frac{h(z) - h(-z)}{2z^{-1}} \quad (8.11)$$

Deste modo, a matriz polifásica pode ser representada por:

$$P(z) = \begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix} \quad (8.12)$$

A matriz $\tilde{P}(z)$ é definida de modo similar e, deste modo, a propriedade de reconstrução perfeita é caracterizada por:

$$\tilde{P}(z^{-1})^t P(z) = I \quad (8.13)$$

É desejado que $P(z)$ e $\tilde{P}(z)$ sejam representados por polinômios de Laurent e conseqüentemente, a transformada *wavelet* é representada pela Fig. 8.7.

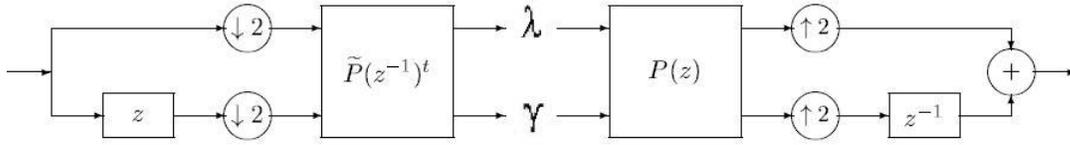


Figura 8.7: Representação polifásica da transformada *wavelet* [1].

Em (8.13), implica que o determinante de $P(z)$ e sua inversa sejam ambos polinômios de Laurent, e isso é possível somente no caso do determinante de $P(z)$ ser um monômio em z [1].

O problema de se encontrar uma transformada *wavelet* inversa utilizando filtros FIR, resulta em encontrar uma matriz $P(z)$ com determinante igual a 1, ou seja, $h_e(z)g_o(z) - h_o(z)g_e(z) = 1$. Deste modo, $\tilde{P}(z)$ e os quatro filtros para a transformada *wavelet* podem ser representados a partir de (8.13) e da regra de Cramer, obtendo-se:

$$P(z)^{-1} = \tilde{P}(z^{-1})^t = \frac{1}{h_e(z)g_o(z) - h_o(z)g_e(z)} \begin{pmatrix} g_o(z) & -g_e(z) \\ -h_o(z) & h_e(z) \end{pmatrix} \quad (8.14)$$

Conseqüentemente, têm-se as seguintes relações [1] [82] :

$$\begin{aligned} \tilde{h}_e(z) &= g_o(z^{-1}) \\ \tilde{h}_o(z) &= -g_e(z^{-1}) \\ \tilde{g}_e(z) &= -h_o(z^{-1}) \\ \tilde{g}_o(z) &= h_e(z^{-1}) \end{aligned} \quad (8.15)$$

E isso implica que:

$$\tilde{g}(z) = z^{-1}h(-z^{-1}) \quad e \quad \tilde{h}(z) = -z^{-1}g(-z^{-1}) \quad (8.16)$$

O exemplo mais trivial de uma matriz polifásica é quando $P(z) = I$, onde $h(z) = \tilde{h}(z) = 1$ e $g(z) = \tilde{g}(z) = z^{-1}$. A transformada *wavelet*, para este caso, apenas realiza uma sub-amostragem em amostras pares e ímpares. Essa transformada é denominada de transformada polifásica, porém no contexto da teoria *lifting* é referenciada como transformada *wavelet* Lazy [1].

Definição 2

Um par de filtros (h, g) é complementar caso a matriz polifásica correspondente, $P(z)$, apresente determinante igual a 1. Se (h, g) for complementar, então (\tilde{h}, \tilde{g}) também será. Para o caso especial em que $h = \tilde{h}$ e $g = \tilde{g}$, a transformada *wavelet* é ortogonal e caso contrário, biortogonal [1].

Teorema 5

Considerando o par de filtros (h, g) sendo complementar, implica que qualquer outro filtro finito g^{new} complementar a h é representado como [1]:

$$g^{new}(z) = g(z) + h(z) s(z^2) \quad (8.17)$$

onde $s(z)$ é um polinômio de Laurent.

Os componentes polifase de $h(z) s(z^2)$ refere-se à $h_e(z) s(z)$ (pares) e $h_o(z) s(z)$ (ímpares). Depois do passo *primal lifting*, a nova matriz polifásica é, conseqüentemente, representada por:

$$P^{new}(z) = \begin{pmatrix} h_e(z) & h_e(z) s(z) + g_e(z) \\ h_o(z) & h_o(z) s(z) + g_o(z) \end{pmatrix} = P(z) \begin{bmatrix} 1 & s(z) \\ 0 & 1 \end{bmatrix} \quad (8.18)$$

Essa operação não altera o determinante da matriz polifásica.

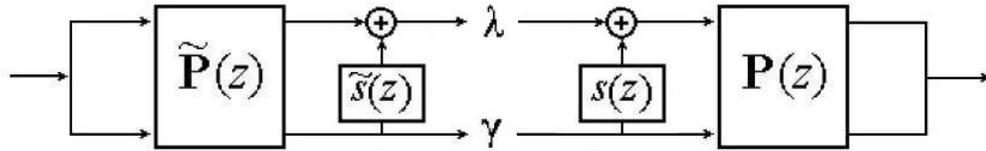
De modo similar e de acordo com (8.15), o esquema *lifting* pode ser aplicado para originar o filtro $\tilde{h}^{new}(z)$ complementar à $g(z)$.

$$\tilde{h}^{new}(z) = \tilde{h}(z) + \tilde{g}(z) s(z^2) \quad (8.19)$$

Vale lembrar que o equacionamento é realizado de acordo com o critério adotado na esquematização. Neste contexto, a matriz polifásica *dual* pode ser representada, conseqüentemente, por:

$$\tilde{P}^{new}(z) = \begin{pmatrix} \tilde{h}_e(z) + \tilde{g}_e(z) \tilde{s}(z) & \tilde{h}_o(z) + \tilde{g}_o(z) \tilde{s}(z) \\ \tilde{g}_e(z) & \tilde{g}_o(z) \end{pmatrix} = \tilde{P}(z) \begin{bmatrix} 1 & \tilde{s}(z) \\ 0 & 1 \end{bmatrix} \quad (8.20)$$

Estes passos correspondem à etapa *primal lifting*, que em outras palavras corresponde em realizar o *lifting* da sub-banda passa-baixa a partir da sub-banda passa-alta. A Fig. 8.8 ilustra o diagrama para este esquema.

Figura 8.8: Esquema *primal lifting*.**Teorema 6**

Considere o par de filtros (h, g) sendo complementar. Qualquer outro filtro finito g^{new} complementar a h é representado como [1]:

$$h^{new}(z) = h(z) + g(z)t(z^2) \quad (8.21)$$

onde $t(z)$ é um polinômio de Laurent. Depois de realizar o passo *dual lifting*, a nova matriz polifásica é representada por:

$$P^{new}(z) = \begin{pmatrix} h_e(z) + g_e(z)t(z) & g_e(z) \\ h_o(z) + g_o(z)t(z) & g_o(z) \end{pmatrix} = P(z) \begin{bmatrix} 1 & 0 \\ t(z) & 1 \end{bmatrix} \quad (8.22)$$

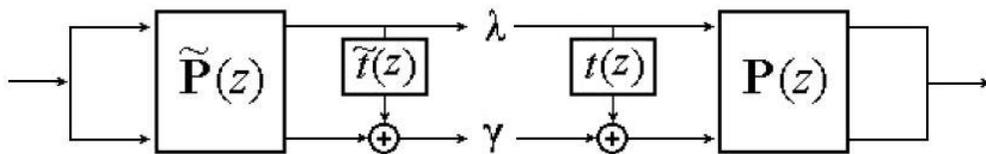
Desta forma, é originado um novo filtro \tilde{g} por meio do *dual lifting*, designado por:

$$\tilde{g}^{new}(z) = \tilde{g}(z) + \tilde{h}(z)t(z^2) \quad (8.23)$$

Onde a matriz polifásica *dual*, por sua vez, pode ser representada como:

$$\tilde{P}^{new}(z) = \begin{pmatrix} \tilde{h}_e(z) & \tilde{h}_o(z) \\ \tilde{g}_e(z) + \tilde{h}_e(z)\tilde{t}(z) & \tilde{g}_o(z) + \tilde{h}_o(z)\tilde{t}(z) \end{pmatrix} = \tilde{P}(z) \begin{bmatrix} 1 & 0 \\ \tilde{t}(z) & 1 \end{bmatrix} \quad (8.24)$$

Vale lembrar que o equacionamento é realizado de acordo com o critério adotado na esquematização. O esquema *dual lifting*, ao contrário do *primal lifting*, é responsável por realizar o *lifting* da sub-banda passa-alta a partir da sub-banda passa-baixa, conforme ilustrado pela Fig. 8.9.

Figura 8.9: Esquema *dual lifting*.

Vale destacar que o passo *dual lifting* corresponde ao passo *predict* e conseqüentemente, *primal lifting* equivale ao passo *update*. Os passos *primal lifting* e *dual lifting* são utilizados para construir transformadas *wavelets* começando a partir da *wavelet* Lazy, obtendo deste modo, um grupo de *wavelets* [77]. É natural construir *wavelets* utilizando estes passos *lifting* e em [1] mostra-se que qualquer transformada *wavelet* utilizando filtros finitos pode ser obtida começando a partir da *wavelet* Lazy, seguido alternativamente por um número finito de passos *lifting* e *dual lifting*.

8.5 O Algoritmo Euclidiano e a Fatoração

O algoritmo euclidiano foi originalmente desenvolvido para encontrar o maior divisor comum de dois números naturais, mas pode ser estendido para encontrar o maior divisor comum de dois polinômios. A principal diferença com o caso polinomial é que a solução não é única. Dois polinômios de Laurent são relativamente primos para o caso em que o maior divisor comum possuir grau zero.

Assumindo dois polinômios de Laurent $a(z)$ e $b(z) \neq 0$ com $|a(z)| \geq |b(z)|$, então sempre existe um polinômio $q(z)$ (quociente), em que $q(z) = |a(z)| - |b(z)|$, e um polinômio $r(z)$ (resto), em que $|r(z)| < |b(z)|$, tendo assim:

$$a(z) = b(z)q(z) + r(z) \quad (8.25)$$

E, desta forma, $q(z)$ e $r(z)$ podem ser representados por:

$$q(z) = a(z)/b(z) \text{ e } r(z) = a(z) \% b(z) \quad (8.26)$$

Teorema 7

Assumindo dois polinômios de Laurent $a(z)$ e $b(z) \neq 0$ com $|a(z)| \geq |b(z)|$ e considerando $a_0(z) = a(z)$ e $b_0(z) = b(z)$, tem-se que:

$$a_{i+1}(z) = b_i(z) \quad (8.27)$$

$$b_{i+1}(z) = a_i(z) \% b_i(z) \quad (8.28)$$

$$q_{i+1} = a_i(z) / b_i(z) \quad (8.29)$$

Deste modo, $a_n(z) = mdc(a(z), b(z))$, onde n é o menor número em que $b_n(z) = 0$. Em (8.28), $b_{i+1}(z)$ refere-se ao resto entre os polinômios.

O resultado desse algoritmo pode ser escrito da forma:

$$\begin{bmatrix} a(z) \\ b(z) \end{bmatrix} = \prod_{i=1}^n \begin{bmatrix} q_i(z) & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} a_n(z) \\ 0 \end{bmatrix} \quad (8.30)$$

Se $a_n(z)$ for um monômio, então $a(z)$ e $b(z)$ são relativamente primos [1].

Teorema 8

Dado que a longa divisão dos polinômios de Laurent não resulta necessariamente em apenas um modo de solução, devido ao fato de diversos modos da fatorização tornarem possíveis, pode-se escolher os quocientes desde que o mdc seja uma constante, onde esta constante será referenciada como C .

Pode-se realizar a fatoração de um filtro FIR (*Finite Impulse Response* - Resposta ao Impulso Finita) de uma transformada *wavelet* existente em passos *lifting*. Portanto, dado um par de filtros complementar (h, g) , sempre existirá polinômios de Laurent $s_i(z)$ e $t_i(z)$ para $1 \leq i \leq m$ e para uma constante, C , que não seja zero, desde que [1]:

$$P(z) = \prod_{i=1}^m \begin{bmatrix} 1 & s_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} C & 0 \\ 0 & 1/C \end{bmatrix} \quad (8.31)$$

Em outras palavras, toda transformada *wavelet* com filtros finitos pode ser obtida começando com a *wavelet* Lazy seguida por m passos *lifting* e *dual lifting* [1].

8.6 Implementação da Decomposição e Reconstrução

Conforme observado pela transformada de Haar, Fig. 8.3, sua decomposição pode ser caracterizada por um passo *prediction* seguido de um passo *update*. Porém, cada *wavelet* quando representada via passos *lifting*, pode ser caracterizada por múltiplos passos *prediction* e *updates*, sendo que estes passos não precisam ocorrer em pares e podem serem adicionados e conseqüentemente, resultando em sucessivos cálculos de médias e detalhes [81] [82].

Neste sentido, implementações por método *lifting*, fazem com que as *wavelets* sejam representadas através do número de passos *prediction* e *update*. Para a transformada de Haar, a implementação para a transformada direta pode ser representada por [1] [82]:

$$\begin{aligned}
s_l^{(0)} &= x_{2l} \\
d_l^{(0)} &= x_{2l+1} \\
d_l &= d_l^{(0)} - s_l^{(0)} \\
s_l &= s_l^{(0)} + 1/2d_l
\end{aligned} \tag{8.32}$$

Em (8.32), pode ser percebido que $s_l^{(0)}$ refere-se às amostras pares e $d_l^{(0)}$ às amostras ímpares. Aqui, os sinais d_l e s_l são utilizados para representar a saída do processo de decomposição, baseado nos sinais de detalhes e de média, respectivamente.

A transformada inversa, para esta transformada, pode ser obtida por:

$$\begin{aligned}
s_l^{(0)} &= s_l - 1/2d_l \\
d_l^{(0)} &= d_l + s_l^{(0)} \\
x_{2l+1} &= d_l^{(0)} \\
x_{2l} &= s_l^{(0)}
\end{aligned} \tag{8.33}$$

O sinal reconstruído, representado por (8.33), é caracterizado por desfazer as operações utilizadas na decomposição.

A complexidade da implementação vai aumentando conforme aumenta-se o comprimento da *wavelet*. Para ilustrar essa afirmação, a *wavelet* de Daubechies de ordem 4 (dB4), que também foi utilizada nas implementações práticas do Capítulo 9, apresenta a seguinte implementação para a transformada direta [1] [82]:

$$\begin{aligned}
d_l^{(1)} &= x_{2l+1} - \sqrt{3}x_{2l} \\
s_l^{(1)} &= x_{2l} + \sqrt{3}/4d_l^{(1)} + (\sqrt{3}-2)/4d_{l+1}^{(1)} \\
d_l^{(2)} &= d_l^{(1)} + s_{l-1}^{(1)} \\
s_l &= (\sqrt{3}+1)/\sqrt{2}s_l^{(1)} \\
d_l &= (\sqrt{3}-1)/\sqrt{2}d_l^{(2)}
\end{aligned} \tag{8.34}$$

E sua inversa é representada da seguinte forma:

$$\begin{aligned}
d_l^{(2)} &= (\sqrt{3}+1)/\sqrt{2}d_l \\
s_l^{(1)} &= (\sqrt{3}-1)/\sqrt{2}s_l \\
d_l^{(1)} &= d_l^{(2)} - s_{l-1}^{(1)} \\
x_{2l} &= s_l^{(1)} - \sqrt{3}/4d_l^{(1)} - (\sqrt{3}-2)/4d_{l+1}^{(1)} \\
x_{2l+1} &= d_l^{(1)} + \sqrt{3}x_{2l}
\end{aligned} \tag{8.35}$$

Percebe-se que, dentre outras diferenças com a *wavelet* de Haar, esta não apresenta os passos *prediction* e *update* em pares. A forma de implementação caracterizada por cada *wavelet* é fundamental para obtenção dos resultados no Capítulo 9.

8.7 Complexidade Computacional

Conforme analisado nas seções anteriores, pelo fato do método *lifting* sub-amostrar o sinal de entrada antes de realizar o processo de filtragem, o processo é executado com metade das amostras fazendo uso de um menor número de operações aritméticas e, deste modo, um rápido processamento é alcançado por fazer ótimo uso das similaridades entre os filtros passa-alta e passa-baixa, facilitando e acelerando a implementação e conseqüentemente, o cálculo dos coeficientes.

Neste sentido, o ganho proporcionado pelo método *lifting* refere-se ao menor número de operações aritméticas (multiplicações e adições) necessário para realizar a TWD, quando comparado ao método tradicional [1] [2]. Este ganho apresenta maior valor conforme o comprimento do filtro da *wavelet* aumenta e, segundo Daubechies e Sweldens, obtendo ganho de até duas vezes para esta circunstância [1]. Porém, ao contrário desta afirmação, Reichel afirma que ganho de até quatro vezes pode ser obtido [2].

Daubechies e Sweldens, afirma que o custo computacional obtido para o método *lifting* pode atingir, para filtros de maiores comprimentos, a metade de operações aritméticas e conseqüentemente, implicando em ganho de dois sobre o método tradicional. Para as *wavelets* da família de Daubechies, foram apresentados a proporção deste ganho para as de ordem 4 (dB4) e 6 (dB6), obtendo 56% e 57%, respectivamente.

No estudo realizado por Reichel, é realizado uma comparação em termos do número de passos *lifting* empregados durante o processo. Este número de passos, referenciados por K , refere-se à quantidade de passos *primal* e *dual lifting* utilizados na realização da TWD, onde o ganho torna-se maior conforme aumenta-se a quantidade destes passos [2]. Sua análise destinou-se a apresentar o ganho aritmético alcançado quando o uso de $K = 2$ e $K = 4$ foi empregado. Para comprovar

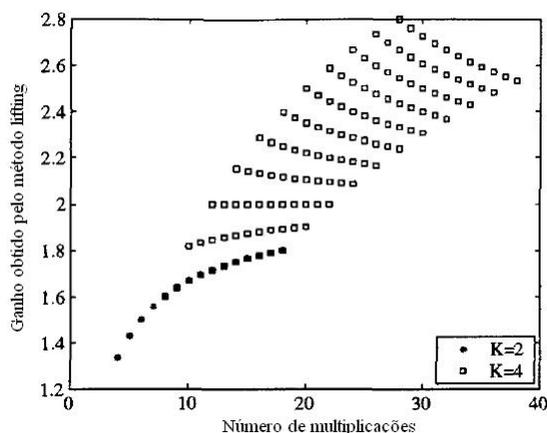


Figura 8.10: Ganho aritmético alcançado pelo método *lifting* [2].

proporções maiores que duas vezes para o ganho, a Fig. 8.10 ilustra a proporção do ganho aritmético multiplicativo obtido pelo método *lifting* sob o método tradicional, onde as curvas foram obtidas para diferentes tamanhos de filtros *lifting*.

Vale frisar que estes ganhos foram determinados teoricamente e, neste sentido, o Capítulo 9 consiste em abordar em que proporção este ganho pode ser obtido na prática, através de imagens de diferentes resoluções como sinal de entrada para realização da TWD, empregando diferentes *wavelets*, onde as implementações foram analisadas via ferramenta interpretada e compilada.

8.8 Comentários

Foram apresentadas as principais características que compõem a teoria do método *lifting* e suas diferenças com o método tradicional abordado nos Capítulos 3, 4 e 5. O conceito de matriz polifásica é amplamente empregado na literatura e, por isso, fez-se necessário sua apresentação dentro do contexto *lifting*, assim como o algoritmo euclidiano, enfatizando o conceito de que iniciando a partir da *wavelet Lazy* e utilizando *lifting*, pode-se construir gradualmente análise multiresolução com propriedades particulares. O método *lifting* foi desenvolvido buscando minimizar a complexidade da TWD, com o intuito de facilitar a implementação realizando a sub-amostragem antes dos processos de filtros e conseqüentemente, acelerando o cálculo dos coeficientes. Neste sentido, o Capítulo 9 mostra no quanto esta aceleração teórica reflete na realização da TWD através de imagens de diferentes resoluções, empregando diferentes *wavelets* e realizando implementações via ferramenta interpretada e compilada.

Capítulo 9

RESULTADOS EXPERIMENTAIS

Na literatura, estudos teóricos foram realizados em busca de obter-se afirmações quanto à proporção do ganho aritmético obtido entre o método *lifting* e o método tradicional. Daubechies e Sweldens propuseram o método *lifting*, que faz uso de um número menor de operações aritméticas chegando-se a atingir, no caso de comprimentos de filtros maiores, metade das operações do método tradicional [1]. Em contraposição, Reichel [2] afirma que este ganho pode ter proporção ainda maior, podendo atingir cerca de quatro vezes o método tradicional.

Embora tais estudos apresentem uma divergência a respeito do ganho proporcionado, a efetividade na aceleração no cálculo dos coeficientes é um fator extremamente atrativo, minimizando a complexidade da TWD e acelerando o tempo de processamento [75] [77]. Adicionalmente, ambos afirmam que este ganho deve tomar proporções maiores na medida em que o comprimento da *wavelet* torna-se maior.

Buscando dar seqüência à mesma linha destas comparações teóricas, esta dissertação objetiva realizar uma análise prática do custo computacional com ênfase para o tempo de processamento comparativo entre ambos os métodos. Como se sabe, métodos *wavelet* são adequados para aplicações que requerem codificação rápida, por exemplo, em comunicações em tempo real via *internet* e, neste sentido, velocidade é um dos parâmetros fundamentais levados em conta na elaboração de projetos. Neste sentido, a Seção 9.2 objetiva apresentar as simulações e análises comparativas com relação ao desempenho em termos de tempo de processamento (análise objetiva) entre o método *lifting* e o método tradicional para diversas resoluções, imagens-teste e diferentes tipos de *wavelets*, dando seqüência a mesma linha de estudos teóricos presentes em [1] e [2]. A presente seção encontra-se dividida em resultados obtidos via ferramenta compilada (Seção 9.2.1) e via ferramenta interpretada (Seção 9.2.2).

Ambos os métodos realizam a decomposição e reconstrução do sinal por métodos matemáticos distintos, mas visando o mesmo sinal reconstruído, conforme discutido adiante.

As condições e particularidades de simulações são apresentadas na Seção 9.1, cabendo a Seção 9.2 abordar os resultados diante de uma análise objetiva, onde a reconstrução perfeita é alcançada e ilustrada na Seção 9.3. A Seção 9.4 realiza uma breve abordagem sobre os resultados obtidos, no que diz respeito as diferenças entre as ferramentas utilizadas. O sumário da análise dos resultados encontra-se na Seção 9.5. Comentários finalizam o capítulo na Seção 9.6.

9.1 Observações sobre as Simulações

As simulações foram realizadas utilizando duas ferramentas distintas, compilada e interpretada. A primeira, refere-se ao ambiente de programação Dev-C++ 4.9.9.2. A segunda, por sua vez, refere-se ao ambiente computacional Matlab 7.1, *software* utilizado como ambiente de programação para a execução, visualização de imagens e construção de gráficos.

Dentro das simulações realizadas nestas duas ferramentas, foram utilizadas imagens de tamanho 256x256 e 512x512 *pixels* em escala de cinza, disponíveis no *website* Waterloo Bragzone¹. Essas imagens são de domínio público e extensamente utilizadas na literatura científica do processamento digital de imagem. As Figs. 9.1 e 9.2 ilustram as imagens testadas, respectivamente, gravadas em formato Bitmap. Tanto para imagens 256x256 como para imagens 512x512, a comparação de tempos de processamento entre os métodos *lifting* e tradicional foi realizada para *wavelets* Daubechies, Symlets e Biortogonais.

¹<http://links.uwaterloo.ca>

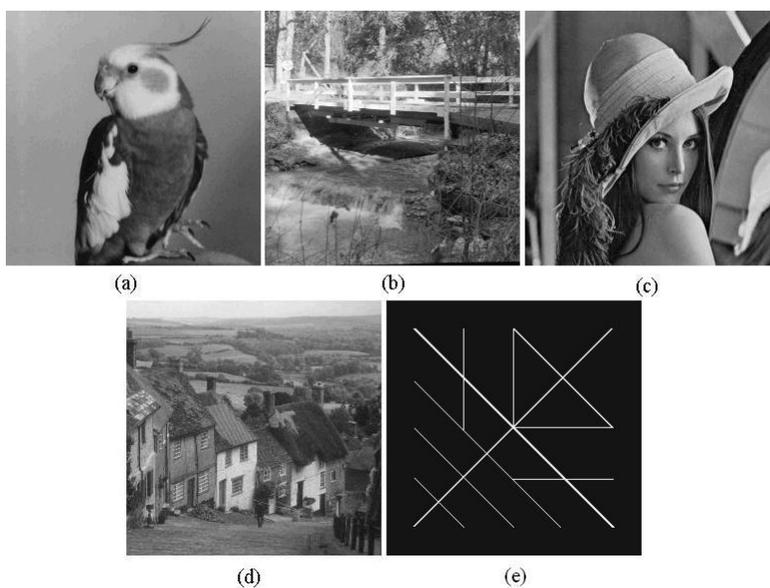


Figura 9.1: Imagens testadas (256×256 pixels): (a) Bird; (b) Bridge; (c) Lena; (d) Goldhill; (e) Crosses.



Figura 9.2: Imagens testadas (512×512 pixels): (a) Barb ; (b) Boat ; (c) Goldhill ; (d) Mandrill; (e) Lena ; (f) Washsat ; (g) Peppers ; (h) Zelda.

O Matlab apresenta algumas limitações para a decomposição e reconstrução das imagens, quando faz-se o uso do método *lifting*. Conseqüentemente, as *wavelets* da família Daubechies ficaram limitadas dentre as *wavelets* de ordem 1 até a de ordem 8. Para as *wavelets* de Symlets e as Biortogonais,

todas as *wavelets* da família foram utilizadas. Na análise via ferramenta compilada, o mesmo critério foi adotado visando comparações futuras.

A medida referente ao tempo de processamento, depende fortemente do desempenho do processador utilizado. Todas as simulações foram desenvolvidas utilizando o mesmo processador e enquanto as simulações foram executadas, o mesmo realizou apenas esta tarefa. O processador utilizado refere-se ao Intel Core 2 Duo com 3,16GHz e 2Gb de memória RAM. Tomou-se o cuidado de todo resultado envolvendo a medida de tempo de processamento, ser obtido pelo intermédio da realização da média tomada durante 1000 execuções de cada situação, de modo a garantir precisão.

A comparação realizada levou em consideração a decomposição e reconstrução do sinal em três níveis, valor este usualmente empregado para análise de imagens. Durante este processo, os coeficientes de aproximação, de detalhes horizontais, verticais e diagonais foram obtidos para cada um dos três níveis, durante a realização da TWD.

Vale ressaltar que, para ambos os métodos empregados para decomposição e reconstrução, em todos níveis foram utilizados apenas duas bandas, embora algumas aplicações façam uso de multi-bandas para o método *lifting* [6].

Após realizar a decomposição e reconstrução das imagens, medidas de PSNR foram igualmente utilizadas de forma a corroborar a perfeita implementação do *lifting*, de posse das imagens reconstruídas.

9.2 Resultados com Análise Objetiva

Conforme comentado, estudos apontam a proporção em que o desempenho aritmético oferecido pelo método *lifting* supera o método tradicional para a realização da TWD [1] [2]. Embora tais estudos apresentem uma divergência a respeito do ganho proporcionado, a efetividade na aceleração no cálculo dos coeficientes é um fator extremamente atrativo, minimizando a complexidade da TWD e acelerando o tempo de processamento [75] [77]. Adicionalmente, ambos afirmam que este ganho deve tomar proporções maiores na medida em que o comprimento da *wavelet* torna-se maior.

A Seção 9.2 objetiva apresentar as simulações e análises comparativas com relação ao desempenho em termos de tempo de processamento (análise objetiva) entre o método *lifting* e o método tradicional para diversas resoluções, imagens-teste e diferentes tipos de *wavelets*, dando seqüência a mesma linha de estudos teóricos presentes em [1] e [2]. A presente seção encontra-se dividida em resultados obtidos via ferramenta compilada (Seção 9.2.1) e via ferramenta interpretada (Seção 9.2.2).

9.2.1 Implementação via Ferramenta Compilada

As simulações comparativas objetivas entre a realização da TWD pelo método *lifting* e pelo método tradicional são apresentadas nesta subseção, utilizando a ferramenta compilada C++. É avaliado o tempo de processamento necessário para decomposição e o tempo de processamento necessário para todo o processo (decomposição e reconstrução) em ambos os métodos (*lifting* e tradicional) em função do comprimento do filtro (n) para *wavelets* Daubechies, Symlets e Biortogonais.

Os resultados são analisados graficamente e quantitativamente, de modo a enfatizar a razão de velocidade obtida entre os métodos tradicional e *lifting*.

Nas análises seguintes, pode-se perceber que imagens de mesma resolução apresentam o mesmo tempo de processamento para realizar a TWD, além de deparar-se com um aumento no ganho conforme aumenta-se o comprimento do filtro das *wavelets*.

Imagens 256x256

O impacto do método *lifting* pode ser percebido, inicialmente, com o emprego da família de Daubechies, observando-se seu excelente desempenho no que tange à velocidade de processamento quando comparado ao método tradicional. A Fig. 9.3 apresenta em (a) o tempo necessário para executar apenas a decomposição e em (b) o tempo necessário para realizar a decomposição e reconstrução; considerando os valores médios de tempo de processamento (T_p) obtidos para o conjunto de imagens 256x256 (Fig. 9.1).

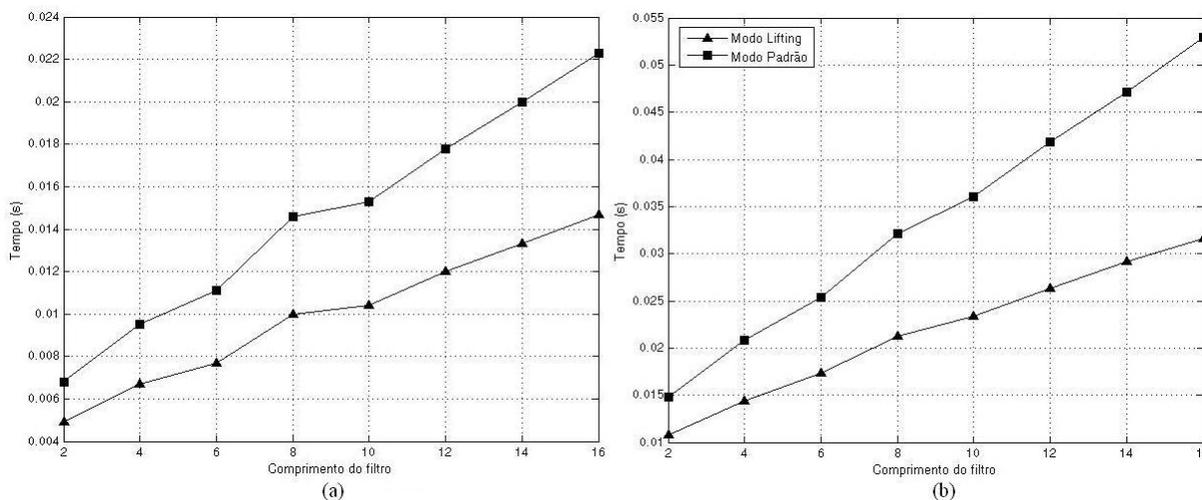


Figura 9.3: Comparação T_p x n para *wavelets* Daubechies utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

As Tabelas 9.1 e 9.2 apresentam, respectivamente, quantitativamente a proporção do ganho em

velocidade na decomposição e a proporção do ganho em velocidade na decomposição e reconstrução da TWD para *wavelets* Daubechies. Na decomposição, verifica-se ganho médio de 1,46 vezes, aplicando o método *lifting*. No processo completo, este ganho médio sobe para 1,52 vezes sobre o método tradicional.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
db1	2	0,0068	0,0049	1,39
db2	4	0,0095	0,0067	1,42
db3	6	0,0111	0,0077	1,44
db4	8	0,0146	0,0100	1,46
db5	10	0,0153	0,0104	1,47
db6	12	0,0178	0,0120	1,48
db7	14	0,0200	0,0133	1,50
db8	16	0,0223	0,0147	1,52

Tabela 9.1: Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* de Daubechies.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
db1	2	0,0148	0,0108	1,37
db2	4	0,0208	0,0144	1,44
db3	6	0,0254	0,0173	1,47
db4	8	0,0321	0,0212	1,51
db5	10	0,0360	0,0234	1,54
db6	12	0,0418	0,0263	1,59
db7	14	0,0471	0,0292	1,61
db8	16	0,0529	0,0316	1,67

Tabela 9.2: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* de Daubechies.

Conforme percebido, o ganho obtido cresce com o aumento do comprimento do filtro, cabendo ao método *lifting* apresentar um desempenho de até 1,67 vezes mais rápido que o método tradicional, durante o processo completo da TWD.

Para a análise das *wavelets* de Symlets, a Fig. 9.4 ilustra a comparação de desempenho do método tradicional e *lifting* para a TWD direta e para a TWD direta e inversa, novamente considerando valores médios obtidos para o mesmo conjunto de imagens 256x256 da Fig. 9.1.

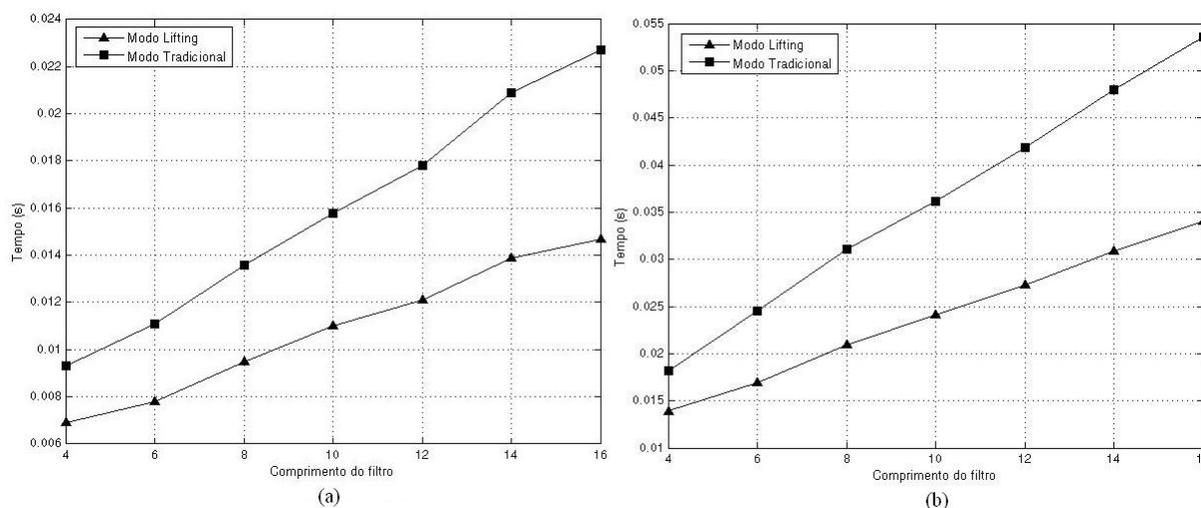


Figura 9.4: Comparação $T_p \times n$ para *wavelets* Symlets utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Novamente observa-se o forte ganho em tempo de processamento utilizando o método *lifting*. A proporção deste ganho pode ser melhor interpretada através das Tabelas 9.3 e 9.4.

<i>Wavelets</i>	n	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,0093	0,0069	1,35
sym3	6	0,0111	0,0078	1,42
sym4	8	0,0136	0,0095	1,43
sym5	10	0,0158	0,0110	1,44
sym6	12	0,0178	0,0121	1,47
sym7	14	0,0209	0,0139	1,50
sym8	16	0,0227	0,0147	1,54

Tabela 9.3: Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* Symlets.

<i>Wavelets</i>	n	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,0182	0,0139	1,31
sym3	6	0,0245	0,0169	1,45
sym4	8	0,0311	0,0209	1,49
sym5	10	0,0361	0,0241	1,50
sym6	12	0,0418	0,0273	1,53
sym7	14	0,0480	0,0309	1,55
sym8	16	0,0536	0,0340	1,58

Tabela 9.4: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* Symlets.

Novamente é percebido que o método *lifting* apresenta melhor desempenho que o método tradicional, revelando ganho médio de 1,45 vezes para decomposição e ganho médio de 1,49 vezes no processo completo. Nota-se similarmente às *wavelets* de Daubechies, ganhos mais acentuados para maiores comprimentos dos filtros das *wavelets*, podendo atingir uma aceleração de 1,58 vezes para a *wavelet* de maior comprimento.

Por fim, o estudo é direcionado para as *wavelets* Biortogonais. Esta família é composta por 15 *wavelets* e foram analisadas mediante suas respectivas ordens de reconstrução. Primeiramente, as *wavelets* de ordem 1, bior1.1, bior1.3 e bior1.5, são analisadas e o tempo de processamento comparativo entre método *lifting* e tradicional gasto por estas, pode ser analisado graficamente pela Fig. 9.5.

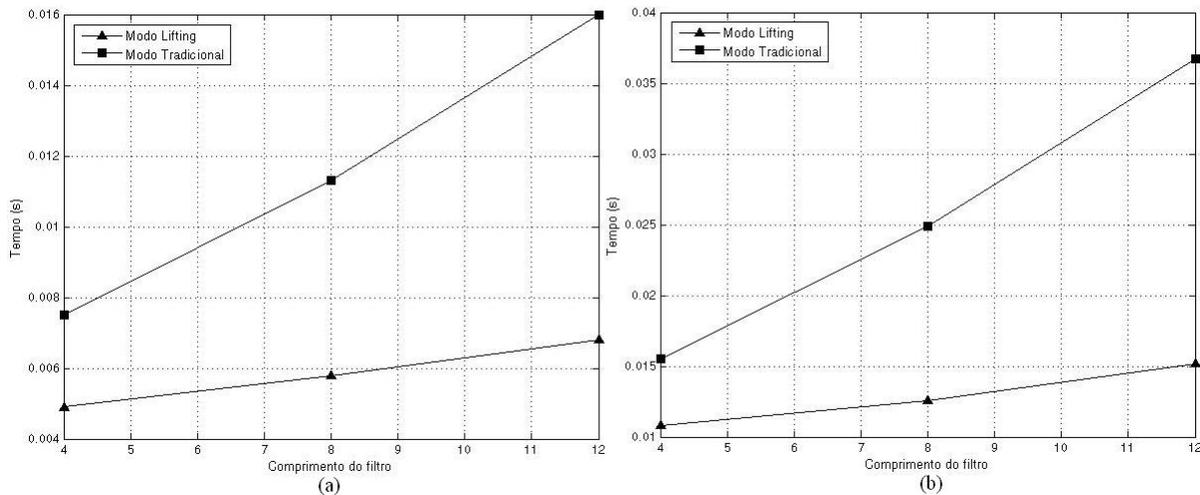


Figura 9.5: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem 1) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Para as *wavelets* bior2.2, bior2.4, bior2.6 e bior2.8, o desempenho do método *lifting* mostrou-se ainda mais significativo, conforme pode ser observado pela Fig. 9.6 e quantitativamente pelas Tabelas 9.5 e 9.6.

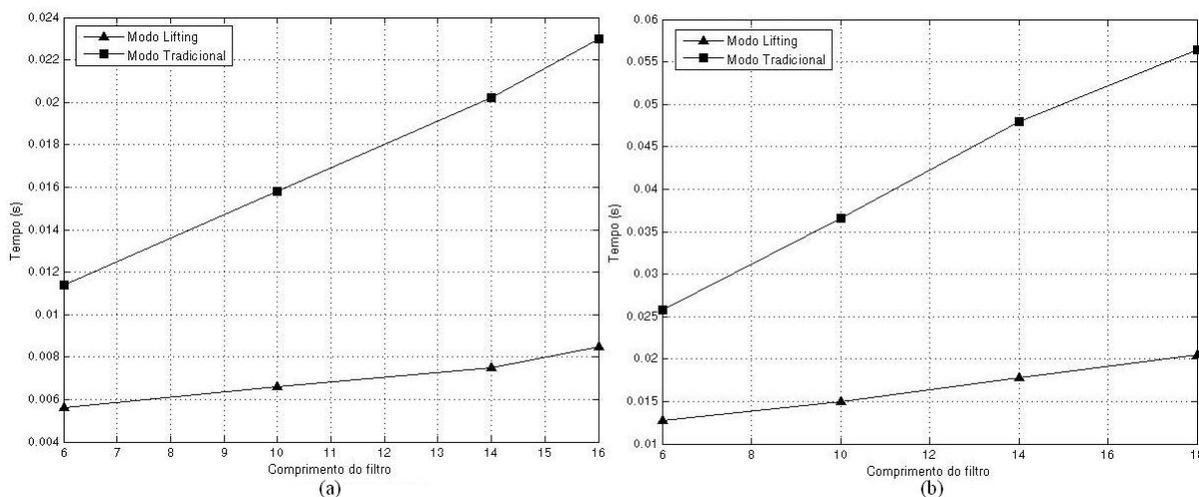


Figura 9.6: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem 2) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

O método *lifting* também apresentou para as *wavelets* bior3.1, bior3.3, bior3.5, bior3.7 e bior3.9 uma boa redução no tempo de processamento se comparado ao método tradicional, tanto para a TWD direta, quanto para a TWD direta e inversa. A Fig. 9.7 ilustra como essa redução é alcançada principalmente para os maiores comprimentos de filtros.

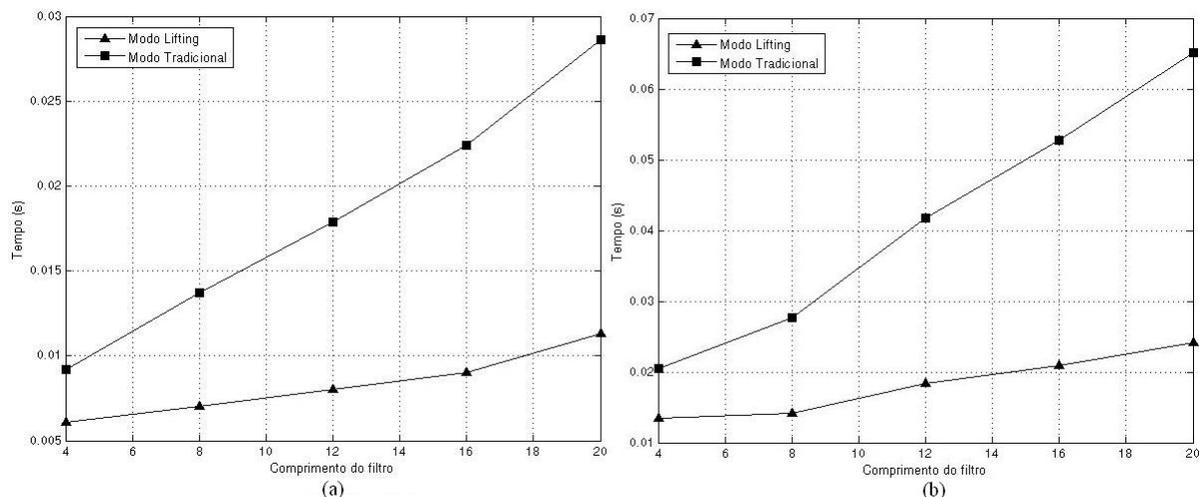


Figura 9.7: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem 3) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Conforme já comentado anteriormente, as *wavelets* bior4.4, bior5.5 e bior 6.8 não pertencem a mesma ordem de reconstrução, mas por apresentarem comprimentos de filtros diferentes, são analisa-

dos pela Fig. 9.8. Para estas, também pode ser percebido que o método tradicional necessita de maior gasto em tempo que o método *lifting* para realizar a TWD.

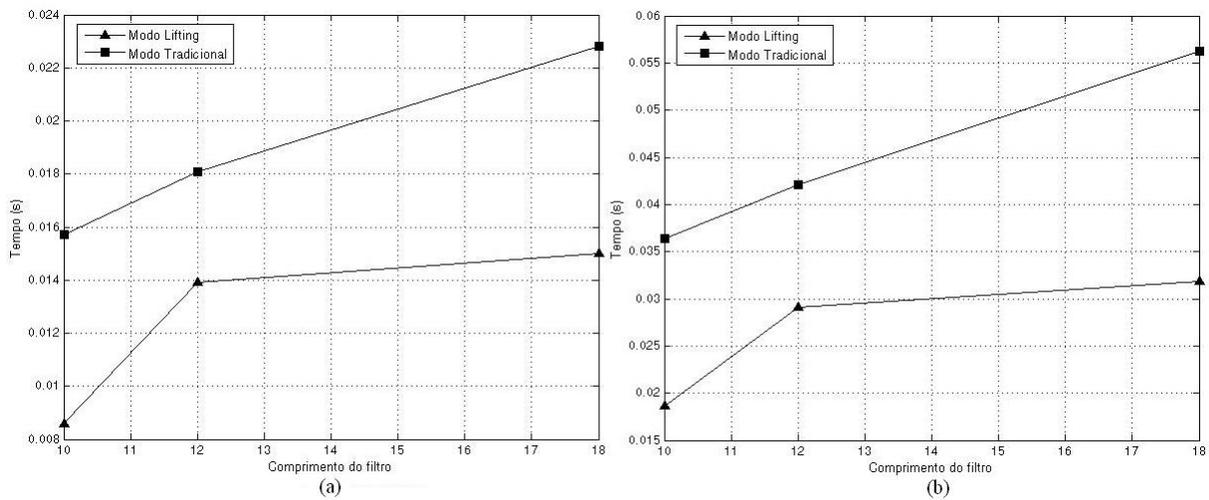


Figura 9.8: Comparação $T_p \times n$ para *wavelets* Biortogonais utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

O comportamento das *wavelets* Biortogonais foi apresentado pelas Figs. 9.5, 9.6, 9.7 e 9.8. As proporções dos ganhos em velocidade na decomposição e no processo completo, respectivamente, encontram-se nas Tabelas 9.5 e 9.6 para cada uma destas *wavelets* Biortogonais.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0075	0,0049	1,53
bior1.3	8	0,0113	0,0058	1,95
bior1.5	12	0,0160	0,0068	2,35
bior2.2	6	0,0114	0,0056	2,04
bior2.4	10	0,0158	0,0066	2,39
bior2.6	14	0,0202	0,0075	2,69
bior2.8	18	0,0230	0,0085	2,71
bior3.1	4	0,0092	0,0061	1,51
bior3.3	8	0,0137	0,0070	1,96
bior3.5	12	0,0179	0,0080	2,24
bior3.7	16	0,0224	0,0090	2,49
bior3.9	20	0,0286	0,0113	2,53
bior4.4	10	0,0157	0,0086	1,83
bior5.5	12	0,0181	0,0139	1,30
bior6.8	18	0,0228	0,0150	1,52

Tabela 9.5: Ferramenta compilada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* Biortogonais.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0155	0,0108	1,44
bior1.3	8	0,0249	0,0126	1,98
bior1.5	12	0,0367	0,0152	2,41
bior2.2	6	0,0258	0,0127	2,03
bior2.4	10	0,0366	0,0150	2,44
bior2.6	14	0,0480	0,0178	2,70
bior2.8	18	0,0564	0,0205	2,75
bior3.1	4	0,0205	0,0134	1,53
bior3.3	8	0,0277	0,0141	1,96
bior3.5	12	0,0418	0,0184	2,27
bior3.7	16	0,0527	0,0209	2,52
bior3.9	20	0,0652	0,0241	2,71
bior4.4	10	0,0364	0,0186	1,96
bior5.5	12	0,0421	0,0291	1,45
bior6.8	18	0,0562	0,0318	1,77

Tabela 9.6: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* Biortogonais.

Para todas as *wavelets* testadas, foi percebido que o método *lifting* realizou a TWD fazendo uso de tempo de processamento consideravelmente menor do que o método tradicional atingindo, respectiva-

mente, os ganhos médios de 2,07 para decomposição e de 2,13 para o processo completo. A razão de velocidade entre os métodos novamente aumentou a medida em que se aumentava o comprimento do filtro da *wavelet* em questão. Para as *wavelets* Biortogonais, o *lifting* mostrou-se ainda mais eficiente que para as demais famílias testadas, atingindo uma razão de velocidade de até 2,75 vezes.

Imagens 512x512

As imagens 512x512 ilustradas na Fig. 9.2 passam a ser utilizadas como imagens-teste para as mesmas comparações a que foram submetidas as imagens 256x256.

Iniciando-se pelas *wavelets* de Daubechies, a Fig. 9.9 apresenta as comparações de desempenho entre os métodos tradicional e *lifting*, apresentando em (a) o tempo necessário para a decomposição e em (b) o tempo necessário para a decomposição e reconstrução, considerando os valores médios de tempo de processamento obtido para o conjunto de imagens 512x512 em questão. Pode ser percebido, mais uma vez, no quanto a utilização do método *lifting* implica em redução significativa no tempo de processamento para a realização da TWD.

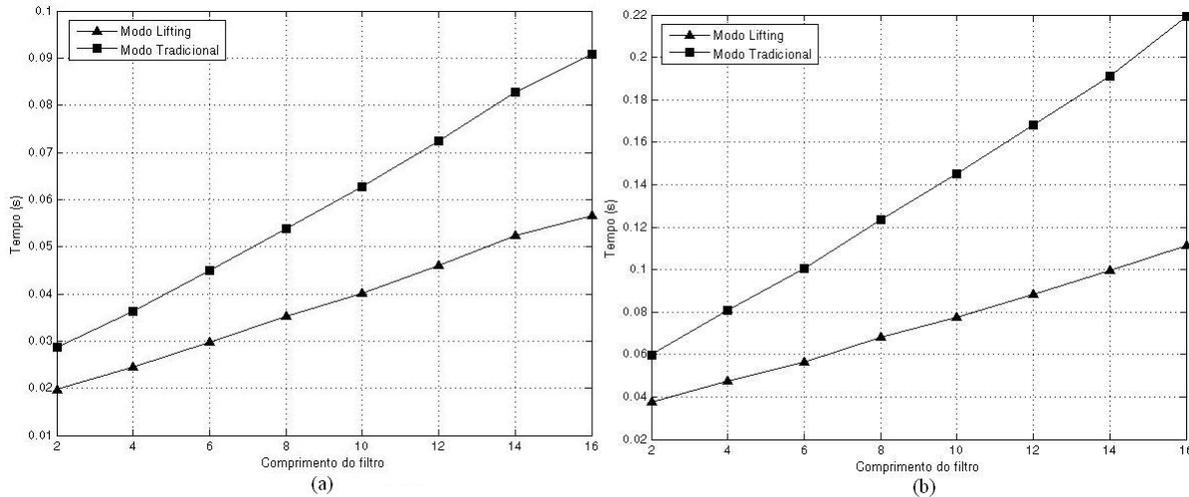


Figura 9.9: Comparação $T_p \times n$ para *wavelets* de Daubechies utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Buscando complementar a Fig. 9.9, as Tabelas 9.7 e 9.8 ilustram, respectivamente, a razão de velocidade para a TWD direta e a razão de velocidade para a TWD direta e inversa para *wavelet* Daubechies. Na decomposição, verifica-se ganho médio de 1,54 vezes, utilizando o método *lifting*. Tal ganho estende-se para 1,82 vezes no processo completo.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
db1	2	0,0287	0,0197	1,46
db2	4	0,0364	0,0245	1,48
db3	6	0,0449	0,0297	1,51
db4	8	0,0539	0,0352	1,53
db5	10	0,0627	0,0402	1,56
db6	12	0,0725	0,0461	1,57
db7	14	0,0828	0,0523	1,58
db8	16	0,0909	0,0566	1,61

Tabela 9.7: Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* Daubechies.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
db1	2	0,0599	0,0375	1,60
db2	4	0,0806	0,0474	1,70
db3	6	0,1004	0,0562	1,79
db4	8	0,1235	0,0680	1,82
db5	10	0,1451	0,0777	1,87
db6	12	0,1679	0,0884	1,90
db7	14	0,1913	0,0996	1,92
db8	16	0,2195	0,1112	1,97

Tabela 9.8: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* Daubechies.

De forma análoga às análises para as imagens 256x256, o ganho continua aumentando na medida em que se aumenta o comprimento do filtro das *wavelets*, atingindo desempenho de até 1,97 vezes sobre o método tradicional.

O desempenho entre os métodos, para as *wavelets* pertencentes à família de Symlets, mostra mais uma vez o método *lifting* apresentando menor tempo de processamento, conforme visto pela Fig. 9.10.

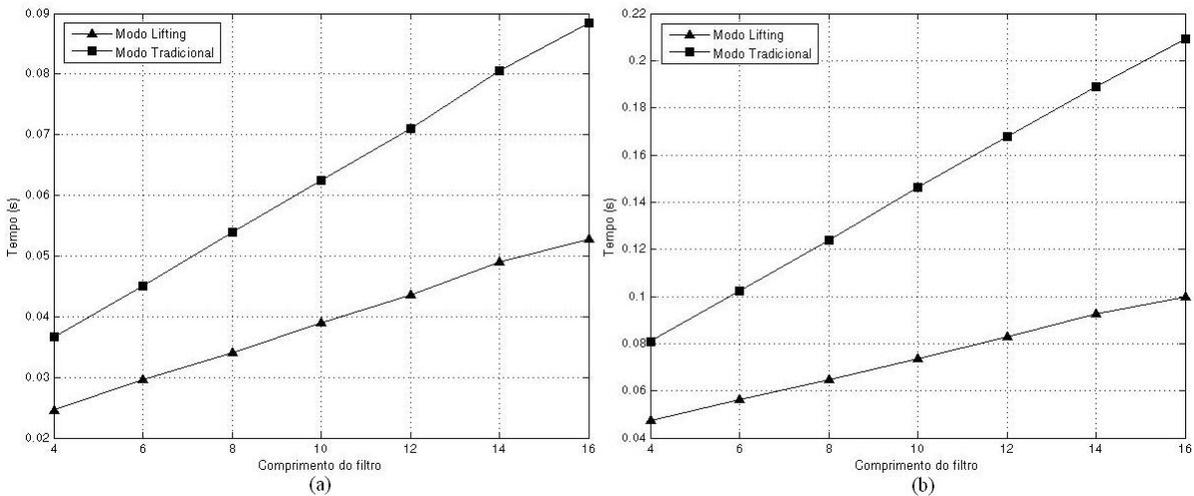


Figura 9.10: Comparação $T_p \times n$ para *wavelets* Symlets utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

A Tabela 9.9, por sua vez, ilustra a razão de velocidade entre os métodos obtida para a decomposição, atingindo ganho médio de 1,59 vezes por parte do método *lifting*. Na TWD direta e inversa, esta razão torna-se ainda maior, onde o método *lifting* realiza o processo em praticamente metade do tempo gasto pelo método tradicional (ganho médio de 1,94 vezes), conforme visto pela Tabela 9.10. Novamente, ganhos maiores são observados com o aumento do comprimento do filtro, atingindo até 2,10 vezes.

Wavelets	n	Tradicional (seg.)	Lifting (seg.)	Razão de velocidade (vezes)
sym2	4	0,0366	0,0246	1,49
sym3	6	0,0451	0,0296	1,52
sym4	8	0,0539	0,0341	1,58
sym5	10	0,0624	0,0390	1,60
sym6	12	0,0710	0,0435	1,63
sym7	14	0,0806	0,0490	1,65
sym8	16	0,0885	0,0527	1,68

Tabela 9.9: Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* Symlets.

Wavelets	n	Tradicional (seg.)	Lifting (seg.)	Razão de velocidade (vezes)
sym2	4	0,0809	0,0473	1,71
sym3	6	0,1022	0,0564	1,81
sym4	8	0,1238	0,0648	1,91
sym5	10	0,1461	0,0735	1,99
sym6	12	0,1678	0,0830	2,02
sym7	14	0,1889	0,0925	2,04
sym8	16	0,2094	0,0998	2,10

Tabela 9.10: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* Symlets.

Finaliza-se as análises para as imagens 512x512, quando *wavelets* Biortogonais são utilizadas para a realização da TWD. Iniciando-se com as *wavelets* bior1.1, bior1.3 e bior1.5, os desempenhos dos métodos são caracterizados pela Fig. 9.11.

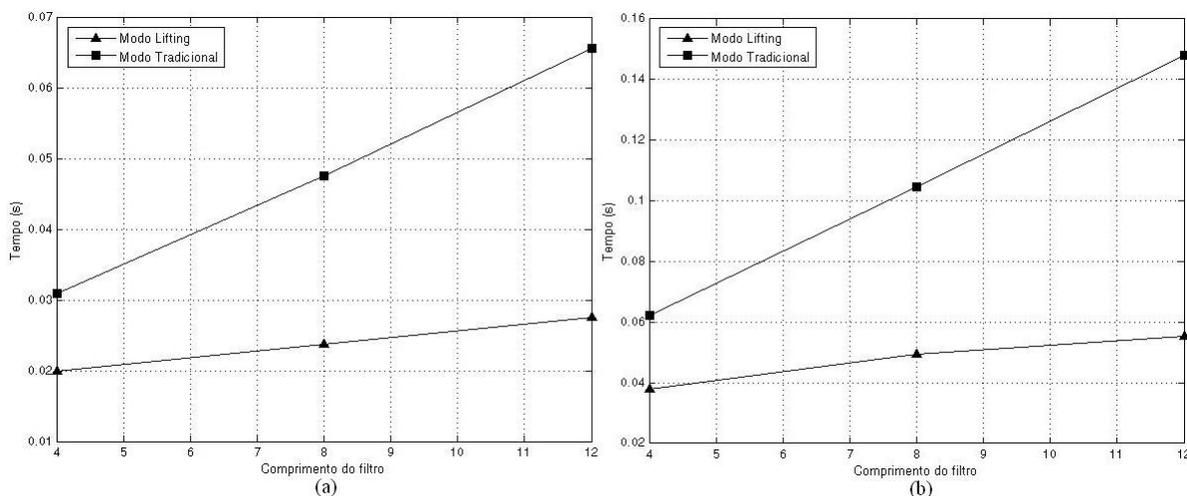


Figura 9.11: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem1) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Novamente esta família de *wavelets* apresenta ganhos mais significativos que as *wavelets* de Daubechies e Symlets. Principalmente para as *wavelets* bior2.2, bior2.4, bior2.6 e bior2.8, a razão de velocidade entre os métodos apresentam os maiores valores, conforme visto graficamente pela Fig. 9.12.

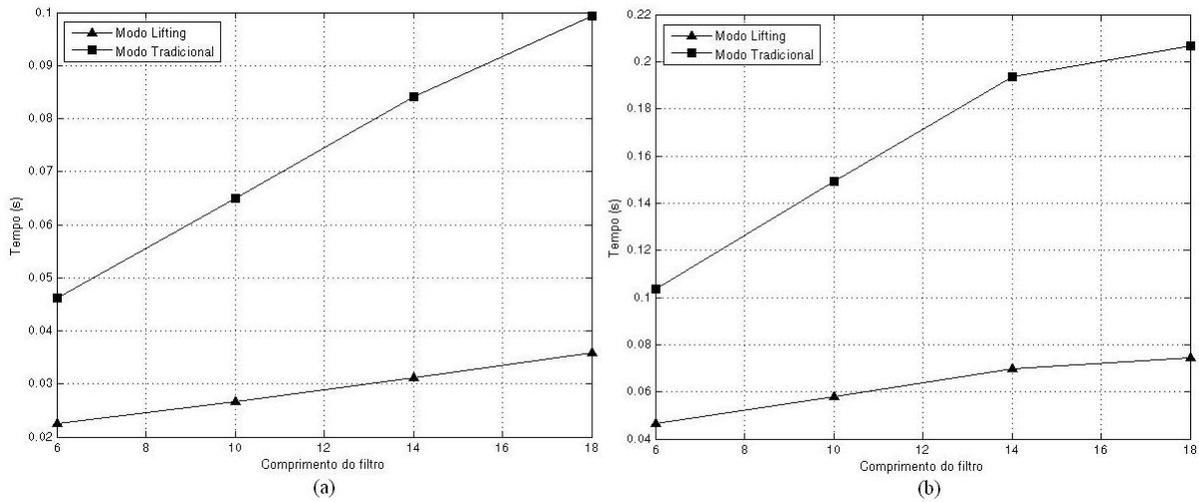


Figura 9.12: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem2) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Nas *wavelets* *bior3.1*, *bior3.5*, *bior3.7* e *bior3.9* o método *lifting* também realiza a TWD com um tempo de processamento bem inferior ao método tradicional. A Fig. 9.13 caracteriza graficamente o desempenho de ambos métodos.

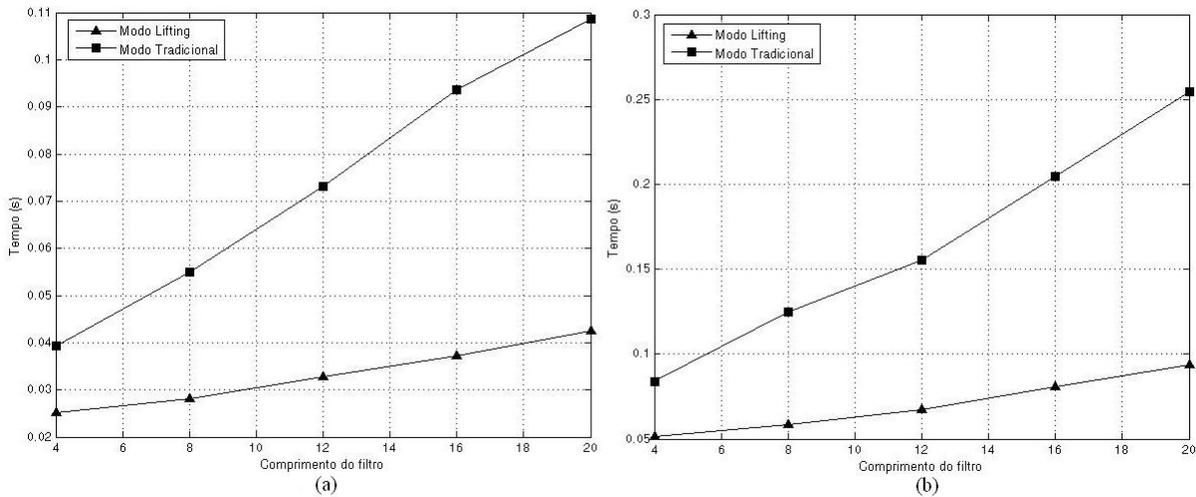


Figura 9.13: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem3) utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

A análise para as imagens 512x512 utilizando a ferramenta compilada é finalizada com a utilização das *wavelets* *bior4.4*, *bior5.5* e *bior6.8*. Como nas demais *wavelets* desta família, a Fig. 9.14 mostra o método *lifting* realizando a TWD com um tempo inferior ao método tradicional.

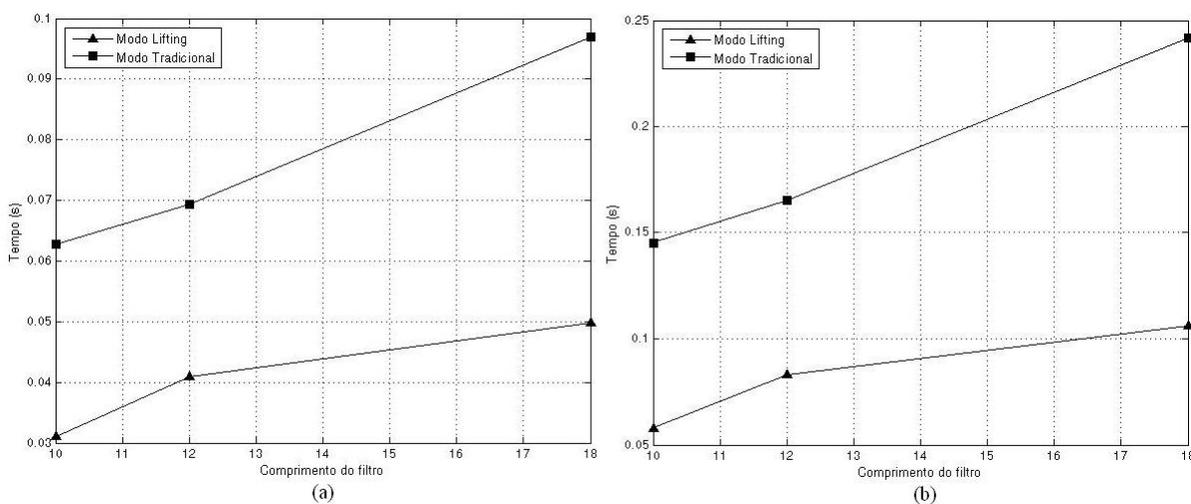


Figura 9.14: Comparação $T_p \times n$ para *wavelets* Biortogonais utilizando ferramenta compilada: (a) Decomposição; (b) Decomposição e reconstrução.

Pelas análises gráficas, foi percebido o quanto o desempenho do método *lifting* mostrou-se superior ao método tradicional para *wavelets* Biortogonais. Complementando, nas Tabelas 9.11 e 9.12 evidenciam numericamente esta razão de velocidade para a TWD direta e para a TWD direta e inversa, respectivamente. Para decomposição, o ganho médio obtido foi de 2,16 vezes e para o processo completo foi de 2,32 vezes.

<i>Wavelets</i>	n	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0309	0,0199	1,56
bior1.3	8	0,0476	0,0237	2,01
bior1.5	12	0,0655	0,0276	2,37
bior2.2	6	0,0462	0,0225	2,05
bior2.4	10	0,0649	0,0267	2,43
bior2.6	14	0,0842	0,0311	2,71
bior2.8	18	0,0994	0,0359	2,76
bior3.1	4	0,0393	0,0252	1,56
bior3.3	8	0,0550	0,0281	1,96
bior3.5	12	0,0732	0,0328	2,23
bior3.7	16	0,0937	0,03716	2,52
bior3.9	20	0,1087	0,0424	2,57
bior4.4	10	0,0627	0,0310	2,03
bior5.5	12	0,0694	0,0410	1,69
bior6.8	18	0,0970	0,0498	1,95

Tabela 9.11: Ferramenta compilada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* Biortogonais.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0622	0,0380	1,63
bior1.3	8	0,1047	0,0493	2,12
bior1.5	12	0,1480	0,0552	2,68
bior2.2	6	0,1034	0,0464	2,23
bior2.4	10	0,1491	0,0581	2,57
bior2.6	14	0,1903	0,0697	2,73
bior2.8	18	0,2065	0,0746	2,77
bior3.1	4	0,0839	0,0515	1,63
bior3.3	8	0,1247	0,0588	2,12
bior3.5	12	0,1555	0,0676	2,30
bior3.7	16	0,2048	0,0806	2,54
bior3.9	20	0,2544	0,0938	2,71
bior4.4	10	0,1453	0,0579	2,51
bior5.5	12	0,1652	0,0830	1,99
bior6.8	18	0,2417	0,1063	2,27

Tabela 9.12: Ferramenta compilada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* Biortogonais.

Percebe-se que para todas as *wavelets* Biortogonais analisadas, o ganho do método *lifting* tornou-se crescente na medida em que o comprimento do filtro era aumentado. Dentre deste contexto, o método *lifting* obteve melhor desempenho para as *wavelets* Biortogonais podendo apresentar, para algumas destas *wavelets*, um desempenho próximo de 3 vezes (2,77).

9.2.2 Implementação via Ferramenta Interpretada

As simulações comparativas objetivas entre a realização da TWD pelo método *lifting* e pelo método tradicional são apresentadas nesta subseção, utilizando a ferramenta interpretada Matlab 7.1. Analogamente ao caso da ferramenta compilada, é avaliado o tempo de processamento necessário para decomposição e o tempo de processamento necessário para todo o processo (decomposição e reconstrução) em ambos os métodos (tradicional e *lifting*) em função do comprimento do filtro (n) para *wavelets* Daubechies, Symlets e Biortogonais.

Imagens 256x256

Iniciando com o emprego da família de *wavelets* de Daubechies, confirma-se que, para as *wavelets* de ordem 1 até ordem 8, a redução no tempo de processamento foi alcançada utilizando o método *lifting*, conforme apresenta a Fig. 9.15. É mostrado na Fig. 9.15 (a) o tempo que cada método necessita para executar apenas a decomposição do sinal e, na Fig. 9.15 (b) o tempo necessário para realizar a

decomposição e reconstrução do sinal (processo completo), considerando os valores médios de tempo de processamento obtido para o conjunto de imagens 256x256 (Fig. 9.1).

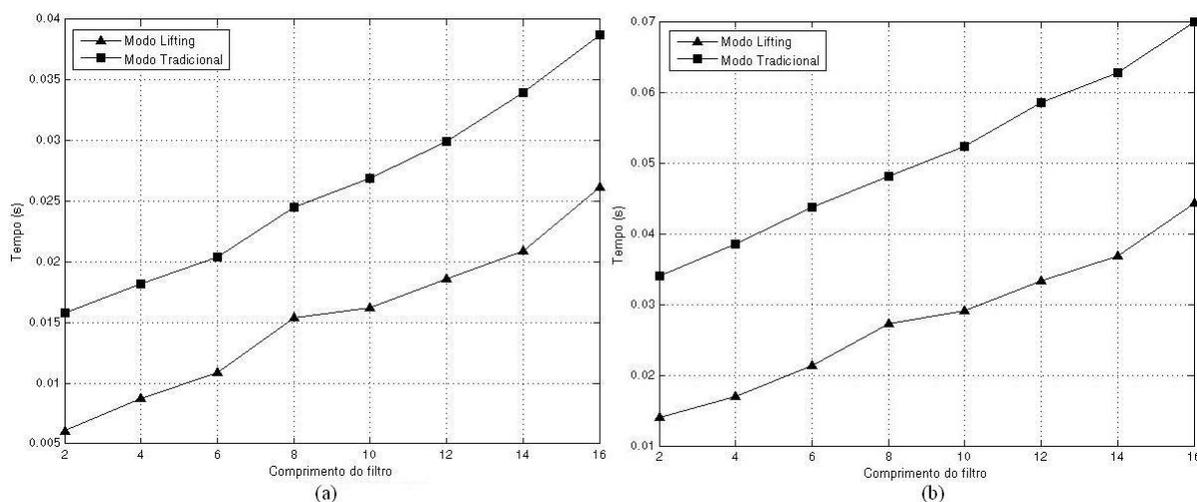


Figura 9.15: Comparação $T_p \times n$ para *wavelets* de Daubechies utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Para a decomposição, a proporção desta razão de velocidade, para cada *wavelet*, pode ser percebida pela Tabela 9.13, atingindo ganho médio de 1,81 vezes aplicando o método *lifting*. A razão em velocidade alcançada para a realização da TWD direta e inversa é mostrada na Tabela 9.14, atingindo 1,92 vezes de ganho sobre o método tradicional.

<i>Wavelets</i>	n	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
db1	2	0,0158	0,0061	2,59
db2	4	0,0182	0,0087	2,09
db3	6	0,0204	0,0109	1,87
db4	8	0,0245	0,0154	1,59
db5	10	0,0269	0,0162	1,66
db6	12	0,0299	0,0186	1,60
db7	14	0,0339	0,0209	1,62
db8	16	0,0387	0,0261	1,48

Tabela 9.13: Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* de Daubechies.

Wavelets	n	Tradicional (seg.)	Lifting (seg.)	Razão de velocidade (vezes)
db1	2	0,0340	0,0139	2,45
db2	4	0,0385	0,0170	2,26
db3	6	0,0437	0,0214	2,04
db4	8	0,0481	0,0273	1,76
db5	10	0,0523	0,0291	1,80
db6	12	0,0585	0,0334	1,75
db7	14	0,0628	0,0369	1,70
db8	16	0,0699	0,0444	1,57

Tabela 9.14: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* de Daubechies.

De acordo com as Tabelas 9.13 e 9.14, percebe-se que o método *lifting* ainda proporciona ganho sobre o método tradicional. Entretanto, contrariamente ao esperado teoricamente e contrariamente aos resultados obtidos através da ferramenta compilada. Na Seção 9.4 e no Capítulo 10 são apresentadas análises e considerações para tais resultados.

A comparação em velocidade obtida entre os métodos *lifting* e tradicional para *wavelets* de Symlets para a TWD direta e para a TWD direta e inversa podem ser vistos, respectivamente, na Fig. 9.16 (a) e 9.16 (b), para as mesmas imagens 256x256 testadas.

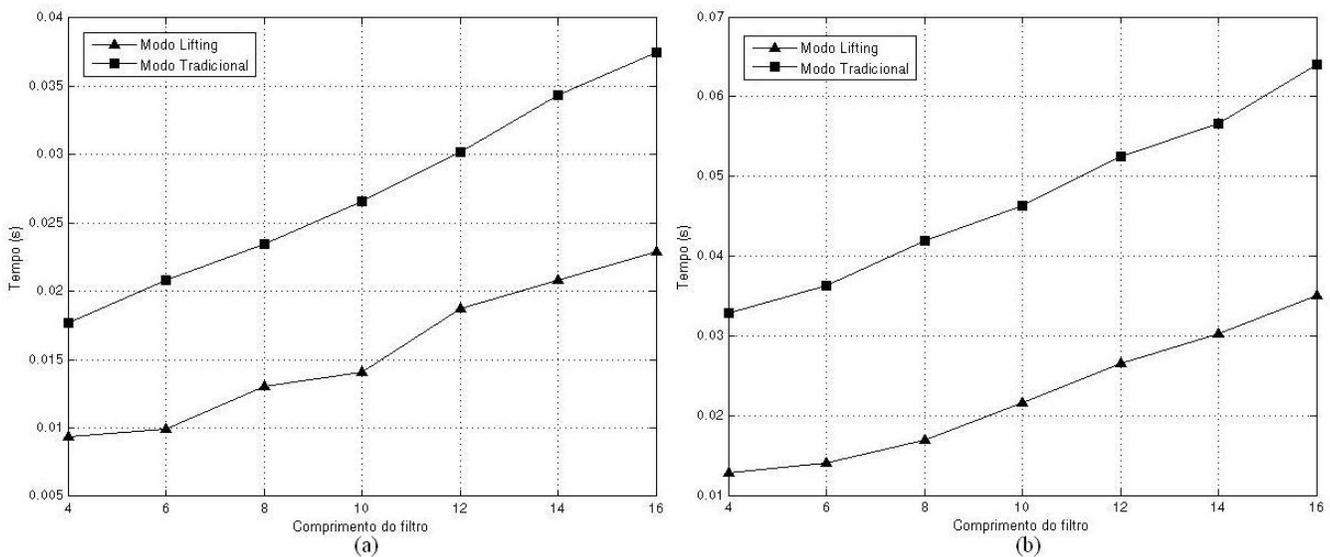


Figura 9.16: Comparação $T_p \times n$ para *wavelets* Symlets utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Com a Tabela 9.15 torna-se possível quantificar mais claramente o ganho proporcionado pelo mé-

todo *lifting* para execução da TWD direta apresentando ganho médio de 1,81 vezes. Os tempos gastos para decompor e reconstruir a imagem-teste colocados na Tabela 9.16, mostram que o desempenho do método *lifting* ainda é superior, atingindo 2,20 vezes. Entretanto, novamente contrário ao esperado teoricamente e contrariamente aos resultados via ferramenta compilada, conforme o comprimento do filtro torna-se maior, novamente o ganho torna-se menor.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,0172	0,0084	2,05
sym3	6	0,0205	0,0102	2,01
sym4	8	0,0233	0,0124	1,87
sym5	10	0,0269	0,0139	1,93
sym6	12	0,0295	0,0185	1,59
sym7	14	0,0333	0,0207	1,61
sym8	16	0,0379	0,0236	1,61

Tabela 9.15: Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* Symlets

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,0328	0,0128	2,56
sym3	6	0,0362	0,0141	2,57
sym4	8	0,0419	0,0169	2,48
sym5	10	0,0462	0,0216	2,14
sym6	12	0,0525	0,0266	1,97
sym7	14	0,0566	0,0303	1,87
sym8	16	0,0641	0,0350	1,83

Tabela 9.16: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* Symlets

Finalmente, cabe a análise para *wavelets* Biortogonais. As análises para esta famílias são realizadas de modo análogo à ferramenta compilada, sendo analisadas de acordo com a ordem de reconstrução das *wavelets*. Iniciando com as de ordem 1, as *wavelets* bior1.1, bior1.3 e bior1.5 são mostradas na Fig. 9.17.

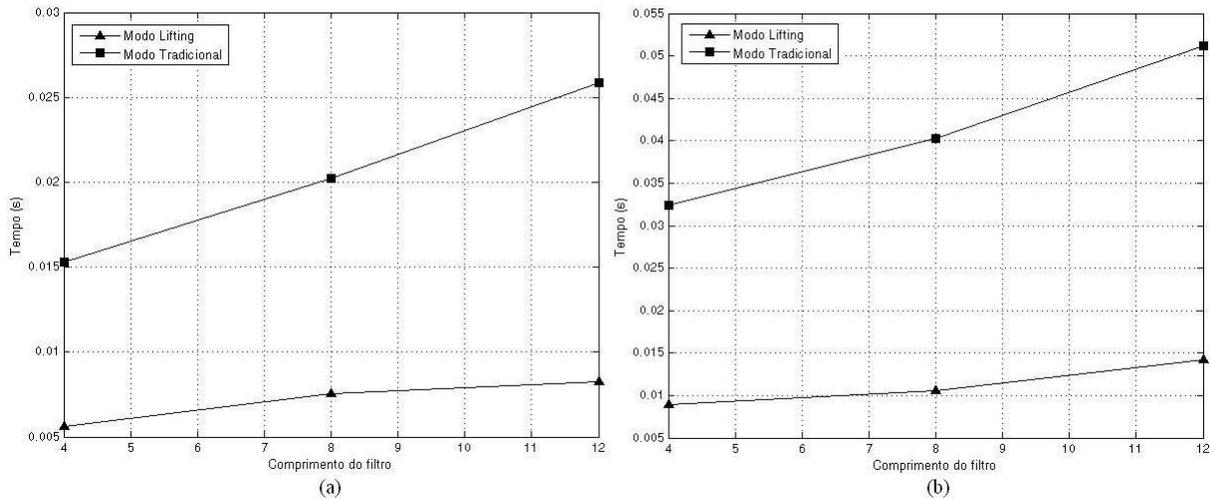


Figura 9.17: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem1) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

O mesmo foi realizado para as *wavelets* bior2.2, bior2.4, bior2.6 e bior2.8. A Fig. 9.18 mostra a proporção do desempenho no aumento da velocidade de processamento utilizando o método *lifting*. Para estas *wavelets* percebe-se que o ganho aumenta conforme o comprimento da *wavelet* torna-se maior, estando de acordo as expectativas teóricas e com toda a análise segundo ferramenta compilada.

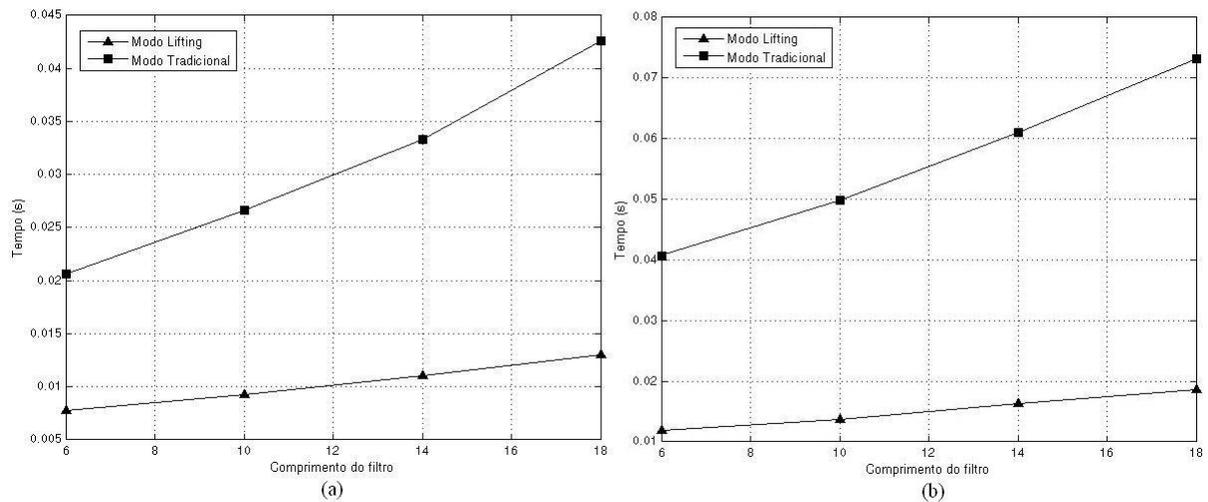


Figura 9.18: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem2) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

O desempenho dos métodos quando utilizado as *wavelets* bior3.1, bior3.5, bior3.7 e bior3.9, é representado pela Fig. 9.19. Nota-se que para estas *wavelets*, a razão de velocidade torna-se maior para os maiores comprimentos de filtros.

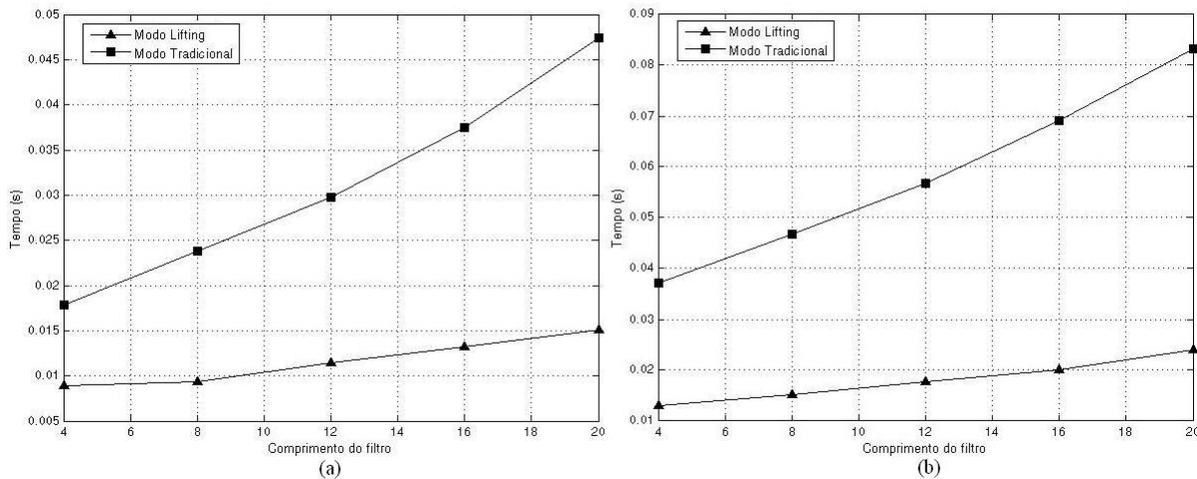


Figura 9.19: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Por fim, as *wavelets* bior4.4, bior5.5 e bior6.8, embora não pertençam a famílias de mesma ordem, apresentam comprimentos de filtros distintos e, deste modo o desempenho proporcionado por estas, na realização da TWD, é caracterizado pela Fig. 9.20.

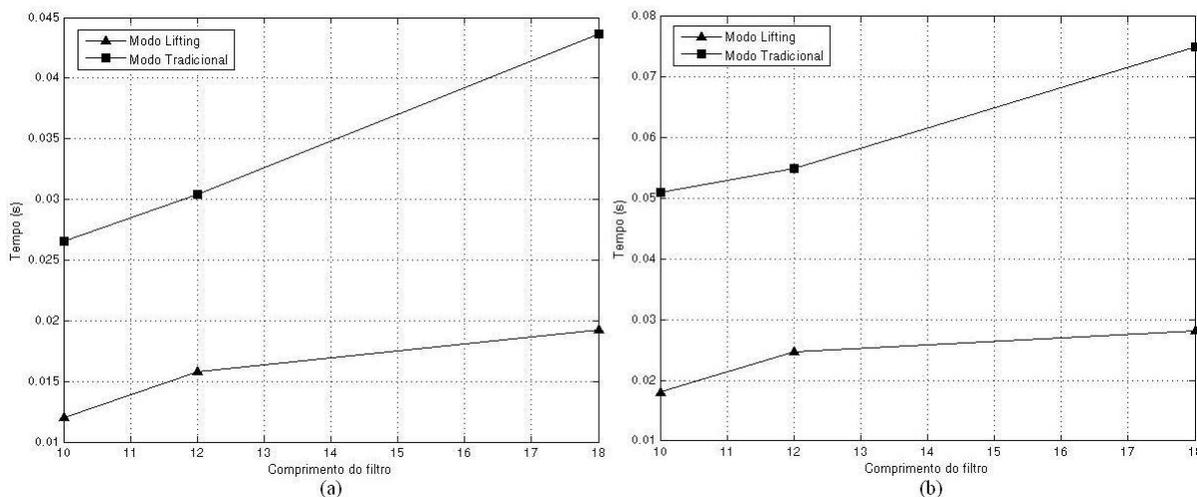


Figura 9.20: Comparação $T_p \times n$ para *wavelets* Biortogonais utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Cabe às Tabelas 9.17 e 9.19, respectivamente, ilustrarem quantitativamente em que proporção se dá o ganho em desempenho oferecido pelo método *lifting* para *wavelets* Biortogonais, no caso da decomposição e posteriormente, para a decomposição e reconstrução.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0153	0,0056	2,73
bior1.3	8	0,0202	0,0076	2,66
bior1.5	12	0,0259	0,0083	3,12
bior2.2	6	0,0207	0,0078	2,65
bior2.4	10	0,0266	0,0093	2,86
bior2.6	14	0,0333	0,0111	3,00
bior2.8	18	0,0426	0,0130	3,28
bior3.1	4	0,0178	0,0089	2,00
bior3.3	8	0,0238	0,0093	2,56
bior3.5	12	0,0298	0,0114	2,61
bior3.7	16	0,0375	0,0132	2,84
bior3.9	20	0,0474	0,0150	3,16
bior4.4	10	0,0266	0,0120	2,22
bior5.5	12	0,0304	0,0158	1,92
bior6.8	18	0,0436	0,0193	2,26

Tabela 9.17: Ferramenta interpretada: Ganho obtido na decomposição de imagens 256x256 utilizando *wavelets* Biortogonais.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0324	0,0090	3,60
bior1.3	8	0,0403	0,0106	3,80
bior1.5	12	0,0512	0,0142	3,61
bior2.2	6	0,0407	0,0119	3,42
bior2.4	10	0,0498	0,0137	3,64
bior2.6	14	0,0611	0,0163	3,75
bior2.8	18	0,0752	0,0187	4,02
bior3.1	4	0,0371	0,0131	2,83
bior3.3	8	0,0468	0,0151	3,10
bior3.5	12	0,0568	0,0177	3,21
bior3.7	16	0,0691	0,0200	3,46
bior3.9	20	0,0832	0,0240	3,47
bior4.4	10	0,0510	0,0180	2,83
bior5.5	12	0,0550	0,0248	2,22
bior6.8	18	0,0750	0,0281	2,67

Tabela 9.19: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 256x256 utilizando *wavelets* Biortogonais.

Com exceção das *wavelets* Biortogonais de ordem 2 e 3, o ganho não se apresentou crescente

conforme aumentava-se o comprimento do filtro da *wavelet*. Para as *wavelets* Biortogonais que obtiveram ganho significativo, pode ser percebido que para decomposição e reconstrução, o método *lifting* mostrou um desempenho de aproximadamente 4 vezes sobre o método tradicional. Para a ferramenta interpretada, contudo, para as *wavelets* Biortogonais, o *lifting* também mostrou-se mais eficiente que para as demais famílias testadas.

Imagens 512x512

A Fig. 9.21 mostra o desempenho para a TWD direta e inversa entre os métodos *lifting* e tradicional para *wavelets* de Daubechies considerando em (a) o tempo necessário para decomposição e em (b) o tempo necessário para o processo completo, considerando valores médios obtidos para o conjunto de imagens 512x512 da Fig. 9.2

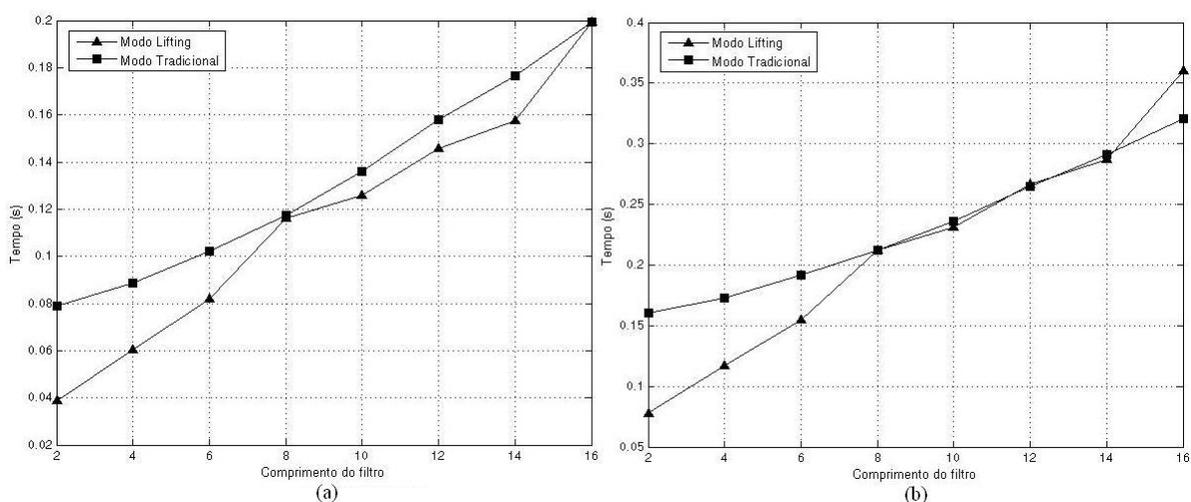


Figura 9.21: Comparação $T_p \times n$ para *wavelets* de Daubechies utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e Reconstrução.

A Tabela 9.20 ilustra o ganho obtido na decomposição para as *wavelets* de Daubechies. O ganho é significativo apenas para baixos comprimentos dos filtros, já que para comprimentos mais altos, os métodos praticamente se equivalem sob o ponto de vista de tempo de processamento, contrariamente ao esperado e de forma análoga ao caso 256x256. O ganho médio foi de 1,25 vezes. A Tabela 9.21, por sua vez, ilustra o ganho obtido para o caso de decomposição e reconstrução do sinal com ganho médio de 1,21 vezes.

Wavelets	n	Tradicional (seg.)	Lifting (seg.)	Razão de velocidade (vezes)
db1	2	0,0789	0,0387	2,03
db2	4	0,0885	0,0604	1,47
db3	6	0,1022	0,0819	1,25
db4	8	0,1173	0,1161	1,01
db5	10	0,1358	0,1259	1,07
db6	12	0,1764	0,1458	1,08
db7	14	0,1794	0,1574	1,12
db8	16	0,1992	0,1994	1,00

Tabela 9.20: Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* de Daubechies.

Wavelets	n	Tradicional (seg.)	Lifting (seg.)	Razão de velocidade (vezes)
db1	2	0,1604	0,0778	2,06
db2	4	0,1725	0,1168	1,48
db3	6	0,1918	0,1551	1,23
db4	8	0,2121	0,2123	1,00
db5	10	0,2362	0,2308	1,02
db6	12	0,2650	0,2662	1,00
db7	14	0,2909	0,2871	1,01
db8	16	0,3207	0,3602	0,89

Tabela 9.21: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* de Daubechies.

A Fig. 9.22 ilustra a análise para *wavelets* Symlets.

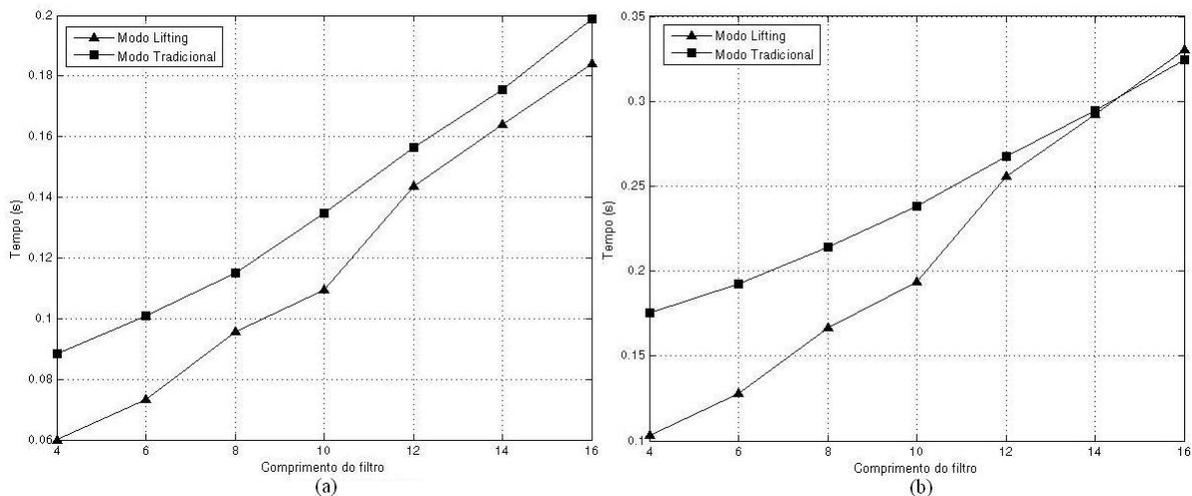


Figura 9.22: Comparação $T_p \times n$ para *wavelets* Symlets utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e Reconstrução.

Para estas *wavelets*, na decomposição foi obtido um ganho de aproximadamente 10% para os filtros de maiores comprimentos, conforme visto na Tabela 9.22, com ganhos de 1,22 vezes, contrariamente ao esperado e de acordo com o caso 256x256. Mas o tempo que foi necessário para realizar a decomposição e reconstrução do sinal, implicou em um ganho de 0% para os filtros de maiores comprimentos, conforme visto quantitativamente pela Tabela 9.23 (ganho médio de 1,25 vezes).

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,0883	0,0600	1,47
sym3	6	0,1008	0,0732	1,38
sym4	8	0,1152	0,0955	1,21
sym5	10	0,1348	0,1094	1,23
sym6	12	0,1565	0,1437	1,09
sym7	14	0,1756	0,1641	1,07
sym8	16	0,1989	0,1839	1,08

Tabela 9.22: Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* Symlets.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
sym2	4	0,1753	0,1032	1,69
sym3	6	0,1926	0,1280	1,50
sym4	8	0,2143	0,1667	1,29
sym5	10	0,2381	0,1934	1,23
sym6	12	0,2674	0,2558	1,05
sym7	14	0,2946	0,2922	1,01
sym8	16	0,3245	0,3303	0,98

Tabela 9.23: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* Symlets.

A análise das imagens 512x512 é encerrada com o emprego da família de *wavelets* Biortogonais. A Fig. 9.23 ilustra o desempenho entre métodos para as *wavelets* bior1.1, bior1.3 e bior1.5.

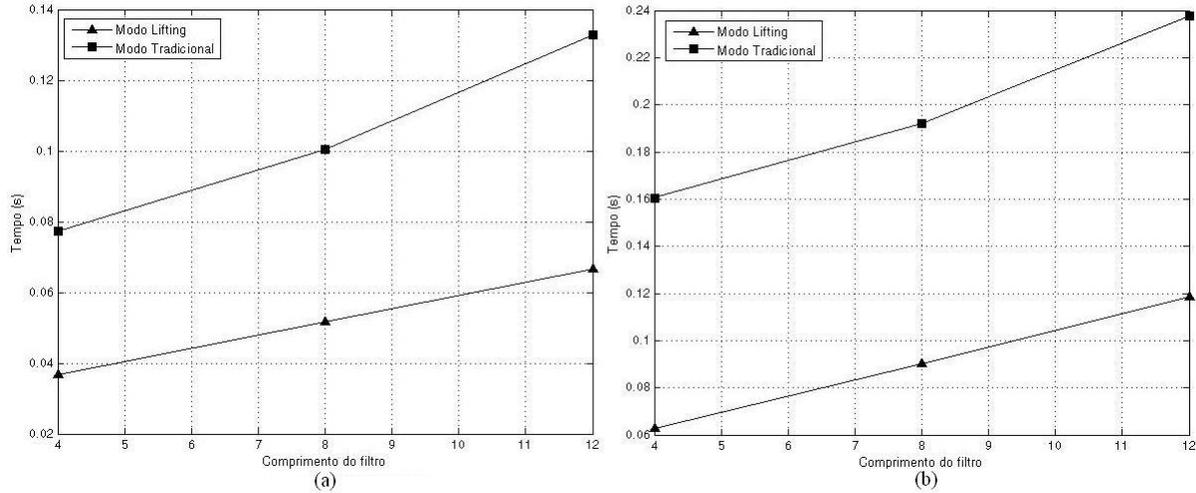


Figura 9.23: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem1) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Para as *wavelets* bior2.2, bior2.4, bior2.6 e bior2.8, novamente foi obtido um bom desempenho para a técnica *lifting* quando comparado ao método tradicional, caracterizado pela Fig. 9.24.

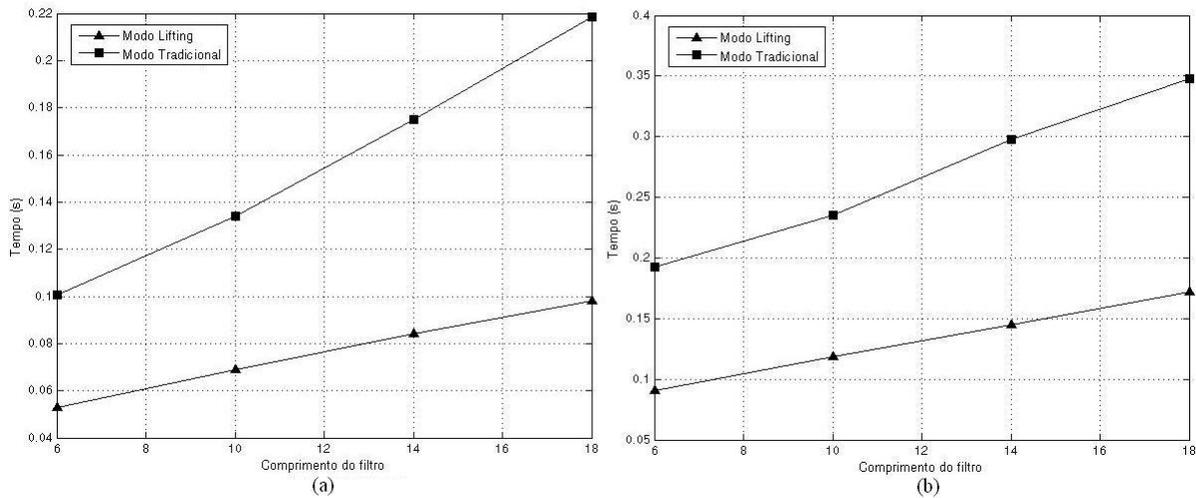


Figura 9.24: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem2) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Para as *wavelets* de ordem 3, o tempo de processamento para o método *lifting*, na execução da TWD, também mostrou-se inferior, conforme visto na Fig. 9.25.

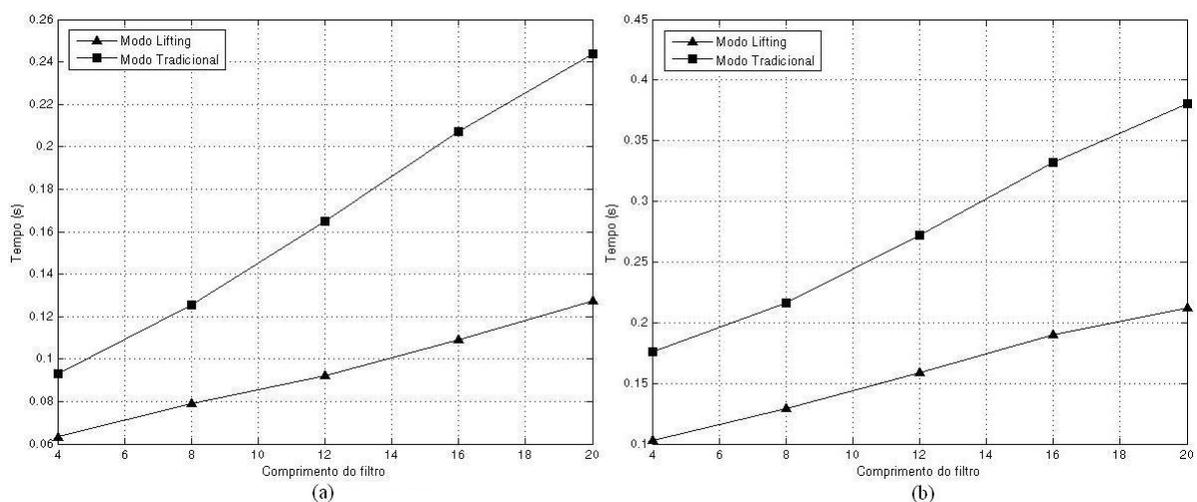


Figura 9.25: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Finalmente, as *wavelets* bior4.4, bior5.5 e bior6.8 foram analisadas e também foi possível deparar com uma melhor performance do método *lifting* sob o método tradicional. A Fig. 9.26 mostra seus respectivos desempenhos:

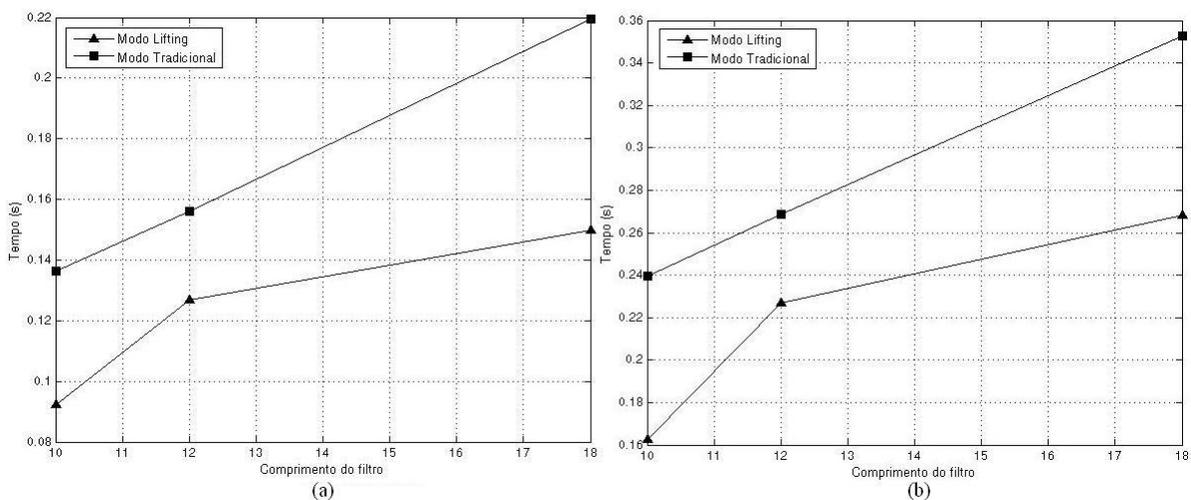


Figura 9.26: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem3) utilizando ferramenta interpretada: (a) Decomposição; (b) Decomposição e reconstrução.

Pode-se perceber, dentre as famílias *wavelets* analisadas, que a família de *wavelets* Biortogonais foi novamente a que obteve melhor desempenho (ganhos médios de 1,80 vezes para a decomposição e 1,83 vezes para o processo completo), conforme mostrado pelas Tabelas 9.24 e 9.25.

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,0773	0,0366	2,11
bior1.3	8	0,1005	0,0517	1,94
bior1.5	12	0,1327	0,0667	1,99
bior2.2	6	0,1005	0,0530	1,90
bior2.4	10	0,1341	0,0688	1,95
bior2.6	14	0,1750	0,0842	2,08
bior2.8	18	0,2186	0,0983	2,22
bior3.1	4	0,0930	0,0633	1,47
bior3.3	8	0,1256	0,0792	1,59
bior3.5	12	0,1650	0,0920	1,79
bior3.7	16	0,2073	0,1091	1,90
bior3.9	20	0,2437	0,1275	1,91
bior4.4	10	0,1364	0,0923	1,48
bior5.5	12	0,1559	0,1267	1,23
bior6.8	18	0,2194	0,1498	1,46

Tabela 9.24: Ferramenta interpretada: Ganho obtido na decomposição de imagens 512x512 utilizando *wavelets* Biortogonais

<i>Wavelets</i>	<i>n</i>	Tradicional (seg.)	<i>Lifting</i> (seg.)	Razão de velocidade (vezes)
bior1.1	4	0,1605	0,0627	2,56
bior1.3	8	0,1922	0,0902	2,13
bior1.5	12	0,2376	0,1185	2,01
bior2.2	6	0,1928	0,0903	2,14
bior2.4	10	0,2356	0,1184	1,99
bior2.6	14	0,2978	0,1452	2,05
bior2.8	18	0,3481	0,1722	2,02
bior3.1	4	0,1759	0,1029	1,71
bior3.3	8	0,2165	0,1293	1,67
bior3.5	12	0,2723	0,1591	1,71
bior3.7	16	0,3323	0,1903	1,75
bior3.9	20	0,3802	0,2121	1,79
bior4.4	10	0,2397	0,1624	1,48
bior5.5	12	0,2685	0,2268	1,18
bior6.8	18	0,3528	0,2684	1,31

Tabela 9.25: Ferramenta interpretada: Ganho obtido na decomposição e reconstrução de imagens 512x512 utilizando *wavelets* Biortogonais

Conforme percebido graficamente e quantitativamente, para as imagens 512x512, as *wavelets* Biortogonais foram as que alcançaram ganho mais significativo, porém este não aumenta com o

comprimento do filtro, contrariando as expectativas teóricas e a ferramenta compilada.

9.3 Reconstrução Perfeita

O método *lifting* realiza a TWD de modo a reduzir consideravelmente sua complexidade computacional, sem introduzir quaisquer distorções objetivas (PSNR) ou subjetivas (artefatos de processamento). Na Seção 9.2, uma análise foi apresentada para verificar o quanto o método *lifting* pode ser mais eficiente que o tradicional em termos de tempo de processamento. Esta seção destina-se portanto, a corroborar a correta implementação das simulações anteriores e seus resultados na medida em que a reconstrução das imagens resultantes das implementações deve ser perfeita.

Neste sentido, pode ser verificado nas Figs. 9.27, 9.28, 9.29, 9.30 e 9.31 que a implementação *lifting* realizada neste trabalho apresentou imagens reconstruídas idênticas as originais, para ambas as ferramentas utilizadas (interpretada e compilada). Através de uma análise subjetiva pôde-se verificar que o sinal reconstruído apresenta um nível excelente corroborado pelo valor de PSNR infinito para as imagens 256x256 e 512x512. Algumas das imagens utilizadas e reconstruções são apresentadas nas Figs. a seguir, a título ilustrativo.

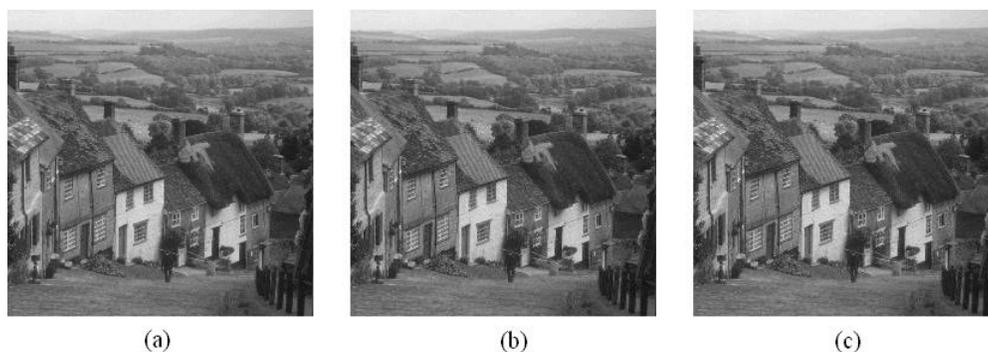


Figura 9.27: Reconstrução obtida pelo método *lifting* (Goldhill - 256x256): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.

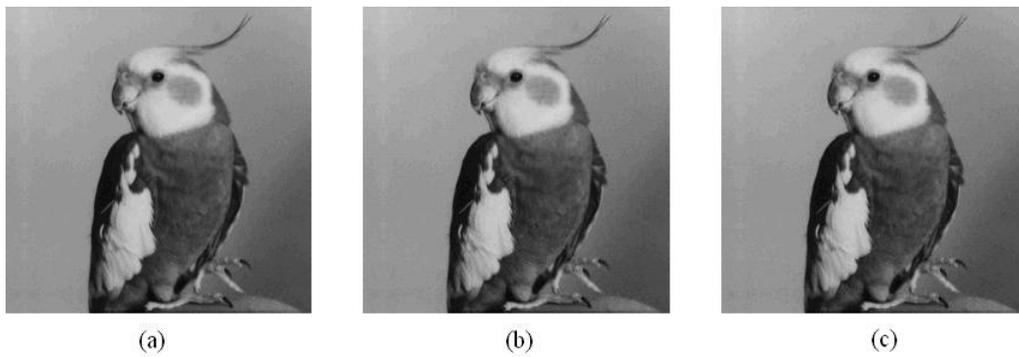


Figura 9.28: Reconstrução obtida pelo método *lifting* (Bird - 256x256): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.



Figura 9.29: Reconstrução obtida pelo método *lifting* (Lena - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.



Figura 9.30: Reconstrução obtida pelo método *lifting* (Boat - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.

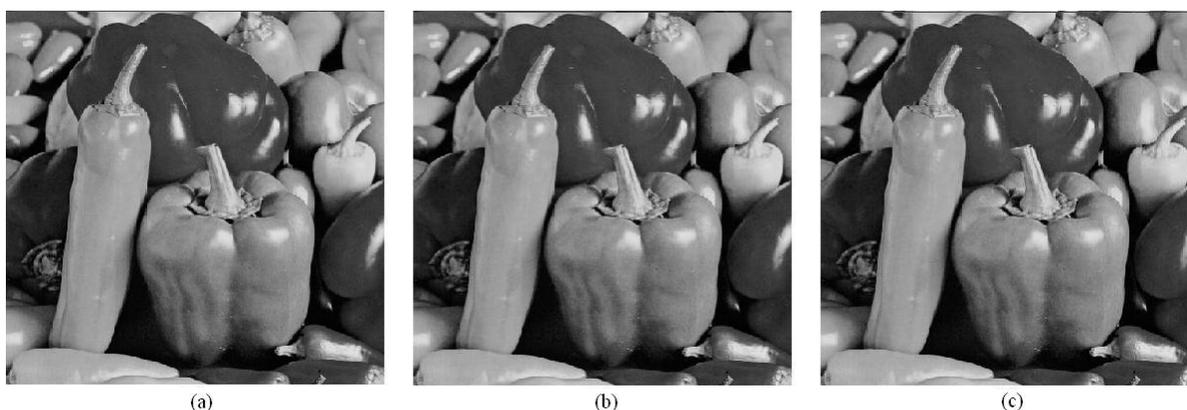


Figura 9.31: Reconstrução obtida pelo método *lifting* (Boat - 512x512): (a) Imagem Original; (b) Ferramenta Compilada; (c) Ferramenta Interpretada.

9.4 Observações sobre Ferramenta Compilada x Ferramenta Interpretada

As premissas estabelecidas por Daubechies [1] e Reichel [2] e esperadas com o estudo da dissertação de que o método *lifting* apresenta ganho crescente em tempo de processamento (T_p), à medida em que se eleva o comprimento do filtro (n), foram plenamente confirmadas pelas simulações utilizando ferramenta compilada. A correta implementação realizada pelo pesquisador foi corroborada pelo PSNR infinito obtido entre imagens originais e reconstruídas (Seção 9.3).

Entretanto, alguns resultados inconsistentes neste aspecto foram obtidos pelas simulações utilizando ferramenta interpretada, chegando a inverter a lógica das referências acima citadas. A seguir, são tecidas considerações e apontamentos a este respeito.

A ferramenta Matlab 7.1 consiste em uma linguagem de programação interpretada de altíssimo nível, compondo uma ferramenta poderosa, em especial para desenvolvimento de algoritmos e computação numérica, permitindo aos pesquisadores de diversas áreas solucionar problemas e implementar sistemas de maneira mais rápida e imediata, se comparado a outras ferramentas de programação.

Entretanto, como linguagem interpretativa, o Matlab em todas as suas versões, no intuito de acelerar as implementações dos usuários, realiza excelentes otimizações nas estruturas de cálculos (somas e multiplicações) a nível de bits, operações e acessos à memória, provocando alterações no tempo de processamento de algoritmos, mesmo entre diferentes versões da própria ferramenta, como pode ser visto exemplificando modestamente na Fig. 9.32 através de um exemplo extremamente simples.

Versão	Tp (seg.)
Matlab 6.1	6,81
Matlab 7.1	3,31

```

function teste(N)
a=[1:1000]/200;
b=[1001:2000]/200;
c=0;
tempo=cputime;
for i=1:N
    c=c+a*b;
end;
tempo=cputime-tempo;
disp(sprintf('Nop=%g, tempo=%f s',N*1000, tempo));

```

Figura 9.32: Tp obtido por diferentes versões do Matlab.

Tais otimizações são frutos da característica interpretativa inerente ao código e são feitas para facilitar a programação de sistema, acelerando a execução dos mesmos.

Logo, os dados comparativos inconsistentes com as referências [1] e [2] obtidos neste trabalho possuem como provável explicação a aceleração obtida por tais otimizações internas ao Matlab.

Indicativo forte de que esta explicação é coerente reside em breves testes realizados pelo pesquisador presentes no Apêndice A, onde os resultados de tempo de processamento obtidos através da ferramenta interpretada indicam dependência das características específicas da imagem a ser processada, fato que não ocorre em linguagens compiladas. Por fugir ao escopo deste trabalho e por consistir em breves testes, tais simulações são apresentadas como Apêndice somente.

Neste contexto, o Matlab constitui-se em uma ferramenta poderosa e excelente. Entretanto, modestamente, observa-se que deve ser usado com ressalvas para procedimentos comparativos de tempo de processamento, em especial quando houver grande quantidade de acessos à memória, como é o caso de filtragens.

9.5 Sumário da Análise de Resultados

Este capítulo concentra os resultados fundamentais da presente dissertação analisando na prática em que proporção o desempenho oferecido pelo método *lifting* é superior ao método tradicional, em termos de tempo de processamento, via ferramenta interpretada e ferramenta compilada, dando seqüência à linha de estudos teóricos apresentados por Daubechies e Sweldens [1] e Reichel [2].

Assim, este capítulo apresentou simulações e análises comparativas com relação ao desempenho em tempo de processamento entre o método *lifting* e o método tradicional para diversas resoluções, imagens-teste e diferentes tipos de *wavelets* (Daubechies, Symlets e Biortogonais). Para todas as imagens analisadas e para as diversas *wavelets* utilizadas, verificou-se em quantas vezes a redução teórica de operações aritméticas pôde influenciar na redução de tempo de processamento, principal foco da pesquisa.

O método *lifting* realiza a sub-amostragem antes da filtragem, procedimento inverso ao método

tradicional, promovendo ganho dada a reordenação da estrutura.

Em [1], afirma-se que conforme o comprimento do filtro é aumentado, o número de operações aritméticas utilizadas pelo método *lifting* tende a reduzir-se pela metade, implicando em um ganho de dois sobre o método tradicional, segundo Daubechies e Sweldens. Reichel, vai além desta afirmação dizendo que este ganho pode chegar a quatro vezes [2].

A Tabela 9.26 sumariza os resultados máximos obtidos neste trabalho para realizar a verificação prática dos trabalhos teóricos de Daubechies e Sweldens e de Reichel, que abordaram ganhos máximos teóricos estimados.

	Média decomposição	Média Completa	MAX
Daubechies	1,54	1,82	1,97
Symlets	1,59	1,94	2,10
Biortogonais	2,16	2,32	2,77

Tabela 9.26: Resultados médios e máximos de tempo de processamento em linguagem compilada para diferentes famílias de *wavelets*.

Com base nos principais resultados sumarizados acima, junto das extensas simulações neste trabalho, verificou-se através de implementações práticas, foco da presente dissertação, o ganho teórico afirmado por Reichel [2], superando realmente as previsões da própria Daubechies e Sweldens [1] que estimavam um ganho máximo de duas vezes. Nesta verificação reside a principal contribuição desta pesquisa.

Entre as ferramentas comparadas, pode-se afirmar que utilizando a ferramenta compilada, os resultados obtidos estão de acordo com a literatura [1]. Para as *wavelets* de Daubechies e Symlets, o ganho que o método *lifting* oferece sobre o método tradicional é de aproximadamente duas vezes para os comprimentos de filtros que apresentam maior tamanho. Para as *wavelets* Biortogonais, ganhos próximos de três foram obtidos para as *wavelets* bior2.4, bior2.6, bior2.8, bior3.7, bior3.9 e bior4.4. As extensas simulações da Seção 9.2.1 também confirmam na prática as afirmações teóricas de Reichel e de Daubechies e Sweldens quanto ao aumento do ganho do método *lifting* com relação ao método tradicional à medida em que se eleva o comprimento do filtro (n).

Dentre as famílias de *wavelets* testadas, a família Biortogonal sempre apresentou maior ganho em tempo de processamento, corroborando sua excelente aplicação no campo de compressão de imagens, aliando este fator a suas vantajosas características: filtros mais flexíveis (decomposição diferente de reconstrução) e mais fáceis de serem projetados; simetria, reconstrução exata, alta regularidade e bom desempenho para compressão (características apresentadas nos Capítulos 6 e 7). Como contrapartida, a família Biortogonal somente apresentou ligeiro aumento de complexidade de implementação, uma vez que devem ser projetadas quatro estruturas de filtragens diferentes ao invés de duas.

Em segundo plano, as extensas simulações realizadas no presente trabalho permitiram observar interessantes características apontadas nas seções anteriores, tais como as afirmações sobre algoritmos de mais rápida execução estarem vinculados a *wavelets* de suporte compacto e características de suporte compacto das três famílias utilizadas nos experimentos.

Entretanto, alguns resultados inconsistentes neste aspecto foram obtidos pelas simulações utilizando ferramenta interpretada, chegando a inverter a lógica das referências acima citadas. A ferramenta Matlab 7.1 consiste em uma linguagem de programação interpretada de altíssimo nível, compondo uma ferramenta poderosa. Entretanto, como linguagem interpretativa, o Matlab em todas as suas versões, no intuito de acelerar as implementações dos usuários, realiza excelentes otimizações nas estruturas de cálculos (somadas e multiplicações) a nível de bits, operações e acessos à memória, provocando alterações no tempo de processamento de algoritmos, mesmo entre diferentes versões da própria ferramenta, como foi exemplificado através de indícios modestamente na Seção 9.4 e Apêndice A. Logo, os dados comparativos inconsistentes com as referências [1] e [2] obtidos neste trabalho possuem como provável explicação a aceleração obtida por tais otimizações internas ao Matlab. Neste contexto, o Matlab constitui-se em uma ferramenta poderosa e excelente. Entretanto, observa-se que deve ser usado com ressalvas para procedimentos comparativos de tempo de processamento, em especial quando houver grande quantidade de acessos à memória, como é o caso de *filtragens*.

As imagens reconstruídas foram analisadas através do PSNR (*Peak Signal-to-Noise Ratio*), medida objetiva utilizada devido à sua extensa utilização tanto em artigos quanto na bibliografia de PDI (Processamento Digital de Imagem - *Image Digital Processing*). Dentro deste contexto, a ferramenta interpretada e a ferramenta compilada obtiveram reconstrução perfeita, assegurado por medida de PSNR infinita, corroborando com a perfeita implementação do *lifting*.

9.6 Comentários

Este capítulo verificou, na prática, através de extensas simulações, análises e conclusões, os trabalhos teóricos de Daubechies e Sweldens e de Reichel acerca do ganho oferecido pelo método *lifting* se comparado ao método tradicional.

Foram realizadas análises para diversas imagens-teste, resoluções, diferentes famílias de *wavelets* (Daubechies, Symlets e Biortogonais) utilizando ferramentas interpretada e compilada.

Capítulo 10

CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

10.1 Comentários sobre o Contexto

Atualmente, o Processamento Digital de Imagens possui uma ampla gama de aplicações que vai desde aplicações médicas, processamento de imagens adquiridas por satélites de sensoriamento remoto, tratamento de imagens geológicas, radares e sonares, teleconferência e, mais recentemente, para HDTV (TV de Alta Definição - *High Definition Television*). A representação digital da informação visual, em sua forma não processada, por característica inerente, gera um número significativo de grandes arquivos.

Embora os métodos de compressão sejam hoje de uso extensivo, o ritmo acelerado no crescimento das aplicações de imagens e vídeos digitais impõem ao mundo tecnológico sua necessidade pelo aumento na capacidade de armazenamento e na velocidade de transmissão. Neste sentido, o trabalho destinou-se em apresentar a proporção do ganho em velocidade que pode ser alcançado para realização da TWD, através do método *lifting*. Como se sabe, métodos *wavelet* são adequados para aplicações que requerem codificação rápida, por exemplo, em comunicações em tempo real via *internet* e, neste sentido, velocidade é um dos parâmetros fundamentais levados em conta na elaboração de projetos.

Na literatura, estudos teóricos foram realizados em busca de obter-se afirmações quanto à proporção do ganho aritmético obtido entre o método *lifting* e o método tradicional. Daubechies e Sweldens propuseram o método *lifting*, que faz uso de um número menor de operações aritméticas chegando-se a atingir, no caso de comprimentos de filtros maiores, metade das operações do método tradicional [1]. Em contraposição, Reichel [2] afirma que este ganho pode ter proporção ainda maior, podendo atingir cerca de quatro vezes o método tradicional.

Embora tais estudos apresentem uma divergência a respeito do ganho proporcionado, a efetividade na aceleração no cálculo dos coeficientes é um fator extremamente atrativo, minimizando a complexidade da TWD e acelerando o tempo de processamento [75] [77]. Adicionalmente, ambos afirmam que este ganho deve tomar proporções maiores na medida em que o comprimento da *wavelet* torna-se maior.

Buscando dar seqüência à linha de estudos teóricos apresentados por Daubechies e Sweldens [1] e por Reichel [2], foi analisado na prática em que proporção o desempenho oferecido pelo método *lifting* é superior ao método tradicional, em termos de tempo de processamento, via ferramenta interpretada e compilada, apresentando simulações e análises comparativas para diversas imagens-teste, resoluções e famílias de *wavelets* (Daubechies, Symlets e Biortogonais).

10.2 Breves Conclusões sobre os Capítulos e Relação com a Proposta

Para a realização do presente trabalho, foi dado maior ênfase à teoria *wavelet*, base para a proposta. Os capítulos foram, basicamente, divididos em quatro principais abordagens. A primeira, possibilitando uma sucinta abordagem teórica sobre as características do SVH (*Human Visual System* - Sistema Visual Humano), além de apresentar o estado da arte em que se encontra a compressão *wavelet* para diferentes tipos de imagens estáticas. A segunda parte, por sua vez, realiza uma abordagem dos principais conceitos relacionados à teoria *wavelet*; enquanto a terceira, apresenta a teoria *lifting*. Finalmente, a quarta parte se destina em apresentar as principais contribuições do trabalho. Deste modo, todo trabalho ficou estruturado da forma que se segue.

No Capítulo 2, foi realizado um apanhado bibliográfico sobre as principais características referente ao SVH assim como a importância das *wavelets* utilizadas em processamento de sinais, enfatizando a importância da transformada *wavelet* em aplicações médicas, de satélite e imagens naturais.

O Capítulo 3 apresentou os princípios básicos da transformada *wavelet* contínua 1D e 2D, suas características e princípios de funcionamento. No Capítulo 4, foi descrito um breve estudo acerca da codificação por subbanda resultando na implementação prática da transformada *wavelet* discreta através dos filtros do algoritmo rápido de Mallat, teoria esta que serve de base para o Capítulo 5, no qual foram apresentados a decomposição e síntese da TWD para o caso ortogonal, tanto para a circunstância unidimensional quanto bidimensional, assim como os sinais passa-baixas e de detalhes resultantes do processo de decomposição.

Para contornar as questões conflitantes impostas pela ortogonalidade, surgiu a biortogonalidade, abordada no Capítulo 6. A biortogonalidade possibilitou o uso da TWD na compressão de imagens com excelente desempenho, devido ao fato dos filtros biortogonais apresentarem características, tais

como, suporte compacto, simetria e regularidade. Complementando os requisitos essenciais para se obter um ótimo desempenho para a compressão de imagens, quando estendido para o caso bidimensional, coube ao Capítulo 7 apresentar as características desejáveis para a TWD na compressão de imagens, em que uma breve descrição das principais diferenças entre *wavelets* ortogonais e bi-ortogonais foi realizada, assim como um breve apanhado dos diversos tipos de *wavelets* geralmente encontrado na literatura, culminando com as principais características das *wavelets* utilizadas.

O Capítulo 8 possibilita o entendimento do método *lifting*, explicado de maneira sucinta mas cumprindo o objetivo de abordar as principais características que a compõem. Embora o método *lifting* e o método tradicional por banco de filtros resultem no mesmo resultado em PSNR no que diz respeito à realização da TWD, existe significativa diferença na proporção do custo computacional referente ao número de operações aritméticas utilizados por ambos métodos. A velocidade computacional, portanto, torna-se atrativa, cabendo ao Capítulo 9 ilustrar os resultados obtidos via ferramentas interpretada e compilada.

Finaliza-se este trabalho apresentando, nesse capítulo, os comentários finais e conclusões referentes aos resultados obtidos, possibilitando uma visão geral do trabalho, assim como o que pode ser esperado do mesmo para contribuições futuras.

10.3 Comentários sobre Contribuições, Simulações e Resultados

Todas as simulações realizadas e apresentadas no Capítulo 9 foram implementadas utilizando a ferramenta compilada DEV-C++ 4.9.9.2 e a ferramenta interpretada Matlab 7.1. Os tempos de processamento englobaram os valores de decomposição e reconstrução de imagens estáticas, onde todos os resultados foram obtidos tomando a média de 1000 execuções para cada situação, de modo que os resultados apresentassem o comportamento mais fiel possível.

Com base nos principais resultados sumarizados acima, junto das extensas simulações neste trabalho, verificou-se através de implementações práticas, foco da presente dissertação, o ganho teórico afirmado por Reichel [2], superando realmente as previsões da própria Daubechies e Sweldens [1] que estimavam um ganho máximo de duas vezes. Nesta verificação reside a principal contribuição desta pesquisa.

Entre as ferramentas comparadas, pode-se afirmar que utilizando a ferramenta compilada, DEV-C++ 4.9.9.2, os resultados obtidos estão de acordo com a literatura [1]. Para as *wavelets* de Daubechies e Symlets, o ganho que o método *lifting* oferece sobre o método tradicional é de aproximadamente duas vezes para os comprimentos de filtros que apresentam maior tamanho. Para as *wavelets* Biortogonais, ganhos próximos de três foram obtidos para as *wavelets* bior2.4, bior2.6, bior2.8, bior3.7, bior3.9 e bior4.4. As extensas simulações da Seção 9.2.1 também confirmam na prática as

afirmações teóricas de Reichel e de Daubechies e Sweldens quanto ao aumento do ganho do método *lifting* com relação ao método tradicional à medida em que se eleva o comprimento do filtro (n).

A família Biortogonal sempre apresentou maior ganho em tempo de processamento, dentre as famílias de *wavelets* testadas, corroborando sua excelente aplicação no campo de compressão de imagens, aliando este fator a suas vantajosas características: filtros mais flexíveis (decomposição diferente de reconstrução) e mais fáceis de serem projetados; simetria, reconstrução exata, alta regularidade e bom desempenho para compressão (características apresentadas nos Capítulos 6 e 7). Como contrapartida, a família Biortogonal somente apresentou ligeiro aumento de complexidade de implementação, uma vez que devem ser projetadas quatro estruturas de filtragens diferentes ao invés de duas.

Extensas simulações realizadas no presente trabalho permitiram observar interessantes características apontadas nas seções anteriores, tais como as afirmações sobre algoritmos de mais rápida execução estarem vinculados a *wavelets* de suporte compacto e características de suporte compacto das três famílias utilizadas nos experimentos.

Entretanto, o Matlab 7.1 por fazer uso de uma linguagem de programação interpretada de altíssimo nível, apresentou alguns resultados inconsistentes neste aspecto, chegando a inverter a lógica das referências acima citadas. Como linguagem interpretativa, o Matlab em todas as suas versões, no intuito de acelerar as implementações dos usuários, realiza excelentes otimizações nas estruturas de cálculos (somas e multiplicações) a nível de bits, operações e acessos à memória, provocando alterações no tempo de processamento de algoritmos, mesmo entre diferentes versões da própria ferramenta, como foi exemplificado através de indícios modestamente na Seção 9.4 e Apêndice A. Logo, os dados comparativos inconsistentes com as referências [1] e [2] obtidos neste trabalho possuem como provável explicação a aceleração obtida por tais otimizações internas ao Matlab. Neste contexto, o Matlab constitui-se em uma ferramenta poderosa e excelente, porém deve ser usado com ressalvas para procedimentos comparativos de tempo de processamento, em especial quando houver grande quantidade de acessos à memória, como é o caso de filtragens.

As imagens reconstruídas foram analisadas através do PSNR (*Peak Signal-to-Noise Ratio*), medida objetiva utilizada devido à sua extensa utilização tanto em artigos quanto na bibliografia de PDI. Como foi realizado apenas a decomposição e reconstrução do sinal, as únicas perdas de informação que poderiam intervir na reconstrução do sinal seriam provenientes dos filtros utilizados pelo processo. Dentro deste contexto, a ferramenta interpretada e a ferramenta compilada obtiveram reconstrução perfeita, assegurado por medida de PSNR infinita, corroborando com a perfeita implementação do *lifting*, o que implica em dizer que a reconstrução obtida pelo método *lifting* não apresentou deterioração na imagem reconstruída. Além do que, por uma rápida avaliação subjetiva, foi percebido que as imagens reconstruídas apresentaram níveis excelentes.

10.4 Sugestões para Trabalhos Futuros

Algumas sugestões que foram visualizadas durante a realização desta pesquisa foram:

- Realizar a TWD utilizando o esquema *lifting* com 3 ou mais bandas, com o propósito de analisar se há diferença quanto ao desempenho obtido para duas bandas, em termos de velocidade de processamento. Se houver, a utilização de mais bandas, implica em diferentes números de operações aritméticas e, neste sentido, torna-se viável saber em que proporção este fato se dá.
- Analisar a realização da TWD com outras famílias de *wavelets* e conseqüentemente, analisar em que proporção o ganho em velocidade entre os métodos *lifting* e tradicional pode ser alcançado.
- Medir a eficiência da compactação da energia de cada família, característica interessante para possibilitar ganho de compressão, para analisar qual a melhor *wavelet* do ponto de vista da compactação através de uma análise estática.

Assim, foram listadas as principais contribuições e resultados do presente trabalho e foram feitas algumas sugestões com o modesto propósito de poder auxiliar o desenvolvimento de outras pesquisas relacionadas com o processamento e a compressão digital de imagens estáticas.

Como se sabe, métodos *wavelet* são adequados para aplicações que requerem codificação rápida, por exemplo, em comunicações em tempo real via *internet* e, neste sentido, para execução de projetos relacionados aos meios rápidos de comunicação, a velocidade computacional é um dos parâmetros fundamental a ser considerado.

Referências Bibliográficas

- [1] I. Daubechies; W. Sweldens – “Factoring Wavelet Transforms into Lifting Steps” – *J. Fourier Anal. Appl.*, vol. 4, no. 3, 245 – 267, 1998.
- [2] J. Reichel – “On the Arithmetic and Bandwidth Complexity of the Lifting Scheme” – *IEEE Transactions on Image Processing*, vol.3, pp. 198-201, Oct. 2001.
- [3] A. L. M. C. S. Silva – “Procedimentos para Método Híbrido de Compressão de Imagens utilizando Transformadas *Wavelet* e Codificação Fractal” – Tese de Doutorado, Universidade Estadual de Campinas, Maio, 2005.
- [4] M. Antonini; M. Barlaud; P. Mathiew; I. Daubechies – “Image Coding Using Wavelet Transform” – *IEEE Transactions on Image Processing*, vol. 01, no. 02, Apr., 1992, pp. 205-230.
- [5] K. A. Kotteri et al., – “A Comparison of Hardware Implementations of the Biorthogonal 9/7 DWT: Convolution Versus Lifting” – *IEEE Transactions on Circuit and Systems*, vol.52, no. 52, May, 2005.
- [6] P. L. Shui; Z. Bao – “M-Band Biorthogonal Interpolating Wavelets via Lifting Scheme” – *IEEE Transactions on Signal Processing*, vol.52, no. 09, Sept., 2004.
- [7] M. Kseneman et al., – “Comparison Between Different Lifting Scheme Algorithms” – *Signals and Systems and Image Processing*, 15th International Conference on 25-28 June 2008, Page(s): 327-330.
- [8] W. K. Pratt – “Digital Image Processing” – 4a ed., *PIKS Scientific Inside*, Califórnia, 2007.
- [9] M. H. Pirenne – “Vision and the Eye” – 2a ed., *Associated Book Publishers*, London, 1967.
- [10] M. C. Ferraz – “Codificação de imagens” – Dissertação de Mestrado, Instituto de Matemática Pura e Aplicada, RJ, Brasil, Jun.,1998.
- [11] G. Wald – “Human Vision and the Spectrum” – *Science*, 101, 2635, Jun. 29, 653–658, 1945.

- [12] H.L. Resnikoff – “The Illusion of Reality” – Springer-Verlag, 1989.
- [13] D.J. Granrath – “The role of human visual models in image processing” – *Proceedings of the IEEE*, 69(5):552-561, May 1981.
- [14] P. K. Brown; G. Wald – “Visual Pigment in Single Rods and Cones of the Human Retina” – *Science*, 144, 3614, Apr. 3, 45–52, 1964.
- [15] G. Wald – “The Receptors for Human Color Vision” – *Science*, 145, 3636, Sep.. 4, 1007–1017, 1964.
- [16] H. Grassman – “On the Theory of Compound Colours” – *Philosophical Magazine*, Ser. 4, 7, Apr., 254–264, 1854.
- [17] Kenneth R. Castleman – “Digital Image Processing” – Ed. Prentice Hall, 1989.
- [18] R. Arthur – “Avaliação Objetiva de Codecs de Vídeo” – Dissertação de Mestrado, FEEC/Unicamp, SP, Brasil, Apr., 2002.
- [19] J. Guo; S. Mitra; B. Nutter; T. Karp – “A Fast and Low Complexity Image Codec based on Backward Coding of Wavelet Trees” – *Proceedings of IEEE Data Compression Conference (DCC’06)* pp. 292- 301, 2006.
- [20] V. Velisavljevic; B. Beferull-Lozano, M. Vetterli – “Space-Frequency Quantization for Image Compression With Directionlets” – *IEEE Transactions on Image Processing*, no. 9, Sep., 2007.
- [21] J. Rombaut; A. Pizurica; W. Philips – “Low Complexity Error Concealment of Wavelet Coded Images in Lossy Packet Networks” – *The 2nd IEE European Conference on Visual Media Production, CVMP 2005*, pp. 78-85, Nov, 2005.
- [22] L. Oliveira et al., "Uma visão sobre Codificação Wavelet de Imagens", *Revista Ciência e Tecnologia*, Unisal, Ano XI, nº18, Jun., 2008.
- [23] L.A. Drake et al. – “ Wavelet analysis in recruitment of loudness compensation” – *IEEE Transactions on Signal Processing*, v.41, n.12, p.3306-12, Dec. 1993.
- [24] S. Mallat – “A Theory for Multiresolution Signal Decomposition: The Wavelet Representation” – *IEEE Transactions on PAMI*, vol. 11, no.7, pp. 674-693, Jul., 1989.
- [25] E. J. Stollnitz; T.D. DeRose; D.H. Salesin – “Wavelets for computer graphics: a primer” – Part 1. *IEEE Computer Graphics and Applications*, v.15, n.3, p.76-84, May 1995.

- [26] E. J. Stollnitz; T.D. DeRose; D.H. Salesin – “Wavelets for computer graphics: a primer” – Part 2. *IEEE Computer Graphics and Applications*, v.15, n.4, p.75-85, July 1995.
- [27] A. Graps – “An introduction to wavelets” – *IEEE Computational Science & Engineering*, v.2, n.3, p.50-61, Summer 1995.
- [28] B. Vidakovic; P. Müller – “Wavelets for kids: a tutorial introduction” – Durham, Duke University, *Institute of Statistics and Decision Sciences*, 1991.
- [29] W. B. Franceschi – “Procedimentos e Práticas para Digitalização de Imagens Médicas” – Dissertação de Mestrado, UTFPR/Curitiba, SP, Brasil, Dez., 2006.
- [30] N. B. Amor; N.E.B. Amara – “DICOM Image Compression by Wavelet Transform” – *IEEE SMC*, 2002.
- [31] U. Bick – “PACS: the silent revolution” – *Eur Radiol*, 9: p. 1152-1160, 1999.
- [32] J. Kivijärvi, et al – “A comparison of lossless compression methods for medical images” – *Comput Med. Imaging Graph*, 22(4): p. 323-39, 1998.
- [33] Gilberto J. Garcia – “Sensoriamento Remoto, Princípios e Interpretação de Imagens” – São Paulo, Nobel, 1982.
- [34] James B. Campbell – “Introduction to Remote Sensing” – 2a ed, Ed. Taylor & Francis, 1996.
- [35] E. M. de Souza; A. Pagamisse; J. F. G. Monico; W. G. C. Polezel – “Comparação das Bases de Wavelets Ortonormais e Biorthonormais: Implementação, Vantagens e Desvantagens no Posicionamento com GPS” – *TEMA Tend. Mat. Apl. Comput.*, 8, no1, 149-158, 2007.
- [36] E. M. de Souza; A. Pagamisse; J. F. G. Monico; W. G. C. Polezel – “Processamento de Sinais com Wavelets: Redução do Efeito do Multicaminho do Sinal GPS” – *TEMA Tend. Mat. Apl. Comput.*, 6, no1, 177-186, 2005.
- [37] J. A. Neto – “Análise da Emergência da TV Digital e seus Impactos na cadeia Produtiva Eletrônica Brasileira” – *Journal of Technology Management & Innovation*, Issue 5, vol.1, 2006.
- [38] C. M. D. S. Freitas; O. M. Chubachi; P. R. G. Luzzardi; R. A. Cava – “Introdução à Visualização de Informações” – *RITA*, Vol.VIII, no2, 2001.
- [39] Z. Wang; Alan C. Bovik – “A Universal Image Quality Index” – *IEEE Signal Processing Letters*, 2002.

- [40] Z. Wang; Alan C. Bovik – “Why Is Image Quality Assessment So Difficult?”, *IEEE Signal Processing Letters*, 2002.
- [41] Z. Wang; Alan C. Bovik; H.R. Sheikh; E.P.Simoncelli – “Image Quality Assessment: From Error Visibility to Structural Similarity” – *IEEE Transactions on Image Processing*, 2004.
- [42] J. L. Mannos; D. J. Sakrison – “The Effects of a Visual Fidelity Criterion on the Encoding of Images” – *IEEE Transactions on Information Theory*, 1974.
- [43] M. C. Q. Farias – “Aplicação da Transformada Wavelet na Compressão de Imagens” – Dissertação de Mestrado, FEEC/UNICAMP, SP, Brasil, Jun., 1998.
- [44] A. Grossman; J. Morlet – “Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape” – *SIAM J. Applied Mathematics*, vol. 15, pp. 726-736, 1984.
- [45] I. Daubechies – “Ten Lectures on Wavelets” – SIAM, Philadelphia, USA, 1992.
- [46] C. F. Chui – “An Introduction to Wavelets” – *Academic Press Inc.*, San Diego, USA, 1992.
- [47] M. Vetterli; J. Kovacevic – “Wavelet and Subband Coding”, Prentice Hall, New Jersey, 1995, pp.488.
- [48] M. C. Q. Farias – "Aplicação da Transformada Wavelet na Compressão de Imagens", Dissertação de Mestrado, FEEC/Unicamp, SP, Brasil, Jun. 1998.
- [49] R. M. Young – “Wavelet Theory and its Applications” – *Kluwer Academic Publishers*, 1992.
- [50] J. W. Woods; S. D. O’Neill – “Subband Coding of Images” – *IEEE Transactions on ASSP*, vol. 34, pp. 1278-1288, 1986.
- [51] R.E. Crochiere – “Sub-band coding” – *Bell Systems Technical Journal*, pages 1633-1653, Sep., 1981.
- [52] E. H. Adelson; P. Burt – “The Laplacian Pyramid as a Compact Image Code” – *IEEE Transactions Commun*, vol. 31, pp.482-540, Oct., 1983.
- [53] O. Egger; W. Li – “Subband coding of images using asymmetrical lter banks” – *IEEE Transactions on Image Processing*, 4(4):478-485, Apr., 1995.
- [54] B. Mahesh; W.A. Pearlman – “Hexagonal sub-band coding for images” – *International Conference on Acoustics, Speech and Signal Processing*, 3:1953-1956, May, 1989.

- [55] E.H. Adelson; F.P. Simoncelli – “Nonseparable extensions of quadrature mirror filters to multiple dimensions” – *Proceedings of the IEEE*, 78(4):652-664, Apr., 1990.
- [56] R.H. Bamberg; M. Smith – “A filter bank for the directional decomposition of images: Theory and design” – *IEEE Transactions on Signal Processing*, 40(4):882-893, Apr., 1992.
- [57] S.G. Mallat – “Multifrequency Channel Decomposition of Images and Wavelet Models” – *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(12):2091-2110, Dez., 1989.
- [58] M. R. Raghuveer; A. S. Bopardikar – “Wavelet Transform: Introduction to Theory and Applications” – Addison Wesley, Califórnia, 1998, pp.308.
- [59] S. Mallat – “A Wavelet Tour of Signal Processing” – Ed. Academic Press, San Diego, USA, 2001.
- [60] E. H. Adelson; E. Simoncelli; R. Hingorani – “Orthogonal Pyramid Transforms for Image Coding” – *Proceedings of SPIE*, vol. 845, pp. 50-58, Oct., 1987.
- [61] B. Porat – “Digital Processing of Random Signals” – Ed. Prentice Hall, Pennsylvania, USA, 1992.
- [62] S. Mallat – “Multiresolution Approximation and Wavelets” – *IEEE Transactions on PAMI*, vol. 11, no.07, Jul, 1989.
- [63] I. Daubechies – “Orthogonal Bases of Compactly Supported Wavelets” – *Communications on Pure Applied Mathematics*, vol. XLI, pp. 990-996, 1988.
- [64] A. Cohen; J. C. Feauveau; I. Daubechies – “Biorthogonal Bases of Compactly Supported Wavelets” – *Communications on Pure Applied Mathematics*, vol. 45, pp. 485-560, 1992.
- [65] L. Prasad; S. S. Iyengar – “Wavelet Analysis With Applications to Image Processing” – Ed. CRC Press, Florida, USA, 1997.
- [66] V. I. B. Sablón; Y. Iano – “Processamento e Compressão do Sinal de Vídeo Utilizando a Transformada Wavelet” – Dissertação de Doutorado, FEEC/UNICAMP, SP, Brasil, Dez., 2002.
- [67] I. Daubechies – “Orthogonal Bases of Compactly Supported Wavelets II – Variations on a Theme” – ATT&Bell Lab., Tech. Report TM 11217-89111617, USA, 1990.
- [68] G. Strang; T. Nguyen – “Wavelets and filter banks” – Wellesley, Wellesley-Cambridge Press, 490p., 1996.

- [69] Y. Meyer – “Wavelets: algorithms and applications” – Philadelphia, SIAM, 133p., 1993.
- [70] M.V. Wickerhauser – “Adapted wavelet analysis from theory to software” – Wellesley, A.K.Peters, 486p., 1994.
- [71] H. M. de Oliveira; L. R. Soares; T. H. Falk – “A Family of Wavelets and a New Orthogonal Multiresolutional Analysis Based on the Nyquist Criterion, *IEEE SBrT Int. Telecomm. Symposium*, Natal, RN, Sep., 2002.
- [72] K. Kuzume; K. Nijjima – “Design of Optimal Lifting Wavelet Filters for Data Compression” – *IEEE*, 1998.
- [73] M. Kseneman et al., – “Comparison Between Different Lifting Scheme Algorithms” – *Systems, Signals and Image Processing*, IEEE, *15th International Conference*, 25-28 Jun., 2008.
- [74] A. Skodras; C. Christopoulos; T. Ebrahimi – “The JPEG 2000 Still Image Compression Standard” – *IEEE Signal Processing Magazine*, Sep. 2001.
- [75] M. Shim; A. Laine – “Overcomplete Lifted Representations for Multiscale Feature Analysis” – *Proceedings of IEEE International Conference on Image Processing*, ICIP98, vol. pp.242-246, Oct, 1998.
- [76] W. Sweldens – “The Lifting Scheme: A Construction of Second Generation Wavelets” – *Appl. Comp. Harmon. Anal.*, vol.3, no.2, pp.186- 200, 1996.
- [77] W. Sweldens – “The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets” – *SIAM J. Math. Anal.*, vol. 9, no.2, pp.511-546, 1997.
- [78] W. Sweldens – “The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions” – *Wavelet Applications III*, A. Laine and M. Unser, Ed., Proc. SPIE, San Diego, CA, vol. 2529, pp.68-79, 1995.
- [79] Y. Zhang; M. Zeytinoglu – “Improved Lifting Scheme for Block Subband Coding” – *IEEE*, 1999.
- [80] J. Kovavcevic; W. Sweldens – “Wavelet families of increasing order in arbitrary dimensions,” *IEEE Trans. Image Processing*, vol. 9, pp. 480–496, Mar., 2000.
- [81] M. Jansen; P.Oonincx – “Second Generation Wavelets and Applications” – Ed. Springer, 2005.
- [82] A.C. Harbo; A. Jensen – “Wavelets and the Scheme Lifting” – *Department of Mathematical Sciences*, Aalborg University, Denmark, 2007.

APÊNDICE A

TEMPO DE PROCESSAMENTO x CARACTERÍSTICAS DA IMAGEM (VIA FERRAMENTA INTERPRETADA)

Os dados comparativos inconsistentes com as referências [1] e [2] obtidos neste trabalho, via ferramenta interpretada, possuem como provável explicação a aceleração obtida por tais otimizações internas ao Matlab. Fortes indícios apontam que os resultados de tempo de processamento obtidos através das ferramenta interpretada dependem das características específicas da imagem a ser processada.

A imagem Crosses (Fig. 9.1), por exemplo, difere do conjunto de imagens testadas 256x256 por apresentar baixa concentração de energia de aproximação em diversas implementações de famílias *wavelets* (Daubechies, Symlets e Biorrtogonais). As Tabelas 10.1-10.3 complementam, respectivamente, esta informação.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	99,2647%	0,3412%	0,4651%	0,1935%	0,0767%
Bridge	96,6097%	0,7214%	1,5454%	1,0568%	0,7881%
Lena	96,8312%	0,7919%	1,8112%	0,8907%	0,4669%
Goldhill	97,9880%	0,8284%	0,9001%	0,5720%	0,5399%
Crosses	23,0829%	1,3291%	19,7751%	25,5429%	31,5991%

Tabela 10.1: *Wavelets* de Daubechies: energia dos coeficientes para as imagens 256x256 testadas.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	99,2928%	0,1142%	0,4723%	0,1726%	0,0623%
Bridge	96,4054%	0,2088%	1,7358%	1,0944%	0,7644%
Lena	97,0122%	0,2841%	1,7844%	0,8075%	0,3959%
Goldhill	97,9347%	0,3380%	0,9710%	0,5617%	0,5326%
Crosses	23,3027%	0,6041%	20,1625%	25,4588%	31,0760%

Tabela 10.2: *Wavelets* de Symlets: energia dos coeficientes para as imagens 256x256 testadas.

	Aprox.	Desvio Padrão	Detalhes-3º nível	Detalhes-2º nível	Detalhes-1º nível
Bird	98,7260%	0,5571%	0,9301%	0,2739%	0,0700%
Bridge	94,0953%	2,5438%	3,5034%	1,7161%	0,6852%
Lena	94,9231%	1,9757%	3,4051%	1,2877%	0,3841%
Goldhill	96,4617%	1,9107%	2,0443%	1,0042%	0,4898%
Crosses	31,3468%	9,6506%	28,2312%	20,6099%	19,8121%

Tabela 10.3: *Wavelets* Biortogonais: energia dos coeficientes para as imagens 256x256 testadas.

As Figs. 10.1-10.6 ilustram para diversas *wavelets* (Daubechies, Symlets e Biortogonais) este índice de que o Matlab apresenta resultados em tempo de processamento dependentes das características específicas da imagem para um mesmo processamento. Nas figuras em questão são apresentados os ganhos entre método *lifting* e método tradicional para as imagens-teste e comparados aos ganhos entre método *lifting* e tradicional para a imagem *Crosses*, que possui característica mais específica quanto à concentração de energia.

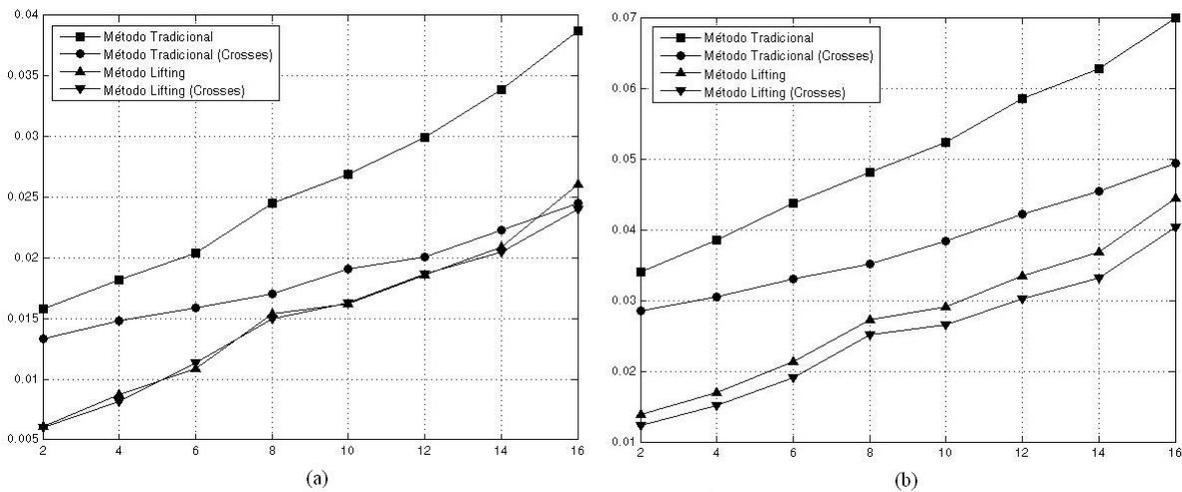


Figura 10.1: Comparação $T_p \times n$ para *wavelets* de Daubechies para a imagem *Crosses*: (a) Decomposição; (b) Decomposição e Reconstrução.

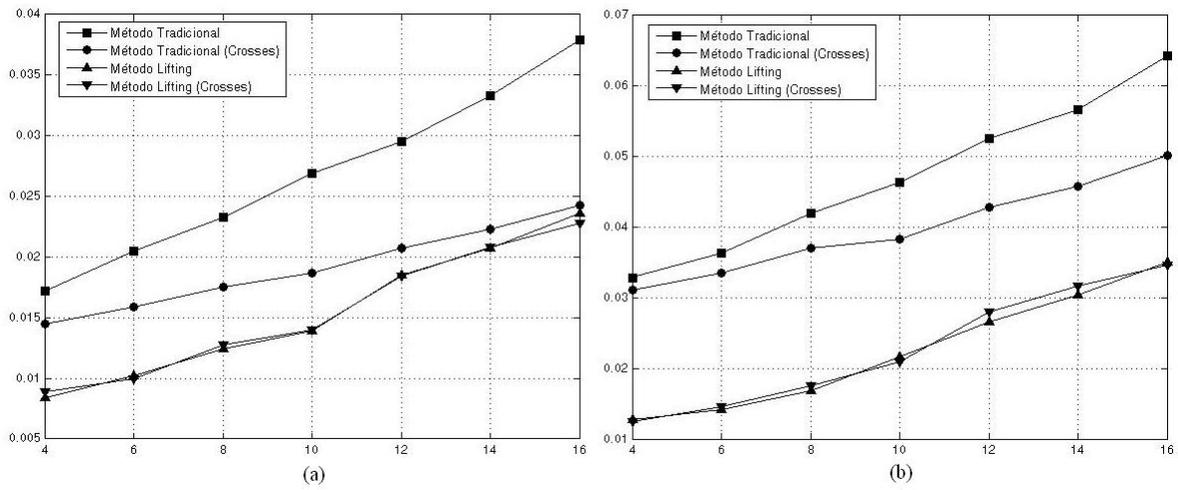


Figura 10.2: Comparação $T_p \times n$ para *wavelets* Symlets para a imagem Crosses: (a) Decomposição; (b) Decomposição e Reconstrução.

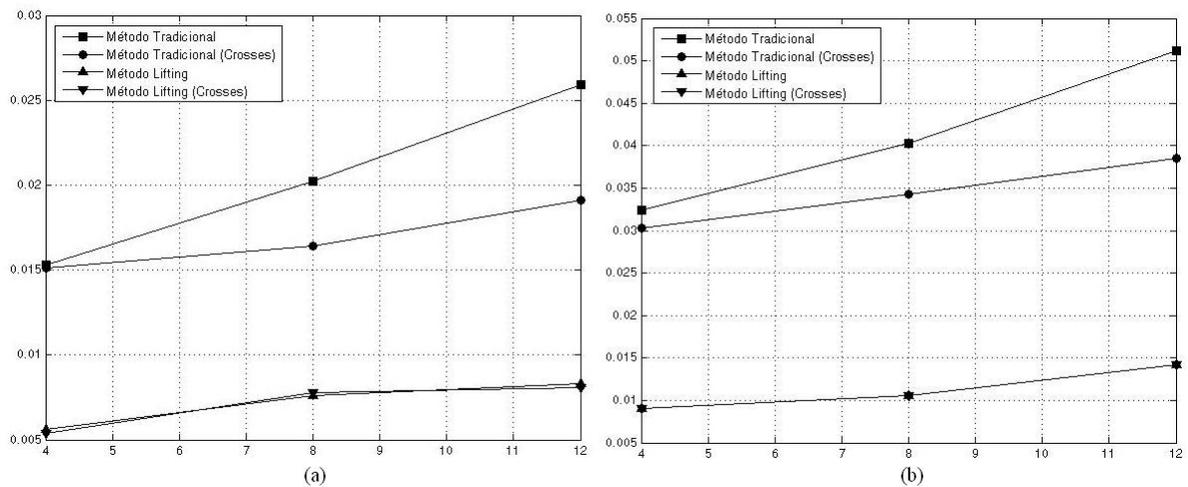


Figura 10.3: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem1) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.

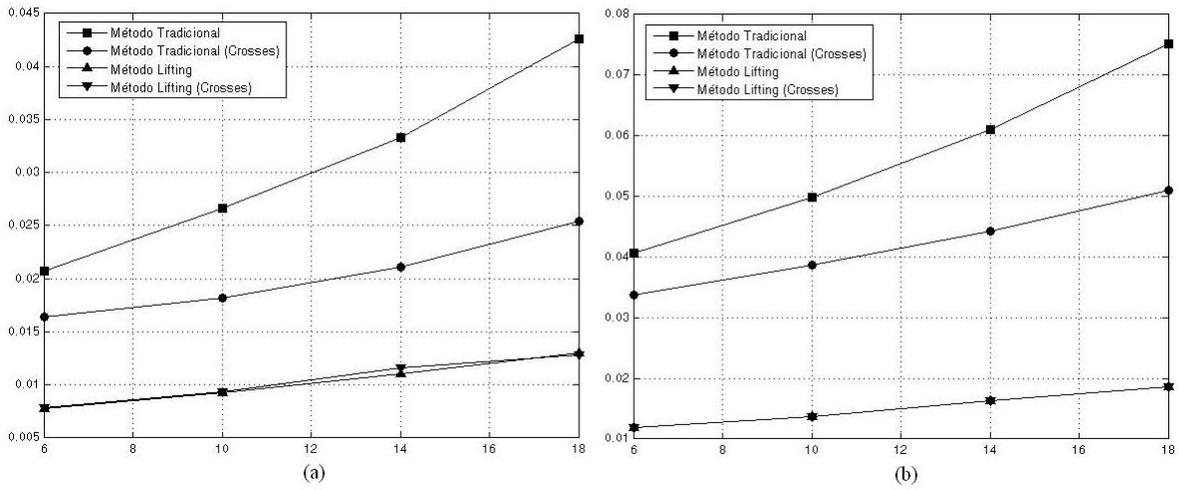


Figura 10.4: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem2) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.

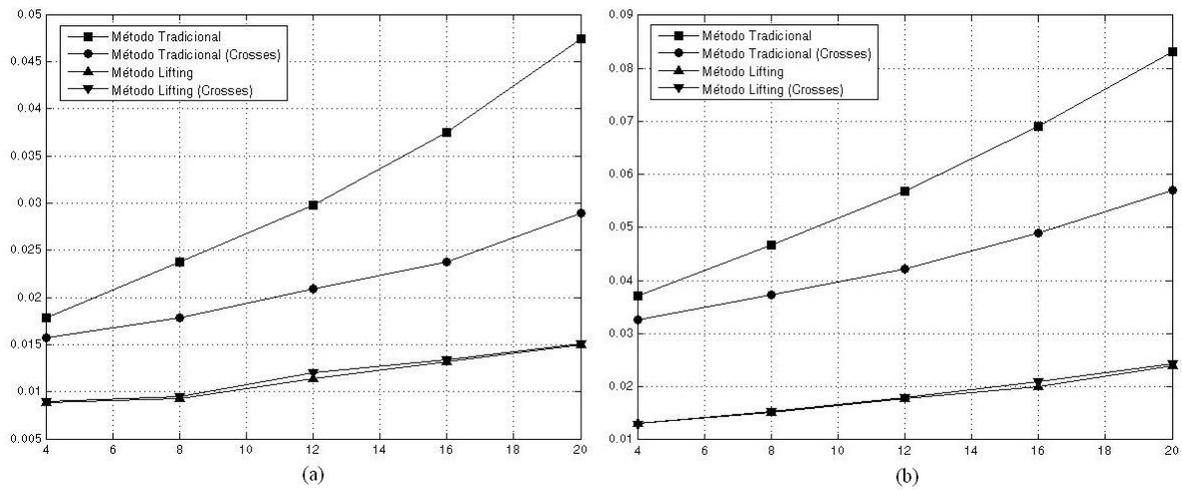


Figura 10.5: Comparação $T_p \times n$ para *wavelets* Biortogonais (ordem3) para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.

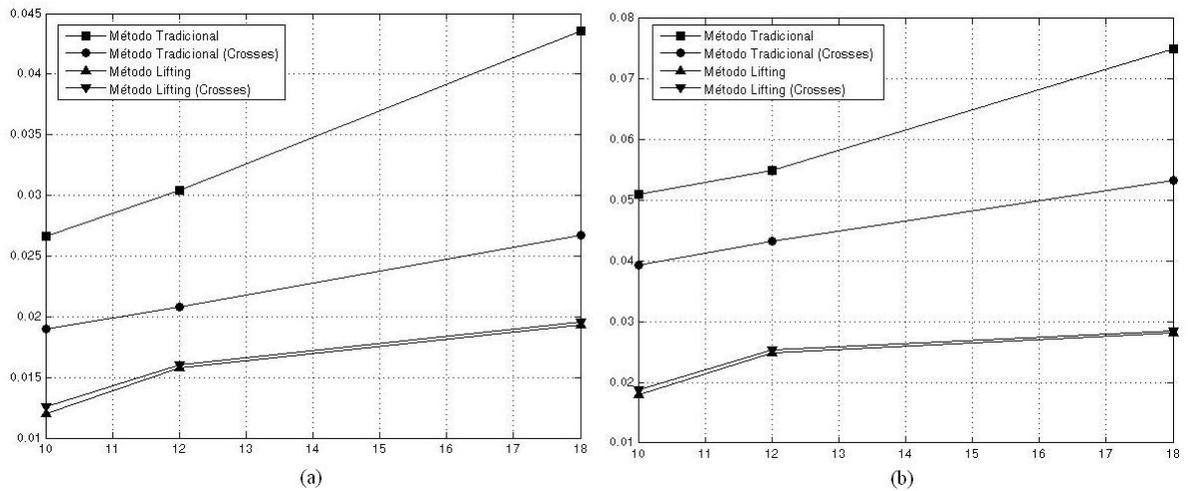


Figura 10.6: Comparação T_p x n para *wavelets* Biortogonais para a imagem Crosses: (a) Decomposição; (b) Decomposição e reconstrução.

Nota-se que há indícios de que para imagens de baixa energia, por exemplo, o Matlab promove ganho menor por parte do *lifting* se comparado a imagens com energia mais significativa, o que não ocorre em linguagens compiladas para os procedimentos na dissertação em questão. Entretanto, para atestar esta informação, um levantamento mais extenso em simulações e com outras imagens, não à Crosses, teria que ser realizado.

O objetivo deste apêndice foi apenas para evidenciar este indício corroborando as hipóteses levantadas quanto a distorções provocadas por ferramentas interpretadas.

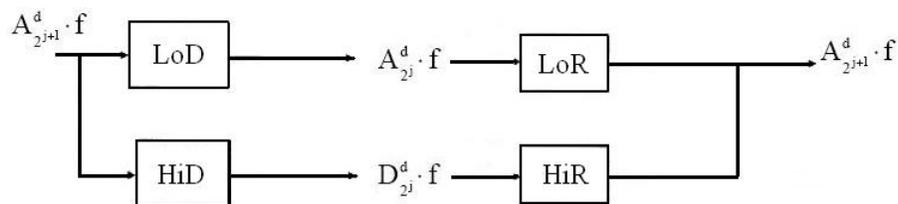
APÊNDICE B

FILTROS UTILIZADOS NAS SIMULAÇÕES

Neste apêndice são apresentados os coeficientes dos filtros utilizados nas simulações para as famílias de *wavelets* Daubechies, Symlets e Biortogonais. Cada uma destas *wavelets* é representada por uma quantidade de coeficientes e conforme a ordem destas *wavelets* aumenta, maior o número que a caracterizam.

Todas *wavelets* analisadas estão descritas em função dos filtros passa-baixa de decomposição (LoD), passa-alta de decomposição (HiD), passa-baixa de reconstrução (LoR) e passa-alta de reconstrução (HiR). Estes valores foram fundamentais para obtenção da TWD do sinal de entrada.

Estes filtros podem ser esquematizados para a decomposição e reconstrução do sinal $A_{2^{j+1}}^d \cdot f$, a grosso modo, por:



Nas páginas seguintes encontram-se os respectivos coeficientes utilizados.

db1	LoD	0.7071	0.7071						
	HiD	-0.7071	0.7071						
	LoR	0.7071	0.7071						
	HiR	0.7071	-0.7071						
db2	LoD	-0.1294	0.2241	0.8365	0.4830				
	HiD	-0.4830	0.8365	-0.2241	-0.1294				
	LoR	0.4830	0.8365	0.2241	-0.1294				
	HiR	-0.1294	-0.2241	0.8365	-0.4830				
db3	LoD	0.0352	-0.0854	-0.1350	0.4599	0.8069	0.3327		
	HiD	-0.3327	0.8069	-0.4599	-0.1350	0.0854	0.0352		
	LoR	0.3327	0.8069	0.4599	-0.1350	-0.0854	0.0352		
	HiR	0.0352	0.0854	-0.1350	-0.4599	0.8069	-0.3327		
db4	LoD	-0.0106	0.0329	0.0308	-0.1870	-0.0280	0.6309	0.7148	0.2304
	HiD	-0.2304	0.7148	-0.6309	-0.0280	0.1870	0.0308	-0.0329	-0.0106
	LoR	0.2304	0.7148	0.6309	-0.0280	-0.1870	0.0308	0.0329	-0.0106
	HiR	-0.0106	-0.0329	0.0308	0.1870	-0.0280	-0.6309	0.7148	-0.2304
db5	LoD	0.0033	-0.0126	-0.0062	0.0776	-0.0322	-0.2423	0.1384	0.7243
		0.6038	0.1601						
	HiD	-0.1601	0.6038	-0.7243	0.1384	0.2423	-0.0322	-0.0776	-0.0062
		0.0126	0.0033						
	LoR	0.1601	0.6038	0.7243	0.1384	-0.2423	-0.0322	0.0776	-0.0062
		-0.0126	0.0033						
	HiR	0.0033	0.0126	-0.0062	-0.0776	-0.0322	0.2423	0.1384	-0.7243
		0.6038	-0.1601						
db6	LoD	-0.0011	0.0048	0.0006	-0.0316	0.0275	0.0975	-0.1298	-0.2263
		0.3153	0.7511	0.4946	0.1115				
	HiD	-0.1115	0.4946	-0.7511	0.3153	0.2263	-0.1298	-0.0975	0.0275
		0.0316	0.0006	-0.0048	-0.0011				
	LoR	0.1115	0.4946	0.7511	0.3153	-0.2263	-0.1298	0.0975	0.0275
		-0.0316	0.0006	0.0048	-0.0011				
	HiR	-0.0011	-0.0048	0.0006	0.0316	0.0275	-0.0975	-0.1298	0.2263
		0.3153	-0.7511	0.4946	-0.1115				
db7	LoD	0.0004	-0.0018	0.0004	0.0126	-0.0166	-0.0380	0.0806	0.0713
		-0.2240	-0.1439	0.4698	-0.7291	0.3965	0.0779		
	HiD	-0.0779	0.3965	-0.7291	0.4698	0.1439	-0.2240	-0.0713	0.0806
		0.0380	-0.0166	-0.0126	0.0004	0.0018	0.0004		
	LoR	0.0779	0.3965	0.7291	0.4698	-0.1439	-0.2240	0.0713	0.0806
		-0.0380	-0.0166	0.0126	0.0004	-0.0018	0.0004		
	HiR	0.0004	0.0018	0.0004	-0.0126	-0.0166	0.0380	0.0806	-0.0713
		-0.2240	0.1439	0.4698	-0.7291	0.3965	-0.0779		
db8	LoD	-0.0001	0.0007	-0.0004	-0.0049	0.0087	0.0140	-0.0441	-0.0174
		0.1287	0.0005	-0.2840	-0.0158	0.5854	0.6756	0.3129	0.0544
	HiD	-0.0544	0.3129	-0.6756	0.5854	0.0158	-0.2840	-0.0005	0.1287
		0.0174	-0.0441	-0.0140	0.0087	0.0049	-0.0004	-0.0007	-0.0001

	LoR	0.0544	0.3129	0.6756	0.5854	-0.0158	-0.2840	0.0005	0.1287
		-0.0174	-0.0441	0.0140	0.0087	-0.0049	-0.0004	0.0007	-0.0001
	HiR	-0.0001	-0.0007	-0.0004	0.0049	0.0087	-0.0140	-0.0441	0.0174
		0.1287	-0.0005	-0.2840	0.0158	0.5854	-0.6756	0.3129	-0.0544
sym2	LoD	-0.1294	0.2241	0.8365	0.4830				
	HiD	-0.4830	0.8365	-0.2241	-0.1294				
	LoR	0.4830	0.8365	0.2241	-0.1294				
	HiR	-0.1294	-0.2241	0.8365	-0.4830				
sym3	LoD	0.0352	-0.0854	-0.1350	0.4599	0.8069	0.3327		
	HiD	-0.3327	0.8069	-0.4599	-0.1350	0.0854	0.0352		
	LoR	0.3327	0.8069	0.4599	-0.1350	-0.0854	0.0352		
	HiR	0.0352	0.0854	-0.1350	-0.4599	0.8069	-0.3327		
sym4	LoD	-0.0758	-0.0296	0.4976	0.8037	0.2979	-0.0992	-0.0126	0.0322
	HiD	-0.0322	-0.0126	0.0992	0.2979	-0.8037	0.4976	0.0296	-0.0758
	LoR	0.0322	-0.0126	-0.0992	0.2979	0.8037	0.4976	-0.0296	-0.0758
	HiR	-0.0758	0.0296	0.4976	-0.8037	0.2979	0.0992	-0.0126	-0.0322
sym5	LoD	0.0273	0.0295	-0.0391	0.1994	0.7234	0.6340	0.0166	-0.1753
		-0.0211	0.0195						
	HiD	-0.0195	-0.0211	0.1753	0.0166	-0.6340	0.7234	-0.1994	-0.0391
		-0.0295	0.0273						
	LoR	0.0195	-0.0211	-0.1753	0.0166	0.6340	0.7234	0.1994	-0.0391
		0.0295	0.0273						
	HiR	0.0273	-0.0295	-0.0391	-0.1994	0.7234	-0.6340	0.0166	0.1753
		-0.0211	-0.0195						
sym6	LoD	0.0154	0.0035	-0.1180	-0.0483	0.4911	0.7876	0.3379	-0.0726
		-0.0211	0.0447	0.0018	-0.0078				
	HiD	0.0078	0.0018	-0.0447	-0.0211	0.0726	0.3379	-0.7876	0.4911
		0.0483	-0.1180	-0.0035	0.0154				
	LoR	-0.0078	0.0018	0.0447	-0.0211	-0.0726	0.3379	0.7876	0.4911
		-0.0483	-0.1180	0.0035	0.0154				
	HiR	0.0154	-0.0035	-0.1180	0.0483	0.4911	-0.7876	0.3379	0.0726
		-0.0211	-0.0447	0.0018	0.0078				
sym7	LoD	0.0027	-0.0010	-0.0126	0.0305	0.0679	-0.0496	0.0174	0.5361
		0.7678	0.2886	-0.1400	-0.1078	0.0040	0.0103		
	HiD	-0.0103	0.0040	0.1078	-0.1400	-0.2886	0.7678	-0.5361	0.0174
		0.0496	0.0679	-0.0305	-0.0126	0.0010	0.0027		
	LoR	0.0103	0.0040	-0.1078	-0.1400	0.2886	0.7678	0.5361	0.0174
		-0.0496	0.0679	0.0305	-0.0126	-0.0010	0.0027		
	HiR	0.0027	0.0010	-0.0126	-0.0305	0.0679	0.0496	0.0174	-0.5361
		0.7678	-0.2886	-0.1400	0.1078	0.0040	-0.0103		
sym8	LoD	-0.0034	-0.0005	0.0317	0.0076	-0.1433	-0.0613	0.4814	0.7772
		0.3644	-0.0519	-0.0272	0.0491	0.0038	-0.0150	-0.0003	0.0019
	HiD	-0.0019	-0.0003	0.0150	0.0038	-0.0491	-0.0272	0.0519	0.3644
		-0.7772	0.4814	0.0613	-0.1433	-0.0076	0.0317	0.0005	-0.0034

	LoR	0.0019	-0.0003	-0.0150	0.0038	0.0491	-0.0272	-0.0519	0.3644
		0.7772	0.4814	-0.0613	-0.1433	0.0076	0.0317	-0.0005	-0.0034
	HiR	-0.0034	0.0005	0.0317	-0.0076	-0.1433	0.0613	0.4814	-0.7772
		0.3644	0.0519	-0.0272	-0.0491	0.0038	0.0150	-0.0003	-0.0019
bior1.1	LoD	0.7071	0.7071						
	HiD	-0.7071	0.7071						
	LoR	0.7071	0.7071						
	HiR	0.7071	-0.7071						
bior1.3	LoD	-0.0884	0.0884	0.7071	0.7071	0.0884	-0.0884		
	HiD	0	0	-0.7071	0.7071	0	0		
	LoR	0	0	0.7071	0.7071	0	0		
	HiR	-0.0884	-0.0884	0.7071	-0.7071	0.0884	0.0884		
bior1.5	LoD	0.0166	-0.0166	-0.1215	0.1215	0.7071	0.7071	0.1215	-0.1215
		-0.0166	0.0166						
	HiD	0	0	0	0	-0.7071	0.7071	0	0
		0	0						
	LoR	0	0	0	0	0.7071	0.7071	0	0
		0	0						
	HiR	0.0166	0.0166	-0.1215	-0.1215	0.7071	-0.7071	0.1215	0.1215
		-0.0166	-0.0166						
bior2.2	LoD	0	-0.1768	0.3536	1.0607	0.3536	-0.1768		
	HiD	0	0.3536	-0.7071	0.3536	0	0		
	LoR	0	0.3536	0.7071	0.3536	0	0		
	HiR	0	0.1768	0.3536	-1.0607	0.3536	0.1768		
bior2.4	LoD	0	0.0331	-0.0663	-0.1768	0.4198	0.9944	0.4198	-0.1768
		-0.0663	0.0331						
	HiD	0	0	0	0.3536	-0.7071	0.3536	0	0
		0	0						
	LoR	0	0	0	0.3536	0.7071	0.3536	0	0
		0	0						
	HiR	0	-0.0331	-0.0663	0.1768	0.4198	-0.9944	0.4198	0.1768
		-0.0663	-0.0331						
bior2.6	LoD	0	-0.0069	0.0138	0.0470	-0.1077	-0.1699	0.4475	0.9667
		0.4475	-0.1699	-0.1077	0.0470	0.0138	-0.0069		
	HiD	0	0	0	0	0	0.3536	-0.7071	0.3536
		0	0	0	0	0	0		
	LoR	0	0	0	0	0	0.3536	0.7071	0.3536
		0	0	0	0	0	0		
	HiR	0	0.0069	0.0138	-0.0470	-0.1077	0.1699	0.4475	-0.9667
		0.4475	0.1699	-0.1077	-0.0470	0.0138	0.0069		
bior2.8	LoD	0	0.0015	-0.0030	-0.0129	0.0289	0.0530	-0.1349	-0.1638
		0.4626	0.9516	0.4626	-0.1638	-0.1349	0.0530	0.0289	-0.0129
		-0.0030	0.0015						

	HiD	0	0	0	0	0	0	0	0.3536
		-0.7071	0.3536	0	0	0	0	0	0
		0	0						
	LoR	0	0	0	0	0	0	0	0.3536
		0.7071	0.3536	0	0	0	0	0	0
		0	0						
	HiR	0	-0.0015	-0.0030	0.0129	0.0289	-0.0530	-0.1349	0.1638
		0.4626	-0.9516	0.4626	0.1638	-0.1349	-0.0530	0.0289	0.0129
		-0.0030	-0.0015						
bior3.1	LoD	-0.3536	1.0607	1.0607	-0.3536				
	HiD	-0.1768	0.5303	-0.5303	0.1768				
	LoR	0.1768	0.5303	0.5303	0.1768				
	HiR	-0.3536	-1.0607	1.0607	0.3536				
bior3.3	LoD	0.0663	-0.1989	-0.1547	0.9944	0.9944	-0.1547	-0.1989	0.0663
	HiD	0	0	-0.1768	0.5303	-0.5303	0.1768	0	0
	LoR	0	0	0.1768	0.5303	0.5303	0.1768	0	0
	HiR	0.0663	0.1989	-0.1547	-0.9944	0.9944	0.1547	-0.1989	-0.0663
bior3.5	LoD	-0.0138	0.0414	0.0525	-0.2679	-0.0718	0.9667	0.9667	-0.0718
		-0.2679	0.0525	0.0414	-0.0138				
	HiD	0	0	0	0	-0.1768	0.5303	-0.5303	0.1768
		0	0	0	0				
	LoR	0	0	0	0	0.1768	0.5303	0.5303	0.1768
		0	0	0	0				
	HiR	-0.0138	-0.0414	0.0525	0.2679	-0.0718	-0.9667	0.9667	0.0718
		-0.2679	-0.0525	0.0414	0.0138				
bior3.7	LoD	0.0030	-0.0091	-0.0168	0.0747	0.0313	-0.3012	-0.0265	0.9516
		0.9516	-0.0265	-0.3012	0.0313	0.0747	-0.0168	-0.0091	0.0030
	HiD	0	0	0	0	0	0	-0.1768	0.5303
		-0.5303	0.1768	0	0	0	0	0	0
	LoR	0	0	0	0	0	0	0.1768	0.5303
		0.5303	0.1768	0	0	0	0	0	0
	HiR	0.0030	0.0091	-0.0168	-0.0747	0.0313	0.3012	-0.0265	-0.9516
		0.9516	0.0265	-0.3012	-0.0313	0.0747	0.0168	-0.0091	-0.0030
bior3.9	LoD	-0.0007	0.0020	0.0051	-0.0206	-0.0141	0.0991	0.0123	-0.3202
		0.0021	0.9421	0.9421	0.0021	-0.3202	0.0123	0.0991	-0.0141
		-0.0206	0.0051	0.0020	-0.0007				
	HiD	0	0	0	0	0	0	0	0
		-0.1768	0.5303	-0.5303	0.1768	0	0	0	0
		0	0	0	0				
	LoR	0	0	0	0	0	0	0	0
		0.1768	0.5303	0.5303	0.1768	0	0	0	0
		0	0	0	0				
	HiR	-0.0007	-0.0020	0.0051	0.0206	-0.0141	-0.0991	0.0123	0.3202
		0.0021	-0.9421	0.9421	-0.0021	-0.3202	-0.0123	0.0991	0.0141
		-0.0206	-0.0051	0.0020	0.0007				

bior4.4	LoD	0	0.0378	-0.0238	-0.1106	0.3774	0.8527	0.3774	-0.1106
		-0.0238	0.0378						
	HiD	0	-0.0645	0.0407	0.4181	-0.7885	0.4181	0.0407	-0.0645
		0	0						
	LoR	0	-0.0645	-0.0407	0.4181	0.7885	0.4181	-0.0407	-0.0645
		0	0						
	HiR	0	-0.0378	-0.0238	0.1106	0.3774	-0.8527	0.3774	0.1106
		-0.0238	-0.0378						
bior5.5	LoD	0	0	0.0397	0.0079	-0.0545	0.3456	0.7367	0.3456
		-0.0545	0.0079	0.0397	0				
	HiD	-0.0135	-0.0027	0.1367	-0.0935	-0.4768	0.8995	-0.4768	-0.0935
		0.1367	-0.0027	-0.0135	0				
	LoR	0.0135	-0.0027	-0.1367	-0.0935	0.4768	0.8995	0.4768	-0.0935
		-0.1367	-0.0027	0.0135	0				
	HiR	0	0	0.0397	-0.0079	-0.0545	-0.3456	0.7367	-0.3456
		-0.0545	-0.0079	0.0397	0				
bior6.6	LoD	0	0.0019	-0.0019	-0.0170	0.0119	0.0497	-0.0773	-0.0941
		0.4208	0.8259	0.4208	-0.0941	-0.0773	0.0497	0.0119	-0.0170
		-0.0019	0.0019						
	HiD	0	0	0	0.0144	-0.0145	-0.0787	0.0404	0.4178
		-0.7589	0.4178	0.0404	-0.0787	-0.0145	0.0144	0	0
		0	0						
	LoR	0	0	0	0.0144	0.0145	-0.0787	-0.0404	0.4178
		0.7589	0.4178	-0.0404	-0.0787	0.0145	0.0144	0	0
		0	0						
	HiR	0	-0.0019	-0.0019	0.0170	0.0119	-0.0497	-0.0773	0.0941
		0.4208	-0.8259	0.4208	0.0941	-0.0773	-0.0497	0.0119	0.0170
		-0.0019	-0.0019						

Tabela 10.4: Coeficientes dos filtros utilizados nas implementações.

APÊNDICE C

CÓDIGOS-FONTES DESENVOLVIDOS

As principais partes dos códigos desenvolvidos em Matlab 7.1 durante a pesquisa encontram-se da forma:

Ferramenta compilada

MÉTODO LIFTING

```
int main(int argc, char *argv[])
{
float x[tamanho_da_imagem], x2[tamanho_da_imagem];
float d0[dimensão] = {coeficiente_filtro};
float p0[dimensão] = {coeficiente_filtro};
float s[dimensão] = {normalização_filtro};
int n,linha,N,nivel;
FILE* arquivo;
int leitura;
float escrita;
int count;
LARGE_INTEGER ticksPerSecond;
LARGE_INTEGER tick;
LARGE_INTEGER start_ticks, end_ticks, cputime;
arquivo = fopen("arquivo.dat","rb");
N= tamanho_resolução;
if (!QueryPerformanceFrequency(&ticksPerSecond))
printf("\tno QueryPerformance not present");
printf ("\tfreq test: %I64Ld ticks/sec\n",ticksPerSecond );
if (!QueryPerformanceCounter(&tick) ) printf("no go counter not installed");
```

```

printf ("\tQueryPerformanceCounter testpoint : %I64Ld ticks\n",tick);
QueryPerformanceCounter(&start_ticks);
for (count = 0; count < valor; count++){
for (nivel = 0; nivel < nivel_desejado; nivel++) {
/* Lifting para a transformada wavelet direta em uma direção */
for (linha = 0; linha < N; linha++)
{
for (n = 1; n<N/2; n++){
x2[linha*N/2 + n + N*N/2] = x[linha*N+2*n] +d0[0]*x[linha*N+2*n+1];
x2[linha*N/2 + n] = x[linha*N+2*n+1] +p0[0]*x2[linha*N/2 + n + N*N/2];
x2[linha*N/2 + n] = s[0]*x2[linha*N/2 + n];
x2[linha*N/2 + n + N*N/2] = s[1]*x2[linha*N/2 + n + N*N/2];
}
}
for (linha = 0; linha < N; linha++) {
for (n = 0; n<N; n++){
x[linha*N + n] = x2[linha + n*N];
}
}
N = N/2;
}
}
QueryPerformanceCounter(&end_ticks);
cputime.QuadPart = (end_ticks.QuadPart- start_ticks.QuadPart)/valor;
printf ("\tElapsed CPU time test: %.9f sec ticks %d\n",
((float)cputime.QuadPart/(float)ticksPerSecond.QuadPart),cputime.QuadPart);
getchar();
}

```

MÉTODO TRADICIONAL

```

int main(int argc, char *argv[])
{
float x[tamanho_da_imagem], x2[tamanho_da_imagem];
float LO_D[dimensão] = {coeficiente_filtro};
float HI_D[dimensão] = {coeficiente_filtro};
float LO_R[dimensão] = {coeficiente_filtro};

```

```

float HI_R[dimensão] = {coeficiente_filtro};
int n,linha,N,nivel;
FILE* arquivo;
int leitura;
float escrita;
int count;
LARGE_INTEGER ticksPerSecond;
LARGE_INTEGER tick;
LARGE_INTEGER start_ticks, end_ticks, cputime;
arquivo = fopen("arquivo.dat","rb");
N= tamanho_resolucao;
if (!QueryPerformanceFrequency(&ticksPerSecond))
printf("\tno go QueryPerformance not present");
printf ("\tfreq test: %I64Ld ticks/sec\n",ticksPerSecond );
if (!QueryPerformanceCounter(&tick) ) printf("no go counter not installed");
printf ("\tQueryPerformanceCounter testpoint : %I64Ld ticks\n",tick);
QueryPerformanceCounter(&start_ticks);
for (count = 0; count < valor; count++){
for (nivel = 0; nivel < nivel_desejado; nivel++) {
/*Convolução para a transformada wavelet direta em uma direção*/
for (linha = 0; linha < N; linha++)
{
for (n = 0; n<N/2; n++) {
x2[linha*N/2+n]=LO_D[dimensão]*x[linha*N+2*n+1]+LO_D[dimensão]*x[linha*N+2*n];
x2[linha*N/2 + n + N*N/2] = HI_D[dimensão]*x[linha*N+2*n+1]+HI_D[dimensão]*x[linha*N+2*n];
}
}
for (linha = 0; linha < N; linha++) {
for (n = 0; n<N; n++){
x[linha*N + n] = x2[linha + n*N];
}
}
N = N/2;
}
}

```

```

QueryPerformanceCounter(&end_ticks);
cputime.QuadPart = (end_ticks.QuadPart- start_ticks.QuadPart)/valor;
printf ("\tElapsed CPU time test: %.9f sec ticks %d\n",
((float)cputime.QuadPart/(float)ticksPerSecond.QuadPart),cputime.QuadPart);
getchar();
}

```

Ferramenta interpretada

MÉTODO LIFTING

%Escolha a imagem, a *wavelet* e o nível desejado de decomposições:

```

I = double(imread('imagem'))/255;
wname = 'nome_da_wavelet';
nivel = nivel_desejado;
ls= liftwave(wname);
displs(ls);
[Lo_D,Hi_D,Lo_R,Hi_R]=wfilters(wname);
%Realiza a transformada wavelet via lifting do sinal (imagem) e calcula-se o tempo:
t=cputime;
xDec = lwt2(imagem,ls,nivel_desejado);
%Extrai os coeficientes de aproximação no nível desejado
ca = lwtcoef2('ca',xDec,ls,nivel_desejado);
%Extrai os coeficientes referente ao detalhe horizontal no nível desejado
ch=lwtcoef2('ch',xDec,ls,nivel_desejado,1);
%Extrai os coeficientes referente ao detalhe vertical no nível desejado
cv=lwtcoef2('cv',xDec,ls,nivel_desejado,1);
%Extrai os coeficientes referente ao detalhe diagonal no nível desejado
cd=lwtcoef2('cd',xDec,ls,nivel_desejado,1);
e=cputime-t;

```

MÉTODO TRADICIONAL

%Escolha a imagem, a *wavelet* e o nível desejado de decomposições:

```

I = double(imread('imagem'))/255;
wname = 'nome_da_wavelet';
nivel = nivel_desejado;
[Lo_D,Hi_D,Lo_R,Hi_R]=wfilters(wname);

```

```
t=cputime;  
[C,S] = wavedec2(I,nivel_desejado,wname);  
%Extrai os coeficientes de aproximação no nível desejado  
cA = appcoef2(C,S,wname,nivel_desejado);  
%Extrai os coeficientes de aproximação no nível desejado  
[cHd1,cVd1,cDd1] = detcoef2('all',C,S,nivel_desejado);  
e=cputime-t;
```

LISTA DE PUBLICAÇÕES

06/2008 **Artigo:** “Uma Visão sobre Codificação Wavelet de Imagens ”

Autores: L. Oliveira, A. L. M. Cruz, F. S. Silva, Y. Iano, R. F. L. Chavez, V. I. B. Sablón.

Tema: Wavelet, Multiresolução, Biortogonalidade, Esquema Lifting, Compressão de Dados.

Aprovado pela Revista Ciência e Tecnologia UNISAL Ano XI - no.18, 2008.