



Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
Departamento de Comunicações



COMPRESSÃO DE IMAGENS DIGITAIS COMBINANDO TÉCNICAS *WAVELET* E *WEDGELET* NO AMBIENTE DE COMUNICAÇÕES MÓVEIS

Autor: Ricardo Barroso Leite

Orientador: Prof. Dr. Yuzo Iano

Co-orientadora: Prof. Dr. Ana Lúcia Mendes Cruz Silvestre da Silva

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

Banca Examinadora

Prof. Dr. Yuzo Iano (presidente)	— DECOM/FEEC/UNICAMP
Prof. Dr. David Fernandes	— ITA
Prof. Dr. Gustavo Fraidenraich	— DECOM/FEEC/UNICAMP

Campinas – SP

7 de julho de 2011

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

Leite, Ricardo Barroso

L536c Compressão de imagens digitais combinando técnicas wavelet e wedglet no ambiente de comunicações móveis / Ricardo Barroso Leite. --Campinas, SP: [s.n.], 2011.

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

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

1. Teoria da codificação. 2. Compressão de imagens. 3. Wavelets (Matemática). 4. Wavelets (Matemática) - Processamento de dados. 5. Televisão digital. I. Iano, Yuzo. II. Silva, Ana Lucia Mendes Cruz Silvestre da . III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Título em Inglês: Digital Image compression combining wavelet and wedgelet techniques in mobile communication environment

Palavras-chave em Inglês: Coding theory, Image compression, Wavelets (Mathematical), Wavelets (Mathematical) - Data processing, Digital television

Área de concentração: Telecomunicações e Telemática

Titulação: Mestrado em Engenharia Elétrica

Banca examinadora: David Fernandes, Gustavo Fraidenraich

Data da defesa: 07/07/2011

Programa de Pós Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Ricardo Barroso Leite

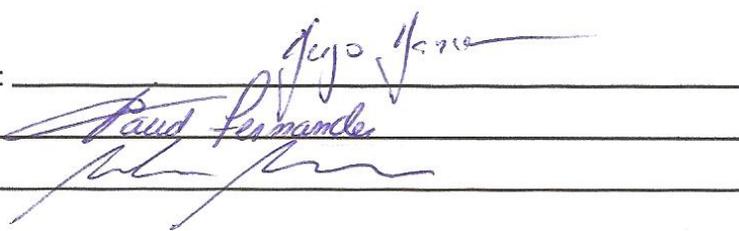
Data da Defesa: 7 de julho de 2011

Título da Tese: "Compressão de Imagens Digitais Combinando Técnicas Wavelet e Wedgelet no Ambiente de Comunicações Móveis"

Prof. Dr. Yuzo Iano (Presidente): _____

Prof. Dr. David Fernandes: _____

Prof. Dr. Gustavo Fraidenraich: _____



The image shows three handwritten signatures in blue ink, each written over a horizontal line. The first signature is for Prof. Dr. Yuzo Iano, the second for Prof. Dr. David Fernandes, and the third for Prof. Dr. Gustavo Fraidenraich.

Resumo

Os avanços em telecomunicações e o desenvolvimento dos equipamentos digitais impulsionaram diversas áreas de pesquisa relacionadas à codificação e compressão de imagens. Dentre as áreas de atuação destacam-se as aplicações para dispositivos móveis (celulares, *smartphones*, *iPhones*, *iPads* entre outros), que se caracterizam por baixas taxas de transmissão de dados. Entretanto, imagens codificadas com os padrões atualmente em estado-da-arte apresentam artefatos visuais característicos, como efeito de bloco e *ringing*. Para contornar a inabilidade das transformadas ortogonais em lidar com a geometria, é proposto na literatura o uso de dicionários *wedgelet* e da decomposição *cartoon*-textura. Nesse contexto, propõe-se um método de codificação híbrido *wedgelet-wavelet* inédito que preserva componentes de *cartoon* e textura, superando em qualidade visual ao uso de dicionários isolados e se aproximando do desempenho de sistemas de codificação completos, tais como o padrão JPEG 2000. Os ganhos de desempenho, principalmente em qualidade visual das imagens reconstruídas para baixas taxas de dados, indicam que a metodologia apresentada pode vir a ser incluída em sistemas de transmissão com restrições de largura de banda, como por exemplo a TV digital móvel.

Palavras-chave: codificação, compressão de imagens, *wavelets*, *wedgelets*, TV digital, TV móvel.

Abstract

Advances in telecommunications and the development of digital equipments have improved several research areas related to coding and image compression. Among these application fields are the mobile devices (cellphones, smartphones, iPhones, iPad, and others), characterized by low data transmission rates. However, images encoded by state-of-the-art standards present characteristic visual artifacts, like blocking and ringing effects. To surpass the disadvantages of orthogonal transforms in dealing with geometry, wedgelets dictionaries and cartoon-texture decomposition are proposed in literature. In this context, a new hybrid wedgelet-wavelet coding method that preserves cartoon and texture components is proposed, achieving better visual quality than the use of isolated dictionaries, approaching the performance of complete codification systems, such as the JPEG 2000. The performance gains, especially concerning visual quality of the reconstructed images using low data rates, show that this methodology might be adopted in restricted bandwidth transmission systems, such as the digital mobile TV.

Key-words: coding, image compression, wavelets, wedgelets, digital TV, mobile TV.

Agradecimentos

Após concluir esse trabalho, ao qual me dediquei muito a fazer, mesmo errando algumas vezes e tendo noites mal-dormidas, o resultado que fica é mais do que positivo. A carreira científica cada vez mais me fascina e convida a aprender e a pesquisar mais. Não poderia deixar de agradecer a todos que me apoiaram, me incentivaram e me inspiraram.

Agradeço ao meu orientador professor Dr. Yuzo, que desde a graduação me deu a oportunidade de trabalhar com ele e que sempre me deu muita liberdade criativa e teve confiança em meu trabalho. Suas reuniões sempre estimularam nosso grupo de pesquisa e acredito que todos nós orientados somos muito gratos. Aprendi muito em cada projeto desenvolvido em seu laboratório (RNP – STB-SCAN, CAPES – RH-TVD, ITASAT).

Agradeço a minha co-orientadora professora Dr. Ana Lúcia e seu marido Fernando, que além de acompanharem de perto todo meu trabalho e colaborarem na escrita deste material, se tornaram meus grandes amigos e sempre foram um exemplo para mim de pesquisadores. Espero continuarmos trabalhando juntos muito tempo ainda.

Agradeço aos professores Dr. David e Dr. Gustavo por participarem da defesa da dissertação, aos quais não conhecia profundamente, mas com os quais espero ter outras oportunidades de colaboração acadêmica.

Agradeço a Faculdade de Engenharia Elétrica e de Computação da Unicamp, na qual obtive minha graduação e agora concluo o mestrado e cujo nível de excelência acadêmica sempre ficará marcado em minha formação. Ao CNPq e a CAPES por ter me dado apoio financeiro para realizar as pesquisas por meio de bolsas de Iniciação Científica e Mestrado, respectivamente.

A toda minha família dedico este trabalho por acreditarem em mim e me amparem em momentos de frustração. Em especial minha mãe, Maria Cláudia, por todo suporte que só uma mãe sabe dar. Sem ela nada disso teria sido possível. Obrigado por ser tão carinhosa e colocar os filhos sempre em primeiro lugar.

Aos meus pais, Roberto e Daniel, cada qual ao seu modo tem ajudado a formar o que eu sou hoje. Orgulho-me de ter os dois em minha vida como referenciais de caráter.

Aos meus irmãos Nádia, Gustavo e Karla, a força de vontade de vocês me motiva e me faz querer estar presente nas conquistas de vocês, assim como têm estado presentes nas minhas. A meus avós, meus tios e tias, meus primos, todos que têm me apoiado muito e aos quais sou muito grato.

Em especial, a minha namorada Angela, minha sempre fiel companheira e melhor amiga. Sua presença alegria minha vida e *a cada dia descubro uma nova razão pra te amar*. Agradeço por todo amor, confiança e paciência que tem tido comigo durante todo o tempo de pesquisa. Sem você me

apoiando este trabalho não teria sido feito. A toda sua família, em especial sua mãe Jolina, que tem acreditado bastante em mim.

Agradeço aos meus colegas do Laboratório de Comunicações Visuais que têm sido grandes amigos além de terem me ajudado quando a pesquisa se tornava difícil. Agradeço também aos meus colegas de graduação, que embora em caminhos distintos, têm sempre mantido o contato e a amizade.

A Deus por iluminar meu caminho e não me deixar fraquejar. Maior do que o orgulho de chegar até aqui é a fé em continuar lutando.

Ricardo Barroso Leite

“Temos de nos tornar na mudança que queremos ver.” - Mahatma Gandhi

Agradecimentos

Agradecimentos especiais ao programa CAPES RH-TVD da **Coordenação de Aperfeiçoamento de Pessoal de Nível Superior** tanto pelo apoio financeiro quanto pelo incentivo acadêmico para que este trabalho pudesse ser realizado.

Sumário

LISTA DE PUBLICAÇÕES	17
LISTA DE FIGURAS.....	19
LISTA DE TABELAS	25
LISTA DE ABREVIATURAS.....	27
CAPÍTULO 1 INTRODUÇÃO	29
1.1 – Contribuições e Organização do Trabalho.....	30
CAPÍTULO 2 COMUNICAÇÕES MÓVEIS DE IMAGENS E VÍDEO	33
2.1 Televisão Digital no Brasil	34
2.2 A Televisão Digital Móvel.....	35
2.3 Padrões de TV Digital Móvel.....	36
2.3.1 DVB-H (<i>Digital Video Broadcasting</i>).....	37
2.3.2 ISDB (<i>Integrated Services Digital Broadcasting</i>) – 1SEG	37
2.3.3 DMB (<i>Digital Media Broadcasting</i>)	38
2.3.4 CMMB (<i>China Multimedia Mobile Broadcasting</i>).....	38
2.3.5 MobiTV Inc.....	39
2.3.6 MediaFLO (<i>Foward Link Only</i>).....	39
2.3.7 MBMS (<i>Multimedia Broadcast Multicast Service</i>) – UMTS.....	39
2.4 Comentário.....	40
CAPÍTULO 3 QUALIDADE VISUAL.....	41
3.1 Sistema Visual Humano	41
3.2 Codificação por Transformadas Ortogonais.....	44
3.3 Artefatos Visuais	46
CAPÍTULO 4 METODOLOGIA <i>CARTOON</i> -TEXTURA	49
CAPÍTULO 5 TRANSFORMADA <i>WAVELET</i>	55

5.1	Introdução: <i>Waves</i> x <i>Wavelets</i>	56
5.2	Transformada <i>Wavelet</i> Contínua (TWC) 1D	56
5.3	Interpretação da TWC por Banco de Filtros.....	58
5.4	TWC Bidimensional.....	60
5.5	Expansão em Série da <i>Wavelet</i> (SW)	61
5.5.1	<i>Wavelets</i> Diádicas Invariantes na Translação (<i>Translation-Invariant Dyadic Wavelet Transform</i>)[55].....	62
5.5.2	<i>Wavelets</i> Diádicas	63
5.5.3	SW Bidimensional.....	65
5.6	Princípios da Transformada <i>Wavelet</i> Discreta (TWD)	65
5.6.1	Codificação por Subbanda.....	65
5.6.2	Introdução ao Algoritmo Rápido para TWD	67
5.7	Resolução e Aproximação por Multiresolução em $L^2(\mathbb{R})$	68
5.8	Aproximação por Multiresolução em $L^2(\mathbb{R})$: propriedades.....	68
5.8.1	O Operador de Projeção	69
5.8.2	Propriedades [55]	69
5.9.1	O Sinal de Aproximação ($A_{2^j}^d \cdot f$)	72
5.9.2	O Sinal de Detalhamento ($D_{2^j}^d \cdot f$)	75
5.9.3	A Decomposição Completa	79
5.11.1	Bases <i>Wavelet</i> Separáveis.....	81
5.12.1	O Sinal de Aproximação 2D ($A_{2^j}^d \cdot f$)	83
5.12.2	O Sinal de Detalhamento 2D ($D_{2^j}^d \cdot f$)	85
5.12.3	A Decomposição 2D Completa	87
5.13	Reconstrução ou Síntese Ortogonal 2D.....	89
5.14	Comentários	90
CAPÍTULO 6 NOVAS FORMAS DE REPRESENTAÇÃO ESPARSA		91
6.1	Aproximações Lineares	92

6.2	Aproximações Não-Lineares	94
6.3	O Problema das Transformadas Isotrópicas e o Uso de Representações Esparsas...	97
6.4	Representações Esparsas e Dicionários Geométricos	99
6.5	Dicionário <i>Wedgelet</i> de 1ª Ordem	103
6.6	Extensões ao Modelo <i>Wedgelets – Surflets e Platelets</i>	108
6.7	Modelos Híbridos – Decomposição <i>Cartoon-Textura</i>	109
CAPÍTULO 7 PROPOSTA E RESULTADOS.....		115
7.1	Método Proposto (WWB – <i>Wedgelet-Wavelet</i> por Blocos).....	115
7.2	Introdução e Observações sobre as Simulações	117
7.3	Escolha dos Melhores Parâmetros.....	118
7.3.1	Determinação do Número de Ângulos	118
7.3.2	Determinação do Número de Distâncias	121
7.3.3	Determinação da Função <i>Wavelet</i>	124
7.4	Codificação <i>Wedgelet-Wavelet</i>	126
7.4.1	Comparações Adicionais com Métodos <i>Wedgelet</i>	135
7.4.2	Comparações Adicionais com Métodos <i>Wavelet</i>	136
7.4.3	Comparação com outros Métodos <i>Wedgelet-Wavelet</i>	141
7.5	Comentários	143
CAPÍTULO 8 CONCLUSÕES E PERSPECTIVAS PARA TRABALHOS FUTUROS		145
8.1	Comentários sobre o Contexto e Contribuições	145
8.2	Conclusões Breves sobre os Capítulos e Relação com a Proposta.....	146
8.3	Comentários sobre Simulações e Resultados	146
8.4	Sugestões para Trabalhos Futuros	147
REFERENCIAS BIBLIOGRÁFICAS		149

Lista de Publicações

LEITE, R. B. ; IANO, Y. ; SILVA, A. L. M. C. S. ; SILVA, F. S. . Source and Channel Coding with Unequal Error Protection for Image Transmission in AWGN Channel. In: International Workshop on Telecommunications 2009, 2009, São Paulo. Proceedings of International Workshop on Telecommunications 2009, February 16th-19th, São Paulo, Brazil, 2009.

LEITE, R. B. ; SILVA, A. L. M. C. S. ; SILVA, F. S. . FPGA Lifted Spline 9-7: Finite Precision Optimization for Image Processing. In: Brazilian Technology Simposium, 2010, Campinas. Proceedings of Brazilian Technology Simposium, 2010.

IANO, Y. ; SAOTOME, O. ; LARICO, R. F. ; SILVA, A. L. M. C. S. ; SILVA, F. S. ; LEITE, R. B. . Turbo convolutional, turbo product and ldpc codes comparisons for low power applications. In: The 13th International Symposium on Wireless Personal Multimedia Communications, 2010, Recife. Proceedings of the 13th International Symposium on Wireless Personal Multimedia Communications. Recife : Iecom Edition, 2010. p. 29-29.

OLIVEIRA, L.; IANO, Y.; A. L. M. C. S. ; SILVA, F. S.; LEITE, R. B.; SILVA, F. S.; Desempenho Comparativo da Velocidade Computacional Entre Os Esquemas Lifting E Bancos de Filtros Wavelets, Revista Ciência e Tecnologia, 2011.

LEITE, R. B.; IANO, Y.; SILVA, A. L. M. C. S.; SILVA, F. S.; Aplicação de Dicionários Geométricos à Compressão de Imagens, Revista Ciência e Tecnologia, 2011.

Lista de Figuras

FIGURA 2.1 – PREVISÃO DO NÚMERO DE TELESPECTADORES DE TV MÓVEL NO MUNDO ENTRE 2007 E 2012 [16].....	36
FIGURA 2.2 – DIVISÃO DO ESPECTRO OFDM DE 13 SEGMENTOS DO PADRÃO ISDB-T _B	38
FIGURA 2.3 – MODOS DE TRANSMISSÃO UMTS (A) UNICAST (PONTO A PONTO) (B) BROADCAST / MULTICAST (SERVING-SERVIDORA; SUPPORT-SUPORTE; CONTENT-CONTEÚDO; SERVICE-SERVIÇO; NETWORK-REDE; CORE-NÚCLEO; ACCESS-ACESSO; BROADCAST-RADIODIFUSÃO; GATEWAY-ENTRADA ; TERRESTRIAL-TERRESTRE)	40
FIGURA 3.1 – OLHO HUMANO (A) SEÇÃO RETA HORIZONTAL [24] (B) CÉLULAS RECEPTORAS [26].	41
FIGURA 3.2 – MODELAMENTO DO HVS (A) FUNÇÃO DE SENSITIVIDADE A CONTRASTE CSF (CONTRAST SENSITIVITY-SENSIBILIDADE AO CONTRASTE; SPATIAL FREQUENCY-FREQÜÊNCIA ESPACIAL; CYCLES/DEGREE- CICLOS/GRAU) (B) DIAGRAMA LOGARÍTMICO DE ESTÍMULOS SENOIDAIS [28].	42
FIGURA 3.3 – RESPOSTA DIRECIONAL DAS CÉLULAS RECEPTORAS, INTERSEÇÃO ENTRE: (A) 2 BLOCOS (B) 4 BLOCOS [29].....	43
FIGURA 3.4 - DIFERENTES TIPOS DE CONTEÚDO PARA TV MÓVEL [31] (A) NOTÍCIAS (B) ESPORTES (C) SÉRIES (D) DESENHOS ANIMADOS (E) TELETEXO (F) CLIPE MUSICAL	44
FIGURA 3.5– DIAGRAMA DE BLOCOS DE UM CODIFICADOR JPEG 2000 [41].....	46
FIGURA 3.6 - EFEITO DE BLOCO EM IMAGEM CODIFICADA DCT (A) ORIGINAL (B) RECUPERADA [43]	46
FIGURA 3.7 – RINGING E EMBAÇAMENTO (BLUR) EM UMA DESCONTINUIDADE UNIDIMENSIONAL [44]	47
FIGURA 3.8 - EXEMPLO DE RINGING EM IMAGEM CODIFICADA POR TWD (A) ORIGINAL (B) RECUPERADA	47
FIGURA 4.1 – DECOMPOSIÇÃO CARTOON-TEXTURA DE [48] (A) ORIGINAL (B) CARTOON (C) TEXTURA	50
FIGURA 4.2 – RESULTADOS DA CODIFICAÇÃO HÍBRIDA DE SPRLJAN E IZQUIERDO PARA A IMAGEM 'BARBARA' [47].....	51
FIGURA 5.1 – EXEMPLOS DE FUNÇÕES WAVELET (A) HAAR (B) DAUBECHIES 2	56

FIGURA 5.2. ESPAÇO TEMPO-FREQÜÊNCIA: (A) SINAL ORIGINAL; (B) ANÁLISE NO DOMÍNIO <i>WAVELET</i>	56
FIGURA 5.3. <i>WAVELETS</i> DE MORLET DESLOCADAS.....	57
FIGURA 5.4. INTERPRETAÇÃO DA TWC POR BANCO DE FILTROS	58
FIGURA 5.5. TWC: (A) BANDAS DE FREQÜÊNCIAS DAS JANELAS; (B) RESOLUÇÃO NO PLANO TEMPO-FREQÜÊNCIA [72].....	59
FIGURA 5.6 - TWC CALCULADA COM “CHAPÉU MEXICANO”. AS CORES “PRETA”, “CINZA” E “BRANCA”CORREPONDEM, RESPECTIVAMENTE, AOS VALORES “POSITIVO”, “ZERO”E “NEGATIVO” DOS COEFICIENTES [55].....	60
FIGURA 5.7 - <i>SPLINE</i> QUADRÁTICA (A) FUNÇÃO <i>WAVELET</i> (B) FUNÇÃO-ESCALA [55].....	62
FIGURA 5.8 - <i>WAVELET</i> DIÁDICA NAS ESCALAS $2^{-7} \leq 2^{-j} \leq 2^{-3}$ PARA SINAL DEFINIDO SOBRE $[0,1]$. A CURVA INFERIOR CARREGA FREQÜÊNCIAS MAIS BAIXAS (ESCALAS MAIORES QUE ESCALA 2^{-3}) [55].	63
FIGURA 5.9 - GRADE DE AMOSTRAGEM NO PLANO DESLOCAMENTO-ESCALA PARA <i>WAVELETS</i> DIÁDICAS ($A_0=1/2$)	64
FIGURA 5.10 – ESQUEMA DE CODIFICAÇÃO POR SUBBANDA [74].....	66
FIGURA 5.11 – CODIFICAÇÃO POR SUBBANDA DE DUAS BANDAS E RECONSTRUÇÃO.	67
FIGURA 5.12 - ALGORITMO DE MALLAT (FWT)	68
FIGURA 5.13 - ALGORITMO INVERSO DE MALLAT	68
FIGURA 5.14 – ESQUEMA ILUSTRATIVO DA RELAÇÃO ENTRE $\phi_{j,n}(x)$ E V_{2^j} PARA $J \in [-1,1]$. [76] ..	72
FIGURA 5.17 – COEFICIENTES <i>WAVELET</i> CALCULADOS NAS ESCALAS 2-J COM <i>WAVELETS SPLINES</i> CÚBICAS (APROXIMAÇÃO $A_{2^j}^d \cdot f$ PARA $J=-5$ E DETALHES $D_{2^j}^d \cdot f$ [55].....	80
FIGURA 5.18 – DIAGRAMA EM BLOCOS PARA IMAGEM DE: (A) UM ESTÁGIO DE DECOMPOSIÇÃO <i>WAVELET</i> ; (B) UM ESTÁGIO DE RECONSTRUÇÃO <i>WAVELET</i>	85
FIGURA 5.19 – DECOMPOSIÇÃO <i>WAVELET</i> ORTOGONAL 2D PARA $J = 3$ [76].	88
FIGURA 5.20 - DECOMPOSIÇÃO <i>WAVELET</i> DAUBECHIES SEPARÁVEL (PASSA-ALTA FOI RESSALTADA PARA COMPARAÇÃO) (A) ORIGINAL (B) DECOMPOSIÇÃO <i>WAVELET</i>	88
FIGURA 5.21 - TF DE UMA FUNÇÃO ESCALA SEPARÁVEL E DAS 3 <i>WAVELETS</i> SEPARÁVEIS CORRESPONDENTES CALCULADAS PARA UMA <i>WAVELET</i> DE DAUBECHIES [55].	89
FIGURA 6.1 – RECONSTRUÇÃO DE UM SINAL UNIDIMENSIONAL DE VARIAÇÃO LIMITADA (A) SINAL ORIGINAL (B) APROXIMAÇÃO FOURIER (C) APROXIMAÇÃO POR <i>WAVELET</i> DAUBECHIES 4 [55]	93
FIGURA 6.2 – (A) SINAL ORIGINAL F (B) DECOMPOSIÇÃO <i>WAVELET</i> , COM OS $N/6$ MAIORES COEFICIENTES <i>WAVELET</i> SYMMLET 4 (C) APROXIMAÇÃO NÃO-LINEAR	96

FIGURA 6.3 – COEFICIENTES <i>WAVELET</i> DE UMA IMAGEM SUAVE COM CONTORNOS BEM DEFINIDOS. AS COMPONENTES DE ALTA FREQUÊNCIA (DETALHES) FORAM NORMALIZADAS PARA MELHOR VISUALIZAÇÃO. [45]	96
FIGURA 6.4 – ESQUEMAS DE APROXIMAÇÃO (A) ORIGINAL ($N = 2562$) (B) DECOMPOSIÇÃO <i>WAVELET</i> , PONTOS ESCUROS CORRESPONDEM A $\left \langle f, \psi_{j,n}^k \rangle \right > T$ (C) APROXIMAÇÃO LINEAR COM $N/16$ (D) APROXIMAÇÃO NÃO-LINEAR COM $N/16$	97
FIGURA 6.5 INEFICIÊNCIA DA <i>WAVELET</i> ISOTRÓPICA EM REPRESENTAR A GEOMETRIA DE UMA IMAGEM. NÚMERO DE COEFICIENTES: (A) 13 (B) 26 (C) 52 [85]	98
FIGURA 6.6 – ARTEFATOS DO TIPO <i>RINGING</i> GERADOS PELA QUANTIZAÇÃO INADEQUADA DOS COEFICIENTES <i>WAVELET</i> AO LONGO DE UM CONTORNO.	98
FIGURA 6.7 – PROJEÇÃO <i>WAVELET</i> (A) IMAGEM $f = \mathbf{1}_\Omega$ (B) NA ESCALA 2^j , AS <i>WAVELETS</i> $\psi_{j,n}^l$ TÊM UM SUPORTE QUADRADO DE LARGURA PROPORCIONAL A 2^j . ESSE SUPORTE CORRESPONDE A UMA GRADE DE INTERVALO 2^j , INDICADA PELOS PEQUENOS PONTOS. PONTOS MAIORES INDICAM <i>WAVELETS</i> COM SUPORTE QUE CRUZA A FRONTEIRA DE Ω PARA $\langle f, \psi_{j,n}^l \rangle \neq \mathbf{0}$	99
FIGURA 6.8 – GEOMETRIAS DE APROXIMAÇÃO (A) ORIGINAL (B) APROXIMAÇÃO COM TRIANGULAÇÃO (C) APROXIMAÇÃO COM UMA BASE <i>WAVELET</i> [86].	100
FIGURA 6.9 – FUNÇÕES BASE 2D (A) <i>WAVELET</i> HAAR ISOTRÓPICA (B) <i>WAVELET</i> HAAR ANISOTRÓPICA COM ORIENTAÇÃO DE 45°	101
FIGURA 6.10 – DECOMPOSIÇÃO <i>WAVELET</i> DIRECIONAL COM $\alpha = 0, \pi/4, \pi/2, 3\pi/4$ E ESCALAS 2^j E 2^{j+1} [55].....	101
FIGURA 6.11 – APROXIMAÇÕES EM BLOCOS NÃO RETANGULARES (A) TRIANGULO ADAPTADO PARA APROXIMAR UMA CURVA (B) ELEMENTO <i>CURVELET</i> DE APROXIMAÇÃO [86].	102
FIGURA 6.12 – DECOMPOSIÇÃO <i>WAVELET</i> DIRECIONAL (A) DECOMPOSIÇÃO <i>WAVELET</i> (B-C) <i>WAVELETS</i> DIRECIONAIS ALPERT DE DIFERENTES COMPRIMENTO E LARGURA. (B'-C') FUNÇÕES <i>BANDELET</i> [55].....	102
FIGURA 6.13 – DECOMPOSIÇÃO <i>WEDGELET</i> (A) IMAGEM ARTIFICIAL (B) DECOMPOSIÇÃO <i>WEDGELET</i> MULTI-ESCALA (C) PARAMETRIZAÇÃO DE UM <i>WEDGELET</i> EM UM QUADRADO $N \times N$ DIÁDICO: θ É A ORIENTAÇÃO ANGULAR DO CONTORNO, D É A DISTÂNCIA NORMALIZADA DO CONTORNO AO CENTRO DO BLOCO E H É A DIFERENÇA DE INTENSIDADES ENTRE CADA LADO DO CONTORNO. ...	104
FIGURA 6.14 – DECOMPOSIÇÃO <i>WEDGELET</i> DE UM TRECHO DE IMAGEM <i>CARTOON</i> (A) ESCALA ÚNICA (TAMANHO DOS BLOCOS FIXO) (B) MULTI-ESCALA [94].	104
FIGURA 6.15 – APROXIMAÇÕES EM <i>QUADTREE</i> (A) <i>QUADTREE</i> COM 967 NÓS CONSTANTES, PSNR: 33.69 DB. (B) <i>QUADTREE</i> COM 248 NÓS <i>WEDGELET</i> (496 COEFICIENTES <i>WEDGELET</i>), PSNR: 33.71 DB [99].	107

FIGURA 6.16 – ESTIMATIVA DO PARÂMETRO ANGULAR DO <i>WEDGELET</i> A PARTIR DE NÍVEIS SUPERIORES [85].....	108
FIGURA 6.17 – EXTENSÕES DAS <i>WEDGELETS</i> (A) <i>ARCLET</i> [101][102] (B) <i>PLATELET</i> [103].....	108
FIGURA 6.18 – EXEMPLO DE <i>SURFLETS</i> (A) $M = 2, K_d = [1,2]$ (B) $M = 2, K_d = [2,3]$ (C) $M = 3, K_d = [1,2]$ (D) $M = 3, K_d = [2,3]$ [104]	109
FIGURA 6.19 – CODIFICAÇÃO <i>WEDGELET-WAVELET</i> DE [46] (A) ORIGINAL (B) APROXIMAÇÃO <i>WEDGELETS</i> (<i>CARTOON</i>) (C) <i>CARTOON</i> APÓS FILTRAGEM (D) RESÍDUO CODIFICADO POR <i>WAVELETS</i> (<i>TEXTURA</i>) (E) IMAGEM RECONSTRUÍDA (0,4 BPP, PSNR = 27,35 dB).	110
FIGURA 6.20 – <i>WEDGELETS</i> NO DOMÍNIO <i>WAVELET</i> (A) PORÇÃO DE UMA IMAGEM CONTENDO UM <i>WEDGELET</i> EM UM BLOCO $S_{j,k}$ (B) <i>WEDGEPRINT</i> OBTIDO PELA PROJEÇÃO DO <i>WEDGELET</i> NA SUBBANDA VERTICAL DO DOMÍNIO <i>WAVELET</i>	111
FIGURA 6.21 – OBTENÇÃO DE UM SÍMBOLO <i>WEDGEPRINT</i>	112
FIGURA 6.22 – ÁRVORE DE DECOMPOSIÇÃO: S – TEXTURA SIGNIFICATIVA (QUANTIZAÇÃO ESCALAR CO COEFICIENTE <i>WAVELET</i>), Z: NÃO SIGNIFICATIVO (ÁRVORE DE ZEROS), W: CONTORNO SIGNIFICATIVO (ÁRVORE <i>WEDGELET</i>) (A) SFQ [106] (B) W-SFQ [107].....	113
FIGURA 6.23 – IMAGEM ‘ <i>CAMERAMAN</i> ’ E DIFERENTES TIPOS DE REGIÕES QUE ELA CONTÉM.	113
FIGURA 6.24 – (A) PORÇÃO DE UMA IMAGEM CONTENDO UM <i>SURFLET</i> EM UM BLOCO $S_{j,k}$ (B) <i>SURFPRINT</i> OBTIDO PELA PROJEÇÃO DO <i>SURFLET</i> NA SUBBANDA VERTICAL DO DOMÍNIO <i>WAVELET</i>	113
FIGURA 6.25 – EXTENSÕES AO MODELO DE MULTI-DICIONÁRIOS <i>CARTOON-TEXTURA</i> [94].	114
FIGURA 7.1 – MÉTODO PROPOSTO (WWB).....	116
FIGURA 7.2 – PROCESSO DE CODIFICAÇÃO-DECODIFICAÇÃO <i>WEDGELET</i>	116
FIGURA 7.3 – IMAGENS NATURAIS DA BASE DE DADOS WATERLOO [111] (A) LENA (B) <i>CAMERAMAN</i> (C) <i>PEPPERS</i> (D) <i>FRUITS</i> (E) BARBARA (F) <i>BOAT</i> (G) <i>CLOCK</i> (H) <i>BABOON</i>	117
FIGURA 7.4 – DESENHOS USADOS NOS TESTES (A) <i>CIRCLES</i> (B) <i>OBELIX</i> (C) BRASIL (D) ÁFRICA (E) <i>ROY</i> (F) UK (G) <i>COW</i> (H) UNICAMP.	118
FIGURA 7.5 – VARIAÇÃO DO NÚMERO DE ÂNGULOS TESTADOS NA DETERMINAÇÃO DO <i>WEDGE</i> (A) 2, (B) 3, (C) 4, (D) 6, (E) 8, (F) 16, (G) 18.....	119
FIGURA 7.6 – RESULTADOS DA VARIAÇÃO DO NÚMERO DE ÂNGULOS (A) IMAGENS NATURAIS (B) DESENHOS.	120
FIGURA 7.7 – VARIAÇÃO DO NÚMERO DE DISTÂNCIAS NA DETERMINAÇÃO DO <i>WEDGE</i> (A) 1, (B) 2, (C) 4, (D) 8, (E) 16.....	121
FIGURA 7.7 – RESULTADOS DA VARIAÇÃO DO NÚMERO DE DISTÂNCIAS (A) IMAGENS NATURAIS (B) DESENHOS.	122

FIGURA 7.8 – COMPARAÇÃO SUBJETIVA PARA A IMAGEM ‘CAMERAMAN’ A 1 BPP (A) ORIGINAL (A) <i>WEDGELET</i> DE 2 ÂNGULOS E 4 DISTÂNCIAS (B) <i>WEDGELET</i> DE 8 ÂNGULOS E 4 DISTÂNCIAS (C) <i>WEDGELET</i> DE 8 ÂNGULOS E 1 DISTÂNCIA.....	123
FIGURA 7.9 – RESULTADOS DA VARIAÇÃO DA FUNÇÃO <i>WAVELET</i> (A) IMAGENS NATURAIS (B) DESENHOS.....	125
FIGURA 7.10 – RESULTADOS DO MÉTODO WWB (A) IMAGENS NATURAIS (B) DESENHOS.....	127
FIGURA 7.11 – USO DE COMPONENTES <i>WEDGELET</i> DE ACORDO COM A TAXA (A) IMAGENS NATURAIS (B) DESENHOS.....	128
FIGURA 7.12 – CODIFICAÇÃO DA IMAGEM ‘CAMERAMAN’ A 0,1736 BPP (A) IMAGEM RECUPERADA PSNR = 23,42 dB (B) MAPEAMENTO DOS BLOCOS CODIFICADOS POR <i>WEDGELETS</i>	130
FIGURA 7.13 – CODIFICAÇÃO DA IMAGEM ‘CAMERAMAN’ A 0,6507 BPP (A) IMAGEM RECUPERADA PSNR = 30,49 dB (B) MAPEAMENTO DOS BLOCOS CODIFICADOS POR <i>WEDGELETS</i>	130
FIGURA 7.14 – CODIFICAÇÃO DA IMAGEM ‘PEPPERS’ A 0,2336 BPP (A) IMAGEM RECUPERADA PSNR = 23,7389 dB (B) BLOCOS CODIFICADOS POR <i>WEDGELETS</i> DESTACADOS EM VERMELHO.....	131
FIGURA 7.15 – CODIFICAÇÃO DA IMAGEM ‘PEPPERS’ A 0,3794 BPP (A) IMAGEM RECUPERADA PSNR = 26,0650 dB (B) BLOCOS CODIFICADOS POR <i>WEDGELETS</i> DESTACADOS EM VERMELHO.....	131
FIGURA 7.16 – CODIFICAÇÃO DA IMAGEM ‘BRASIL’ A 0,3431 BPP (A) IMAGEM RECUPERADA PSNR = 37,77 dB (B) MAPEAMENTO DOS BLOCOS CODIFICADOS POR <i>WEDGELETS</i>	132
FIGURA 7.17 – CODIFICAÇÃO DA IMAGEM ‘BRASIL’ A 0,5828 BPP (A) IMAGEM RECUPERADA PSNR = 51,79 dB (B) MAPEAMENTO DOS BLOCOS CODIFICADOS POR <i>WEDGELETS</i>	132
FIGURA 7.18 – COMPARAÇÃO SUBJETIVA DA IMAGEM ‘CAMERAMAN’ COM MÉTODOS <i>WAVELET</i> (A) ORIGINAL (B) WWB 0,3251 BPP PSNR = 26,5223 dB (C) <i>WEDGELETS</i> 0,3194 BPP 25,8670 dB (D) <i>WAVELETS</i> 0,3194 BPP PSNR = 25,6961 dB.....	133
FIGURA 7.19 – COMPARAÇÃO SUBJETIVA DA IMAGEM ‘PEPPERS’ COM MÉTODOS <i>WAVELET</i> (A) ORIGINAL (B) WWB 0,2336 BPP PSNR = 23,7389 dB (C) <i>WEDGELETS</i> 0,2404 BPP 22,6040 dB (D) <i>WAVELETS</i> 0,2337 BPP PSNR = 23,9179 dB.....	134
FIGURA 7.20 – COMPARAÇÃO SUBJETIVA DA IMAGEM ‘BRASIL’ COM MÉTODOS <i>WAVELET</i> (A) ORIGINAL (B) WWB 0,3431 BPP PSNR = 37,77 dB (C) <i>WEDGELETS</i> 0,3560 BPP 37,6629 dB (D) <i>WAVELETS</i> 0,3500 BPP PSNR = 32,6461 dB.....	135
FIGURA 7.21 – COMPARAÇÃO DA CODIFICAÇÃO <i>WEDGELET</i> DE [109] COM SÍMBOLOS BINÁRIOS E O MÉTODO <i>WEDGELET-WAVELET</i> PROPOSTO (A) IMAGENS NATURAIS (B) DESENHOS.....	136
FIGURA 7.22 – COMPARAÇÃO COM MÉTODOS <i>WAVELET</i> (A) IMAGENS NATURAIS (B) DESENHOS..	137
FIGURA 7.23 – COMPARAÇÃO SUBJETIVA DA IMAGEM ‘CAMERAMAN’ COM MÉTODOS <i>WAVELET</i> (A) ORIGINAL (B) WWB 0,3251 BPP PSNR = 26,5223 dB (C) SPIHT 0,3431 BPP PSNR = 27,0453 dB (D) JPEG 2000 0,3479 BPP PSNR = 29,1556 dB (E) JPEG 0,3347 BPP PSNR = 26,4713 dB.....	139

- FIGURA 7.24 – COMPARAÇÃO SUBJETIVA DA IMAGEM ‘BRASIL’ COM MÉTODOS *WAVELET* (A) ORIGINAL (B) WWB 0,3431 BPP PSNR = 37,77 DB (C) SPIHT 0,3432 BPP PSNR = 17,8789 DB (D) JPEG 2000 0,3477 BPP PSNR = 38,8455 DB (E) JPEG 0,3544 BPP PSNR = 33,2960 DB. 140
- FIGURA 7.25 - COMPARAÇÃO SUBJETIVA COM OUTROS MÉTODOS *WEDGELET-WAVELET* (A) ORIGINAL (B) WWB 0,3522 BPP PSNR = 26,6524 DB (C) WWB 0,6278 BPP PSNR = 30,3371DB (D) MÉTODO DE [46] A 0,4 BPP PSNR = 28,48 DB (E) MÉTODO DE [46] COM FILTRO POSTERIOR A 0,4 BPP PSNR = 27,35 DB. 142
- FIGURA 7.26 - COMPARAÇÃO SUBJETIVA COM OUTROS MÉTODOS *WEDGELET-WAVELET* (A) WWB 0,1519 BPP PSNR = 23,0640 DB (B) MÉTODO APRESENTADO EM [105] (W-SFQ) 0,146 BPP PSNR = 25,94 DB. 143

Lista de Tabelas

Tabela 2.1 - Usuários de serviços de telecomunicação no Brasil [7].....	35
Tabela 2.2 – Penetração dos serviços de telecomunicação no Brasil [7].....	35
Tabela 2.3 – Penetração dos serviços de telecomunicação no Brasil relativo ao número de domicílios [7].....	36
Tabela 7.1 – Tempo de processamento para imagens naturais, variando número de ângulos	120
Tabela 7.2 – Tempo de processamento para desenhos, variando número de ângulos.....	121
Tabela 7.3 – Tempo de processamento para fotografias, variando número de distâncias.....	123
Tabela 7.4 – Tempo de processamento para desenhos, variando número de distâncias	123
Tabela 7.5 – Tempo de processamento para imagens naturais, variando a função <i>wavelet</i>	125
Tabela 7.6 – Tempo de processamento para desenhos, variando a função <i>wavelet</i>	126
Tabela 7.7 – Tempo de processamento dos diferentes métodos para imagens naturais.....	129
Tabela 7.8 – Tempo de processamento dos diferentes métodos para desenhos	129

Lista de Abreviaturas

ISDB - *Integrated Services Digital Broadcasting Terrestrial*
 ATSC - *Advanced Television Systems Committee*
 DVB - *Digital Video Broadcasting*
 HDTV - *High Definition Television*
 JPEG - *Joint Photographic Experts Group*
 MPEG - *Moving Picture Experts Group*
 SAR - *Synthetic-Aperture Radar*
 SBTVD - *Sistema Brasileiro de TV Digital*
 GPRS - *General Packet Radio Service*
 Anatel - *Agência Nacional de Telecomunicações*
 QVGA - *Quarter Video Graphics Array*
 SDTV - *Standard Definition Television*
 LDTV - *Low Definition Television*
 SFN - *Single Frequency Network*
 OFDM - *Orthogonal Frequency-Division Modulation*
 QPSK - *Quadrature Phase Shift Keying*
 BML - *Broadcast Markup Language*
 BPSK - *Binary Phase Shift Keying*
 QAM - *Quadrature Amplitude Modulation*
 LDPC - *Low-Density Parity-Check*
 DMB - *Digital Multimedia Broadcasting*
 CMMB - *China Mobile Multimedia Broadcasting*
 MBMS - *Multimedia Broadcast Multicast Services*
 UMTS - *Universal Mobile Telecommunications System*
 GSM - *Global System for Mobile Communications*
 DCT - *Discrete Cosine Transform*
 JND - *Just Noticeable Difference*
 HVS - *Human Visual System*
 CSF - *Contrast Sensitivity Function*
 PSNR - *Peak Signal to Noise Ratio*
 SSIM - *Structural SIMilarity*
 TWD - *Transformada Wavelet Discreta*
 DFT - *Discrete Fourier Transform*
 KLT - *Karhunen-Loève Transform*
 TWC - *Transformada Wavelet Contínua*
 TF - *Transformada de Fourier*
 SW - *Série Wavelet*
 FWT - *Fast Wavelet Transform*
 QV - *Quantização Vetorial*
 SPIHT - *Set Partitioning In Hierarchical Trees*
 EZW - *Embedded Zerotree Wavelet*
 SFQ - *Space Frequency Quantization*
 W-SFQ - *Wedgelet - Space Frequency Quantization*
 WWB - *Wedgelet-Wavelet por Blocos*
 bpp - *bits por pixel*

Capítulo 1

Introdução

Os avanços em comunicações e equipamentos digitais (câmeras, celulares e impressoras), a disputa de mercados de TV digital pelos padrões ISDB-T, ATSC e DVB-T e principalmente a padronização do sistema de HDTV (*High Definition Television* – Televisão de Alta Definição) no Brasil, impulsionaram diversas áreas de pesquisa relacionadas à codificação e compressão de imagens. Nesse contexto, pesquisas acerca de novas técnicas para codificação têm recebido grande atenção por parte de diversos pesquisadores.

No âmbito da compressão de imagens e de vídeo, tais avanços são precedidos por transformadas ortogonais e biortogonais, e quantização de coeficientes. A transformada *wavelet* tem obtido grande eficiência em termos de qualidade visual e PSNR (*Peak Signal to Noise Ratio* – Relação Sinal Ruído de Pico), apresentando algoritmos rápidos, boa capacidade de concentrar a energia do sinal em poucos coeficientes, transmissão progressiva e ausência de artefatos de blocagem. Essas vantajosas características permitiram sua inserção em importantes padrões de compressão de imagem e vídeo, tais como o JPEG 2000 [1], MPEG-4 e MPEG-7[2].

Entretanto, imagens codificadas com a transformada *wavelet* apresentam artefatos característicos, principalmente *ringing* e embaçamento. Diversas pesquisas têm sido realizadas na área, utilizando quantização adaptativa e métodos híbridos, para tentar minimizar esses artefatos [3]. Propôs-se até mesmo a utilização de um novo espaço de funções, as chamadas *countourlets*, *curvelets*, *bandelets* e dicionários geométricos como *wedgelets* [4], unidos a técnicas de decomposição *cartoon-textura*. Essas novas técnicas serão descritas neste trabalho e comparadas com métodos em estado-da-arte baseados em *wavelets*.

Muitas aplicações que se utilizam da codificação/compressão de imagens, principalmente em casos em que se torna necessária a recuperação de bordas e de texturas, podem ser beneficiadas pelas técnicas de decomposição *cartoon-textura* e desses novos espaços de funções. Possíveis aplicações incluem avaliação objetiva de qualidade de imagens, modelamento estatístico e identificação de texturas, biometria (reconhecimento de face, íris e impressão digital), imagens médicas (tomografia, mamografia), sensoriamento remoto (imagens ópticas e de SAR – *Synthetic Aperture Radar* [5]), compressão de vídeo (HDTV), desenho animado [6], entre outros.

Nesse contexto, este trabalho propõe um novo método de compressão híbrido *wedgelet-wavelet* dentro da metodologia *cartoon-textura*, objetivando ganho de desempenho por meio da exploração da geometria das imagens.

1.1 – Contribuições e Organização do Trabalho

Este trabalho apresenta as principais bases do dicionário geométrico *wedgelet* e sua aplicação em compressão de imagens. É definido um esquema simples de codificação sob a forma de *quadtree* e os melhores parâmetros *wedgelet* são definidos de forma a minimizar o erro quadrático médio das imagens reconstruídas.

Logo, propõem-se como contribuições inéditas deste trabalho:

- Otimização dos parâmetros do dicionário *wedgelet* para codificação de imagens, buscando minimizar erro objetivo.
- Modificação no algoritmo SPIHT de codificação usando *wavelets* para otimizar a relação taxa-distorção ao invés de utilizar taxa de dados pré-fixada.
- Criação de um método de compressão híbrido *wedgelet-wavelet*, que escolhe o melhor dicionário para cada região da imagem, de acordo com a relação taxa-distorção e a característica *cartoon* ou textura da imagem.

São feitas comparações com algoritmos baseados em transformada *wavelet* e *wedgelet* puros; um método híbrido, utilizando *wedgelets* e *wavelets*, de forma a preservar componentes de *cartoon* e textura, é proposto. Os ganhos principalmente em qualidade subjetiva das imagens reconstruídas para baixas taxas de dados indicam que a metodologia apresentada poderia ser incluída em sistemas de transmissão de dados com restrições de taxa, como por exemplo a TV digital móvel. A dissertação está organizada de acordo com os seguintes capítulos:

Capítulo 2 Comunicações Móveis de Imagens e Vídeo: Por se tratar de um campo de muitas inovações tecnológicas recentes interessantes à presente proposta, aplicações envolvendo o ambiente de baixas taxas de transmissão são apresentadas. O ambiente de transmissão móvel é muito restrito e apresenta limitações na banda de transmissão, que se traduz em baixas taxas de dados, menores que 1 bit por pixel. É dada atenção especial à TV digital móvel, o panorama geral e sua relação com a TV digital de alta definição. São incluídas descrições dos padrões mundiais mais recentes, principalmente em relação à codificação de imagens e vídeo.

Capítulo 3 Qualidade Visual: Para o desenvolvimento de codificadores de imagem é muito importante a compreensão dos aspectos relevantes ao observador. Nesse capítulo são apresentadas características da visão humana, e sua relação com os métodos de compressão. Os padrões JPEG e

JPEG 2000, estado-da-arte, são apresentados e os artefatos produzidos por eles são discutidos.

Capítulo 4 Metodologia *Cartoon-Textura*: Os métodos de codificação de imagem presentes na literatura se baseiam em técnicas matemáticas distintas, gerando artefatos visuais para imagens que não correspondem às suas bases de codificação. A metodologia *cartoon-textura* busca unir a potencialidade de 2 métodos em codificar conteúdos distintos: geometria bem definida e variação limitada (*cartoon*) e padrões oscilatórios (*textura*).

Capítulo 5 Transformada *Wavelet*: A transformada *wavelet* é considerada estado-da-arte para a análise/codificação de imagens, principalmente por sua capacidade de isolar componentes em espaço e frequência conjuntamente, além de apresentar algoritmos rápidos e transmissão progressiva, destacando-se como aplicação o padrão de compressão JPEG 2000.

Capítulo 6 Novas Formas de Representação Esparsa: Embora as *wavelets* tenham um ótimo desempenho na compressão de imagens fotográficas, elas falham em representar de forma econômica informações geométricas como contornos de uma imagem *cartoon*. Em baixas taxas de dados, essa ineficiência se traduz em artefatos do tipo *ringing*. Para superar essa deficiência, novas formas de representação são apresentadas, com destaque especial para o dicionário *wedgelet*, adotado nesta proposta.

Capítulo 7 Proposta e Análise dos Resultados Experimentais: Nesse capítulo são encontrados os melhores parâmetros *wedgelet* para a compressão de imagens e é proposto um método híbrido que utiliza nós *wedgelet* e nós *wavelet* de forma a recuperar características *cartoon* e *textura*. Os resultados do codificador híbrido são comparados com os resultados de sistemas-base *wedgelet* “puros” e com o sistema-base *wavelet* “puro”. Os resultados obtidos foram analisados quanto ao tempo de codificação, PSNR, e qualidade subjetiva para imagens fotográficas conhecidas na literatura científica e para imagens com característica geométrica acentuada, como desenhos.

Capítulo 8 Conclusões e Sugestões para Trabalhos Futuros: Finaliza-se apresentando comentários e conclusões acerca dos resultados obtidos, analisando-se as perspectivas de melhorias e extensões do procedimento proposto. Espera-se, por meio deste trabalho e das sugestões aqui apresentadas, estimular o uso da compressão híbrida *cartoon-textura* usando o dicionário de *wedgelets*, bem como outros dicionários de representação esparsa.

Capítulo 2

Comunicações Móveis de Imagens e Vídeo

Os avanços na área de compressão de imagem e vídeo têm gerado uma ampla gama de aplicações para indústrias e serviços nas mais diversas áreas de atuação humana, envolvendo desde a disputa acirrada mundial por mercados de TV digital (*broadcasting*), comunicações via satélite, imagens médicas, até produtos de acessibilidade. Nesse universo de áreas de atuação, destacam-se as aplicações para dispositivos móveis (celulares, *iPhones*, *iPads*, entre outros). Novas técnicas que sirvam como “ferramentas” para tratamento e compressão de imagem e vídeo vêm experimentando uma explosão de interesse acadêmico e comercial.

Uma gama de novas aplicações e formas de comunicação vem surgindo para competir com a televisão tradicional. Com isso os telespectadores se tornaram mais exigentes, principalmente quanto à qualidade da imagem a ser assistida. Independente do tamanho da tela, exige-se cada vez mais definição na imagem, a custos menores e transmissões mais rápidas. Os equipamentos se desenvolveram a ponto de permitirem a codificação e decodificação de vídeo mesmo na linha mais básica de *smartphones*. Essas tendências devem ser levadas em conta ao se desenvolver um algoritmo de processamento voltado para multimídia.

O ambiente de comunicações móveis, no entanto, é muito mais restrito do que as aplicações fixas. A largura de banda disponível para a transmissão de dados geralmente é reduzida para dar conta de atender toda a demanda, além de ser um canal muito ruidoso. Para superar essas restrições, torna-se fundamental o trabalho da área de processamento digital de sinais em codificar os dados com atenção para baixas taxas de transmissão.

Muitas aplicações multimídia já são possíveis hoje com dispositivos móveis disponíveis no mercado, desde comunicação ponto-a-ponto (envio de imagens, videofone,...) a conteúdos personalizados por operadoras ou *broadcast*. A TV digital móvel é uma aplicação que apresenta muito potencial, dadas às novas oportunidades de negócios para conquistar usuários que passam muito tempo em trânsito, ou que apenas requerem maior personalização.

Neste capítulo será abordada a importância da TV digital, sua evolução no país e um enfoque

nas aplicações de TV digital móvel, área muito promissora e que tem potencial de se beneficiar da metodologia *cartoon-textura* que será apresentada (capítulo 4) e do dicionário *wedgelet* (capítulo 6), devido às baixas taxas de codificação.

2.1 Televisão Digital no Brasil

A televisão é um dos bens de consumo mais presente. Cerca de 95% da população possui um aparelho de TV em casa [7]. A programação transmitida pelas emissoras tem como objetivo informar, entreter, definir padrões de consumo e até mesmo orientar a agenda dos expectadores. Por ser um meio de comunicação passivo e simples, é usado por todas as faixas etárias e classes sociais. No contexto comercial não há como ignorar o alcance das chamadas publicitárias na venda de produtos e difusão de marcas. Além de um meio de comunicação e cultura estratégico, a televisão é também um grande propulsor de inovação tecnológica e desenvolvimento industrial.

Os estudos em transmissão de televisão digital no Brasil começaram no final da década de 1990 com o trabalho da Comissão Brasileira de Comunicações CBC.2 da Agência Nacional de Telecomunicações (Anatel). Em 2003, a presidência da república assinou o Decreto n.º 4.901, que criou o Sistema Brasileiro de TV Digital Terrestre, ou SBTVD. O padrão adotado no Brasil foi o ISDB-T_B, uma adaptação do padrão japonês ISDB-T, escolhido por atender melhor as necessidades de energia nos receptores, mobilidade e portabilidade sem custo para o consumidor.

As transmissões comerciais com o padrão escolhido iniciaram em 2007 em São Paulo. Até maio de 2011, o sistema também havia sido lançado em 480 cidades brasileiras [8]. De acordo com anúncio do governo brasileiro, a transmissão analógica será encerrada em 2016, quando todas as emissoras transmitirão apenas o sinal digital no Brasil.

A implantação da transmissão digital com o formato de alta definição (HDTV – *High Definition Television*) é um fator adicional para a inovação e competitividade dos produtos. Para as emissoras, dentre as principais diferenças entre a TV analógica e a digital se destacam:

- O sinal digital requer menos potência, garantindo melhor eficiência na transmissão.
- Como pode ser recebido por diferentes dispositivos, abrange maior público.
- Por meio da interatividade, pode optar por novas formas de publicidade, ou mesmo de vendas se tiver disponível um canal de retorno.
- Com o canal de retorno é possível também obter dados do usuário, como estatísticas de audiência, em tempo real.

Para comparação, a resolução da televisão analógica, por ter perdas na transmissão pelo ar, chega a no máximo 333×480 pixels. Na transmissão digital, com o codificador de vídeo H.264, é possível transmitir até 2 canais *full-HD* (1080i), 4 canais *half-HD* (720p) ou 8 SD (480p) pela

mesma transmissora. No entanto, de acordo com a portaria "Nº 24, fevereiro de 2009, Norma Geral para Execução dos Serviços de Televisão Pública Digital do Ministério das Comunicações", só é permitido transmitir um programa em diferentes formatos, simultaneamente. A multiprogramação é permitida apenas a canais do governo federal. [9]

As principais inovações do ISDB-T_B em relação ao padrão japonês, foram: [10]

- Codificação de áudio e vídeo H.264 em diferentes níveis e perfis. Para transmissão fixa, H.264-HP@L4.0, formatos 480i, 480p, 720p e 1080i, taxas de 25, 30, 50 e 60 quadros/segundo. A compressão de áudio usa MPEG-4 AAC 48 kHz, nos perfis e níveis AAC@L4 (som estéreo) e HE-AAC@L4 (som 5.1).
- Possibilidade de atualização de *software* e controle remoto com interatividade.
- Normas para transmissão de conteúdo seguro.
- Canal de retorno pode usar qualquer plataforma (*ethernet, modem, Wi-fi, GPRS*).

2.2 A Televisão Digital Móvel

Os dispositivos de comunicações móveis têm apresentado adoção muito acelerada no Brasil. Segundo a Anatel [11], existem 153,67 milhões de usuários de telefonia móvel no Brasil, com um crescimento mensal de 1,3 milhões de assinantes. Em média, é feita a troca de aparelho celular a cada 12 meses [14]. Segundo a Anatel [12], a penetração da tecnologia celular era de 68,2 % em 2008 e passou a 82,44 % em 2009. Outro estudo promovido pelo Yankee Group contesta, afirmando ser em torno de 50%, desconsiderando usuários que utilizam vários *chips* promocionais [13]. Mesmo assim, os números chegam a ser muito significativos considerando os altos custos do serviço. Como mostram as Tabelas 2.1 a 2.3, os serviços de comunicações móveis são o segmento que mais cresce, perdendo em penetração apenas para os serviços de rádio e TV fixa (aberta).

Tabela 2.1 - Usuários de serviços de telecomunicação no Brasil [7]

Milhões	2003	2004	2005	2006	2007	2008
Telefones fixos	39,2	39,6	39,8	38,8	39,4	41,3
Celulares	46,4	65,6	86,2	99,9	121,0	150,6
TV por assinatura	3,6	3,9	4,2	4,6	5,3	6,3
Banda larga	1,2	2,3	3,9	5,7	7,7	10,0
Usuários de <i>internet</i>	-	-	32,1	35,3	44,9	53,9

Tabela 2.2 – Penetração dos serviços de telecomunicação no Brasil [7]

Densidades por 100 hab.	2003	2004	2005	2006	2007	2008
Telefones fixos	21,8	21,7	21,5	20,8	20,9	21,7
Celulares	25,8	35,9	46,6	53,2	63,6	78,1
TV por assinatura	2,0	2,1	2,3	2,5	2,8	3,3
Banda larga	0,7	1,3	2,1	3,1	4,1	5,3

Tabela 2.3 – Penetração dos serviços de telecomunicação no Brasil relativo ao número de domicílios [7]

% Domicílios com	2001	2002	2003	2004	2005	2006	2007
Rádio	88,0%	87,9%	87,8%	87,8%	88,0%	87,9%	88,1%
Televisão	89,0%	90,0%	90,1%	90,3%	91,4%	93,0%	94,5%
Telefone (fixo ou celular)	58,9%	61,7%	62,0%	65,4%	71,6%	74,5%	77,0%
Microcomputador	12,6%	14,2%	15,3%	16,3%	18,6%	22,1%	26,6%
Microcomputador com acesso à <i>internet</i>	8,60%	10,3%	11,5%	12,2%	13,7%	16,9%	20,2%
Total de domicílios (milhares)	46.507	48.036	49.712	51.753	53.114	54.610	56.344

Nota: Até 2003, não inclui área rural de Rondônia, Acre, Amazonas, Roraima, Pará e Amapá.

Observando esses dados, é inevitável a previsão de que o celular irá se unir a televisão, aumentando ainda mais a cobertura atendida por esses serviços. A Figura 2.1 mostra a tendência de crescimento do número de usuários de televisão móvel em todo o mundo. Com a TV integrada ao celular por meio da tecnologia digital, se torna mais acessível à produção e distribuição de conteúdo audiovisual feito localmente, para garantir empregos, gerar conteúdo e proteger a identidade nacional. [15]

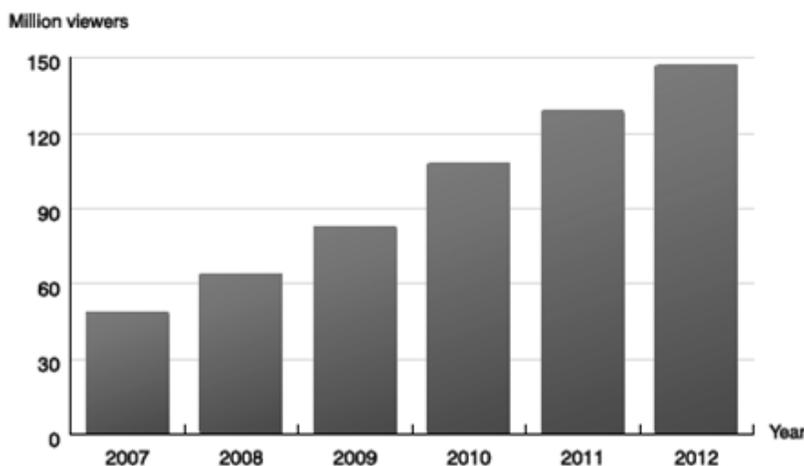


Figura 2.1 – Previsão do número de telespectadores de TV móvel no mundo entre 2007 e 2012 [16]

2.3 Padrões de TV Digital Móvel

A resolução de imagem em dispositivos móveis é próxima do formato QVGA (320×240). Como apontado em [17], um serviço de vídeo simples tipicamente pode utilizar de seis a dez vezes a banda dos canais da rede de telefonia móvel, o que levaria, quando os serviços de multimídia se tornarem mais populares, a rede a "travar". A solução mais viável seria a transmissão de TV móvel por um padrão específico.

O padrão brasileiro permite 3 tipos de conteúdo: HDTV (alta definição), SDTV (definição padrão) e LDTV (baixa definição, adequado para transmissão móvel e portátil). As transmissoras podem transmitir simultaneamente conteúdos divididos nos 13 segmentos dentro do espectro

OFDM (Figura 2.2). O 1SEG (seção 2.3.2) é uma tecnologia que permite a recepção móvel gratuita, tanto em celulares quanto em veículos em movimento, uma vez que o sinal para os receptores móveis trafega no mesmo canal que a transmissão em alta definição. Em contrapartida, outros sistemas para recepção de TV móvel como padrão europeu (DVB-H), e o MediaFLO, exigem um canal adicional somente para TV portátil, gerando a necessidade de licitações adicionais de canais e, conseqüentemente, impondo tarifação ao consumidor final (não existe receita publicitária para arcar com custos de transmissão como na TV aberta).

Embora os padrões estabeleçam métodos para a modulação, codificação de canal e codificação de fonte, o enfoque deste trabalho está especificamente na compressão dos sinais de imagem e vídeo. Os codificadores estabelecidos pelos padrões servem como base de comparação para novas propostas, que podem vir a integrar futuros sistemas de TV digital e outras aplicações de transmissão multimídia.

2.3.1 DVB-H (*Digital Video Broadcasting*)

O DVB-H [18] é uma adaptação do padrão europeu DVB-T de televisão digital terrestre para o ambiente móvel. Pode ser usado como uma rede independente ou como canal de descida para outras redes móveis.

É utilizada a tecnologia de divisão de tempo (*time slicing*) para reduzir o consumo de bateria nos dispositivos. Dados são transmitidos sob a forma de pacotes IP. Cada pacote pode conter até 2 Mbits de dados (incluindo paridade). Existem 64 *bits* de paridade para cada 191 *bits* de dados, usando um código corretor de erros Reed-Solomon.

Dados adicionais são usados no *bitstream* para identificar a célula. É adicionado também o modo 4K e um entrelaçador (*interleaver*) mais eficiente para melhor recepção em movimento. O sistema é mais fácil de ser usado em uma rede de frequência única (SFN – *Single Frequency Network*) Parâmetros de modulação e codificação de fonte são ajustados pelas operadoras. Exemplo:

- Portugal: formato QCIF, AVC coding (H.264)
- França: H263, *bitrate* = 270 kbit/s (vídeo+áudio)

2.3.2 ISDB (*Integrated Services Digital Broadcasting*) – 1SEG

No padrão ISDB-T [10], cada canal ocupa uma banda de 6 MHz na faixa de UHF, dividida em 13 segmentos de 428 kHz cada. O 1SEG usa um desses segmentos para enviar um *transport stream*.



Figura 2.2 – Divisão do espectro OFDM de 13 segmentos do padrão ISDB-T_B

Assim como o ISDB-T, o 1SEG usa modulação QPSK com código corretor de erros 2/3 e intervalo de guarda 1/4. A taxa total de dados é 416 kbits/s. Para a codificação de vídeo é usado o H.264 *Baseline Profile* com resolução de 320×240 *pixels* (formato 4:3) ou 320×180 *pixels* (formato 16:9) com *bitrate* entre 220 a 320kbit/s, para áudio é usado AAC-HE v2 com *bitrate* entre 48 e 64kbit/s. Dados adicionais são transmitidos usando o padrão BML (*Broadcast Markup Language*) com 10 a 100 kbits/s. Há, porém, modificações feitas para o padrão brasileiro. Enquanto o 1SEG japonês possui taxa de quadros máxima de 15fps para transmissões, o máximo no padrão brasileiro é 30fps. Além disso, o padrão brasileiro utiliza uma versão mais nova do codificador de áudio (HE-AAC v.2 *low complexity*). Esse é um dos fatores que impedem que receptores 1SEG japoneses recebam adequadamente transmissões 1SEG brasileiras. O alcance do sinal é de 40 km da antena transmissora, à medida que o sinal 1SEG é muito robusto. O uso no *desktop/notebook* requer um processamento de no mínimo 1,4GHz, pois ele utiliza muita compressão na transmissão.

2.3.3 DMB (*Digital Media Broadcasting*)

DMB [19] é uma tecnologia sul-coreana, parte de um projeto de tecnologia de informação para transmissão de multimídia para receptores móveis, com o objetivo de substituir o rádio FM, tendo sido testada por outros países, como Alemanha, França, China, Holanda e Canadá. O primeiro serviço oficial de TV móvel surgiu em 2005. Há especificações para transmissão via satélite (S-DMB) e terrestre (T-DMB). Desde 2006 na Coreia do Sul o serviço é usado por 7 emissoras de TV, 12 emissoras de rádio e 8 canais de dados. Os canais são multiplexados em 6 portadoras ocupando as faixas 8 e 12 do espectro VHF. Para a codificação de vídeo utiliza-se MPEG-4 *Part 10* (H.264) e MPEG-4 *Part 3* BSAC ou HE-AAC V2 para o áudio.

2.3.4 CMMB (*China Multimedia Mobile Broadcasting*)

Padrão implantado na China pelo SARFT (*State Administration of Radio, Film, and Television*), é comparável ao padrão DVB-SH: um sistema híbrido que suporta rotas via terrestre ou via satélite (*gap-fillers*). O sinal irradiado consiste em 39 canais de serviço e 1 canal de controle, divididos em *time slots* (TDM). Utiliza multiplexação OFDM com modulações BPSK, QPSK e 16-QAM. Na codificação de canal utiliza Reed-Solomon concatenado com LDPC e entrelaçadores. Para a codificação de fonte é usado MPEG-4 AVC (H.264) para vídeo e AAC para áudio. Não é

possível encontrar muitas informações sobre o padrão, pois só é publicado em partes e sob o selo de “confidencial”. [20]

2.3.5 MobiTV Inc.

O MobiTV é um serviço pago de recepção de conteúdo via redes sem fio já existentes. Surgiu em 1999 e está disponível nos EUA pelas operadoras T-Mobile, Verizon Wireless, Sprint, AT&T Mobility, Alltel, Cellulsouth, U.S. Cellular, nTelos Wireless, e no Canadá pela TELUS Mobility. A companhia afirma ter mais de 10 milhões de assinantes e oferecer conteúdo ao vivo e gravado de muitas redes populares. [21] Segundo a descrição disponível no site oficial [21], suporta todo tipo de formato de entrada e combinações de formatos de saída, resoluções e taxas, além de controle contra pirataria (DRM – *Digital Rights Management*).

2.3.6 MediaFLO (*Foward Link Only*)

A Qualcomm em conjunto com as operadoras Verizon e AT&T lançou o serviço MediaFLO, em 2007 e 2008, respectivamente. Suporta pelo menos 20 canais de *video streaming* QVGA (240x320 *pixels*), dez canais de áudio estéreo e mais de 800 minutos de pequenos vídeos armazenados chamados *clipcasting*. O padrão também consome pouca potência - 4 horas em média para uma bateria de 850 mA. [22]

Utiliza a faixa de frequências 716-722 MHz que estava alocada para o canal UHF 55. A transmissão utiliza OFDM com modulação QPSK e 16-QAM em uma banda de 5,55 MHz. A taxa de dados é de 200-250 kbit/s. Em 2010, a Qualcomm, empresa responsável pelo sistema, anunciou que estaria suspendendo a venda para novos consumidores. [23]

2.3.7 MBMS (*Multimedia Broadcast Multicast Service*) – UMTS

O MBMS é um serviço de envio *broadcast* de conteúdo via redes celulares GSM e UMTS já existentes, oferecendo uma taxa de até 256 kbit/s. Tem a grande vantagem de a infra-estrutura de rede já existir para os operadores de rede móvel, assim a implantação pode ser mais barata em comparação com a construção de uma nova rede para os serviços. A capacidade de transmissão permite um número ilimitado de usuários com carga de rede constante. Além disso, também permite a possibilidade de transmitir informações simultaneamente para muitos assinantes de celular (*Broadcast / Multicast*), como mostrado na Figura 2.3.

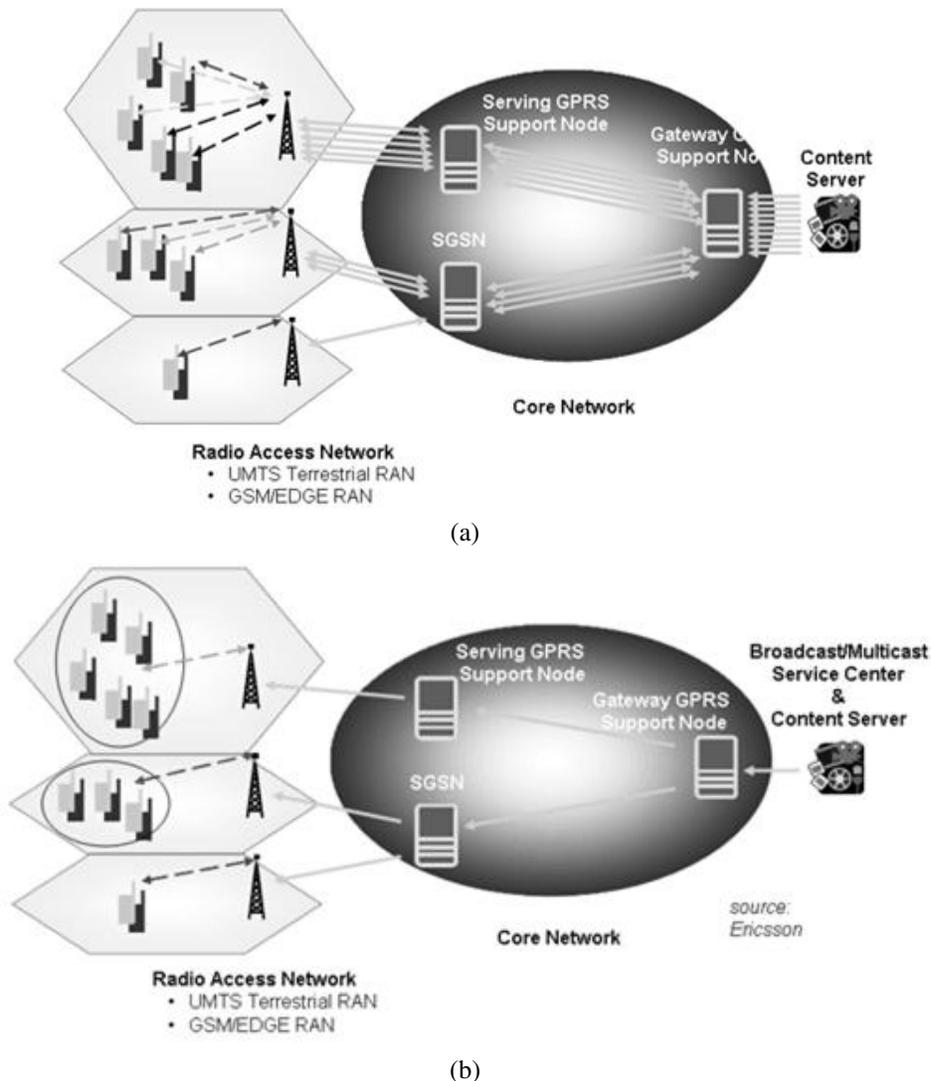


Figura 2.3 – Modos de transmissão UMTS (a) *unicast* (ponto a ponto) (b) *broadcast / multicast* (*serving*-servidora; *support*-suporte; *content*-conteúdo; *service*-serviço; *network*-rede; *core*-núcleo; *access*-acesso; *broadcast*-radiodifusão; *gateway*-entrada ; *terrestrial*-terrestre)

2.4 Comentário

Como mostrado, os padrões de TV digital móvel utilizam métodos de codificação com taxas muito baixas, para se adequar aos requisitos de largura de banda e imunidade a ruídos. Na maior parte dos casos, como por exemplo no padrão nipo-brasileiro 1SEG, a taxa é menor que 1 *bit* por *pixel*. Os algoritmos utilizados para a compressão são derivados do MPEG-2 ou MPEG-4/H.264, em *profiles* mais simples, que não utilizam estágios de codificação de entropia avançados ou predição, e usam a transformada discreta de cossenos (DCT), que tem desempenho inferior às transformadas *wavelet* (capítulo 5). Isso compromete a qualidade da imagem recebida pelo usuário, gerando artefatos visuais característicos, como será mostrado no capítulo 3.

Capítulo 3

Qualidade Visual

Neste capítulo serão abordados conceitos básicos da fisiologia da visão humana para determinar os requisitos de qualidade de imagem necessários para os algoritmos de codificação. O sistema visual humano opera de forma complexa para realizar a tarefa de gerar informação para o cérebro a partir da luz incidente no olho. Esse sistema é insensível a pequenas variações no estímulo visual (*JND – Just Noticeable Difference*), o que garante certa folga nos sistemas de codificação de imagem e vídeo. Diferenças na imagem reconstruída podem ser toleráveis dependendo da aplicação em que a imagem está submetida.

Os métodos em estado-da-arte são em sua maioria baseados em decomposições do sinal amostrado por transformadas ortogonais isotrópicas. Essa abordagem clássica apresenta erros que se traduzem em artefatos visuais característicos quando em ambientes de baixa taxa de *bits*, como é o caso da TV digital. Essas imperfeições e métodos para contorná-las, por meio da metodologia *cartoon-textura*, serão introduzidos no capítulo 4.

3.1 Sistema Visual Humano

O sistema visual humano (HVS – *Human Visual System*) é composto basicamente por órgãos receptores (olhos), uma unidade de processamento (cérebro) e por nervos que transmitem informação dos olhos para o cérebro. A Figura 3.1 ilustra a seção horizontal do olho humano.

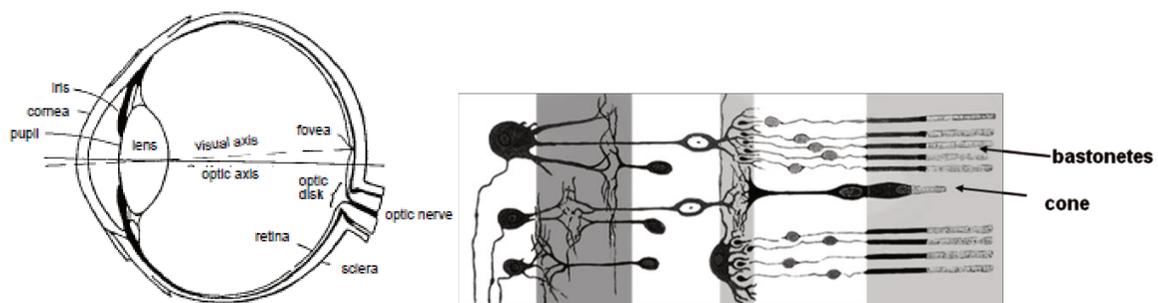


Figura 3.1 – Olho humano (a) seção reta horizontal [24] (b) células receptoras [26].

A luz atinge inicialmente a córnea, que tem a função de focalizar a luz incidente, e em seguida, passa por um líquido transparente, denominado humor aquoso [25]. Uma parcela da luz incidente é

barrada pela íris, uma musculatura que pode aumentar ou diminuir sua abertura, chamada de pupila, de acordo com a necessidade de entrada de luz. Em seguida, a luz atravessa o cristalino, que determina a focalização do olho. Por meio dos músculos ciliares pode-se alterar seu raio de curvatura, possibilitando aumentar a distância focal. A luz atravessa o humor vítreo e chega à retina, uma membrana ocular interna que possui receptores de sinais luminosos, os cones e bastonetes (Figura 3.1), que geram impulsos nervosos.

Na vizinhança do nervo óptico não há cones e bastonetes, sendo portanto desprovido de visão e conseqüentemente, possuindo um ponto cego nessa região [25]. Essa região é totalmente ocupada pelo feixe de fibras, sendo que dessa forma, os objetos de uma cena que são focalizados no ponto cego não são vistos efetivamente. A visão humana completa a informação luminosa correspondente ao ponto cego com a informação de regiões vizinhas [27].

Um fato muito importante para a modelagem de sistemas de processamento de imagem é que o sistema visual é mais sensível ao contraste (variação local de luz) do que ao valor absoluto de iluminação. A função de sensibilidade a contraste (CSF – *Contrast Sensitivity Function*) caracteriza a resposta em frequência do HVS. Conforme a Figura 3.2, a CSF exibe um comportamento de um filtro passa-faixa.

Para obter a CSF são feitos experimentos com estímulos senoidais em frequências determinadas. As frequências espaciais são dadas em ciclos por grau do ângulo visual, logo depende da distância entre o observador e a imagem. Quando o observador se afasta, a frequência aumenta, movendo-se a direita na curva de CSF, requerendo maior contraste para ser percebido. A partir dessa observação, são definidos os tamanhos de tela e resoluções necessários para serem observados detalhes. A resposta em frequência é importante também para melhorar a quantização das componentes em frequência sem causar impacto perceptível ao usuário.

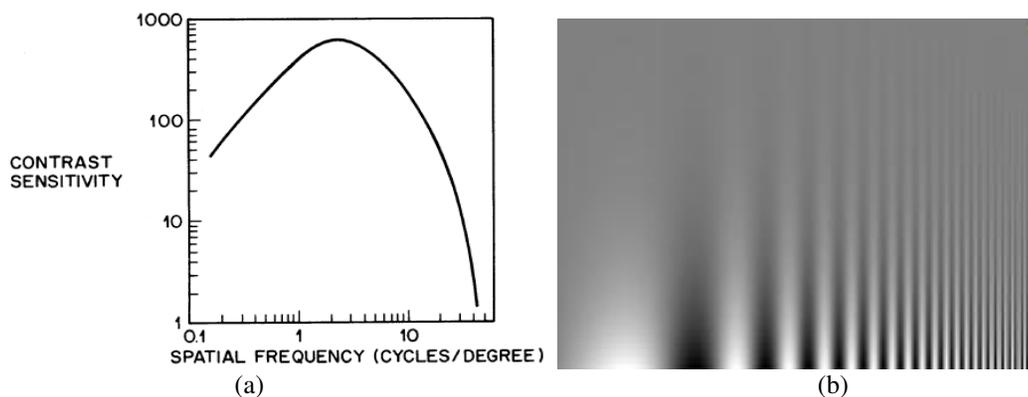


Figura 3.2 – Modelamento do HVS (a) Função de sensibilidade a contraste CSF (*contrast sensitivity-sensibilidade ao contraste; spatial frequency-frequência espacial; cycles/degree- ciclos/grau*) (b) Diagrama logarítmico de estímulos senoidais [28].

A explicação para o comportamento passa-faixa é que as células receptoras sozinhas do HVS

não definem a informação, mas atuam em conjunto. Seus estímulos são combinados, como em um filtro. Estímulos de mesma intensidade (baixo contraste) se anulam e estímulos com distância espacial menor do que o espaçamento entre os receptores não são percebidos, sendo aproveitados apenas estímulos de frequência média, que casam bem com os momentos do filtro.

Com relação à direcionalidade, as células receptoras são dispostas circularmente, oferecendo uma resposta aproximadamente isotrópica (Figura 3.3). A resposta visual é muito pronunciada para regiões entre 2 bordas de mesma direção (Figura 3.3.a), mas pode gerar distorções em regiões de encontro de bordas com diferentes direções (Figura 3.3.b), como é ilustrado na chamada grelha de Hermann. Essa disposição ilustra a importância da definição da direcionalidade das bordas na visão humana.

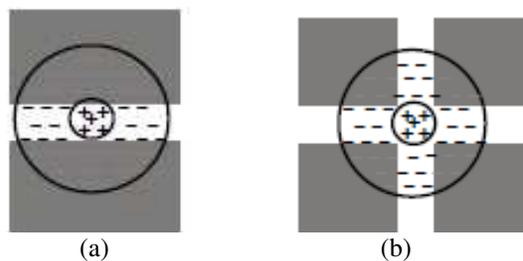


Figura 3.3 – Resposta direcional das células receptoras, interseção entre: (a) 2 blocos (b) 4 blocos [29].

As imagens processadas digitalmente normalmente apresentam artefatos de degradação devido às baixas taxas de *bits* disponíveis. Com a baixa resolução, a compreensão da informação pelo usuário pode ficar ainda mais dificultada (erros mais evidentes). Essa percepção varia de acordo com cada indivíduo e com a imagem. Métricas objetivas de qualidade, como PSNR e SSIM [30] não são adequadas para determinar artefatos específicos, sendo necessários estudos subjetivos, como em [31], onde é feito um estudo especificamente para determinar efeitos de qualidade em ambientes móveis, com 135 voluntários. Nesses estudos e também em [32] e [33] foi apontado que a avaliação de qualidade é influenciada pelo tipo de conteúdo [34]. O tipo do conteúdo determina a velocidade do movimento, quantidade de cores, regiões de interesse, brilho e contraste. É interessante notar que quando o usuário desconhecia o conteúdo (não reconheceu), as notas da avaliação objetiva foram superiores a quando o conteúdo era familiar. A Figura 3.4 mostra imagens típicas para TV móvel.

Em [35], [36] e [37] são realizados estudos a fim de caracterizar a percepção de conteúdos em baixa resolução em telas pequenas. Foram detectadas baixas avaliações por parte dos usuários para resoluções menores que 168×126 *pixels*, independente do conteúdo. Foi apontado que a legibilidade de textos é um fator de grande influência e deveria ser tratado de forma separada da imagem e renderizado pelo receptor (similar ao problema da geometria das imagens tratado nesta dissertação; ver capítulo 4).

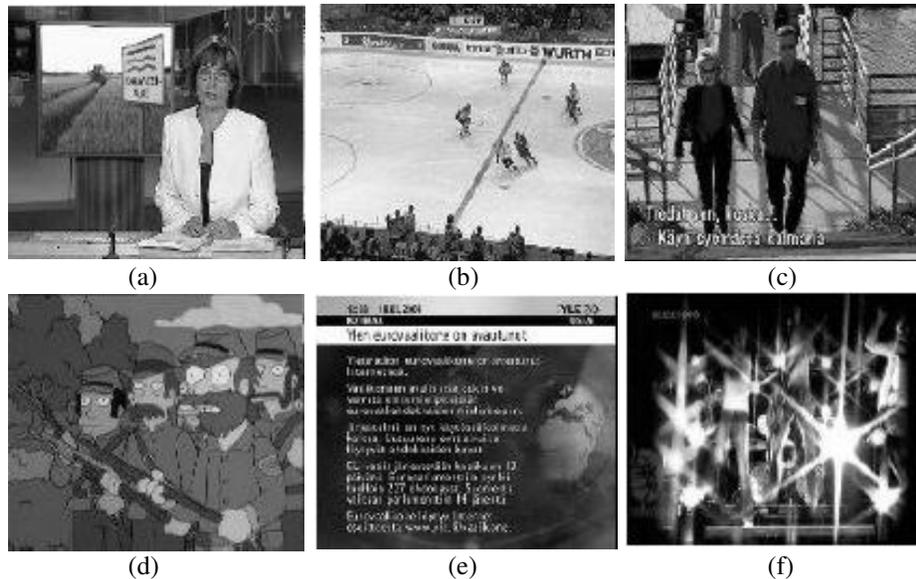


Figura 3.4 - Diferentes tipos de conteúdo para TV móvel [31] (a) notícias (b) esportes (c) séries (d) desenhos animados (e) teletexto (f) *clipe* musical

3.2 Codificação por Transformadas Ortogonais

De acordo com a teoria de Shannon para codificação de fonte [38], existe um código que representa uma dada informação a um tamanho médio mínimo, dado pela sua entropia. A entropia é uma medida introduzida por Shannon e caracteriza a incerteza de uma grandeza ou variável aleatória, e pode ser bem definida para dados digitais como imagens.

Embora de grande aplicação, os codificadores de entropia são muito genéricos e, quando sozinhos, não garantem grande compressão. Para melhores resultados em compressão de imagens e vídeo, são precedidos por transformadas ortogonais e biortogonais, e quantização (compressão com perdas – *lossy*).

No codificador, o estágio de transformadas é usualmente seguido por um estágio de quantização, que remove informações sem influência significativa na imagem reconstruída, tornando a reconstrução da imagem inexata. O nível de compressão e a perda na qualidade dependem do número de níveis discretos produzidos pela quantização. Dentre as transformadas usadas para a codificação de imagens, destacam-se a DCT e a TWD.

Nas aplicações, a DCT é normalmente operada sobre blocos de imagem de tamanho fixo 8×8 ou 4×4. Essa divisão em blocos permite que os dados de entrada possuam uma maior correlação interna (facilitando a codificação entrópica e eliminação de redundâncias). No entanto, a quantização concentrada em blocos leva ao surgimento do chamado “efeito de bloco”, uma descontinuidade entre os blocos.

O padrão mais utilizado para a compressão de imagens foi durante anos o JPEG, para imagens

estáticas, e as variações do MPEG (MPEG-1, MPEG-2 e MPEG-4) para vídeo. Ambos os esquemas se utilizam da DCT como bloco principal. Padrões mais modernos têm tentado resolver os problemas inerentes ao uso da DCT (como o efeito de bloco) e garantir melhor qualidade visual em relação à taxa de *bits* (*bitrate*).

O H.264 ou MPEG-4 Part 10 AVC [40] é um padrão de compressão de vídeo mais moderno, que consegue taxas de compressão muito maiores do que os padrões MPEG anteriores. Além da DCT, o H.264 utiliza também estágios de predição de movimento, predição entre blocos dentro de um mesmo quadro (predição *intra*) e filtragem para remover blocos. A DCT utilizada é uma versão modificada, que realiza quantização de forma integrada e otimiza o cálculo com estágios aritméticos usando números inteiros. O padrão completo exige um sistema de alta capacidade computacional e por isso, a maioria das aplicações utiliza apenas as versões mais simples do padrão. Devido aos problemas de artefatos visuais da DCT (seção 3.3) e ao peso computacional, o H.264 completo é mais usado para vídeos em resolução HDTV, com altas taxas de *bits*, inviável para aplicações móveis.

Um problema encontrado na aplicação da DCT é que, no caso de dados computacionais como as imagens, tenta-se aproximar uma função de domínio limitado a um conjunto de funções de domínio infinito (estacionárias). Muitos sinais requerem uma aproximação mais flexível, onde seja possível variar o tamanho da janela a fim de determinar mais precisamente tempo e frequência. A análise *wavelet* representa um passo nessa direção: uma técnica de janelamento com regiões de tamanho variável. A análise *wavelet* permite o uso de intervalos longos de tempo onde existe informação de baixa frequência e intervalos menores de tempo para informação de alta frequência, pois suas funções-base são limitadas. Com isso, elementos transientes de um sinal podem ser representados mais eficientemente por uma quantidade de informação menor. No capítulo 5, tal transformada será abordada.

A TWD 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 TWD em importantes padrões recentes de compressão de imagem e vídeo, tais como o JPEG2000 (*Joint Photographic Experts Group* [1], Figura 3.5), MPEG-4 e MPEG-7 (*Moving Picture Experts Group*) [2] e em várias modificações.

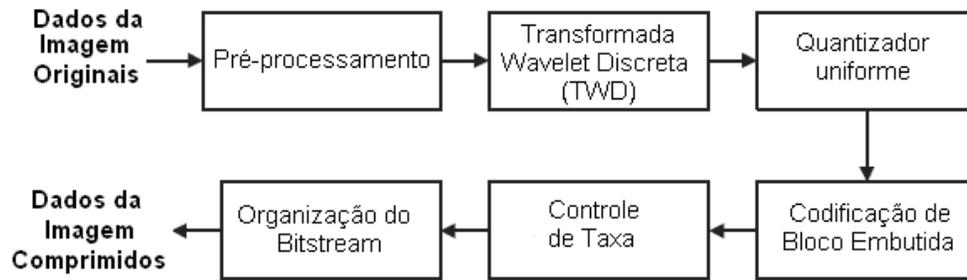


Figura 3.5– Diagrama de blocos de um codificador JPEG 2000 [41]

3.3 Artefatos Visuais

Para atingir taxas de *bits* aceitáveis, os algoritmos de compressão sacrificam parte da qualidade. Essas perdas são percebidas no caso de imagens digitais por artefatos visuais característicos do processo utilizado.

A degradação mais conhecida (característica que está presente em sistemas que envolvem uma representação por transformada ortogonal estacionária, como DFT, DCT e KLT), é o efeito de bloco. Trata-se de uma deterioração na qual a imagem recebida apresenta padrões retangulares que não estavam presentes na imagem original (Figura 3.6). As discontinuidades se tornam visíveis quando a quantização é pouco refinada (baixa taxa de *bits*) e são mais perceptíveis em áreas planas. [42]

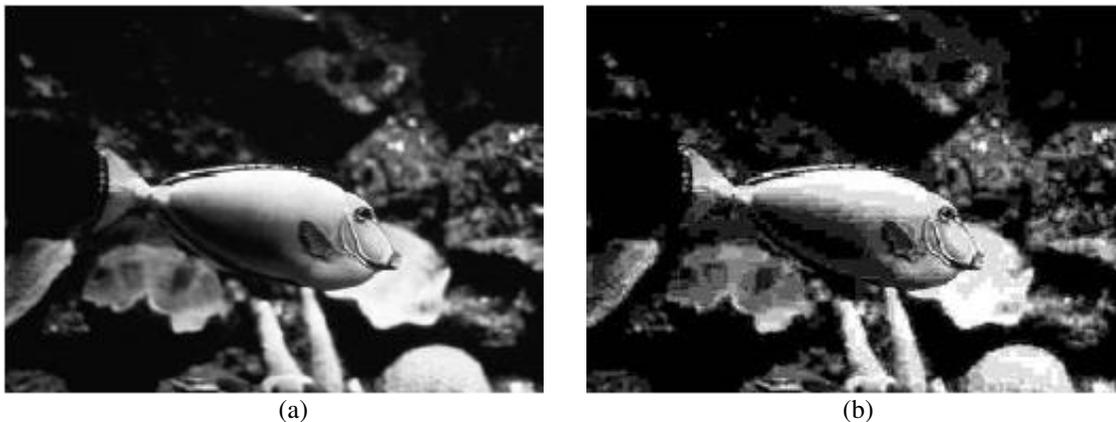


Figura 3.6 - Efeito de bloco em imagem codificada DCT (a) original (b) recuperada [43]

Métodos baseados em transformadas *wavelet* não apresentam efeito de blocos (os erros de quantização de um coeficiente não se espalham, pois advêm de uma representação compacta, não-estacionária), mas introduzem outros artefatos como *ringing* e embaçamento (*blurring*). O efeito de *ringing*, também chamado de efeito de Gibbs, é causado pela quantização ou truncamento dos coeficientes de alta frequência necessários à representação de um sinal com discontinuidades. São bastante perceptíveis na vizinhança de contornos da imagem. Em geral, artefatos de *ringing* são mais difíceis de caracterizar do que efeito de blocos. O embaçamento é um artefato que também surge pela falta de componentes de alta frequência, do lado oposto ao *ringing* (Figura 3.7). A Figura

3.8 mostra o efeito de *ringing* para a imagem ‘Lena’ quando é feita a codificação TWD com poucos coeficientes.

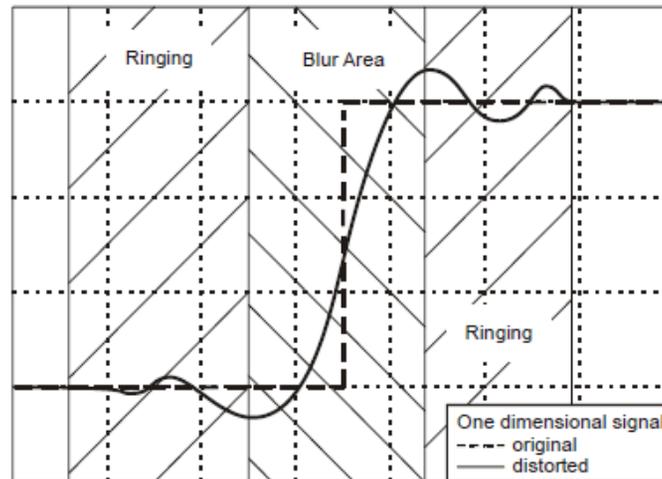


Figura 3.7 – *Ringing* e embaçamento (*blur*) em uma descontinuidade unidimensional [44]



Figura 3.8 - Exemplo de *ringing* em imagem codificada por TWD (a) original (b) recuperada

Diversas pesquisas têm sido realizadas utilizando quantização adaptativa e métodos híbridos, para tentar minimizar esses artefatos [45]. Propôs-se até mesmo a utilização de um novo espaço de funções, como as *countourlets*, *curvelets* e *bandelets*, ou dicionários de *wedgelets* [46], em técnicas de decomposição *cartoon*-textura, como será mostrado no próximo capítulo 4.

Capítulo 4

Metodologia *Cartoon*-Textura

Um conceito que tem sido usado por muitos pesquisadores é a distinção entre imagem do tipo *cartoon* e imagem do tipo textura. Uma imagem *cartoon* consiste de contornos e regiões com cores homogêneas, como em um desenho. Em uma imagem textura não existe uma informação geométrica bem caracterizada, o sinal se assemelha a um ruído ou a uma distribuição estacionária. Imagens naturais podem ser decompostas em componentes *cartoon* e textura. Nessa abordagem, pode-se definir também textura como toda a informação que não é *cartoon*.

Existem várias formas de se fazer a decomposição da imagem em componentes *cartoon* e textura. Embora as definições de textura variem de acordo com a aplicação a que se destinam, algumas propriedades são desejáveis para a ferramenta de transformação independente do contexto [47]:

- Isolar a componente de *cartoon* da componente textura para melhor uso de uma transformada adequada.
- Atingir boa separação em termos de baixa correlação entre *cartoon* e textura, obtendo alto grau de autonomia entre os algoritmos de compressão de cada componente.
- A componente de *cartoon* deve ser uma representação suave e sem textura da imagem, com contornos preservados.

Em [48], métodos de separação muito robustos são analisados, com base no trabalho de Meyer [49]. Com esses métodos obtém-se uma separação eficiente da imagem, como mostrado para a imagem ‘Barbara’ na Figura 4.1.

A idéia de comprimir em separado as componentes de contorno e os detalhes é bastante antiga, com alguns métodos remetendo-se á década de 80, antes mesmo dos principais padrões de compressão por DCT surgirem [50][51]. No entanto, esses métodos não atingiam taxas desejáveis e eram muito complexos, baseando-se muitas vezes em quantizações vetoriais e outros métodos adaptativos.

A formulação do que é tratado nesse texto como “*cartoon*” surgiu (e até hoje ainda é mais estudado) nas aplicações de recuperação de imagens por ruídos aditivos e embaçamento. Por isso, não se supôs inicialmente que essa técnica fosse aplicável para compressão, pois as imagens

cartoon não apresentam os requisitos de baixa entropia e não permitem recuperar fielmente a imagem original (compressão sem perdas). No entanto, outros pesquisadores mais recentes consideraram que essa decomposição poderia ser útil principalmente em baixas taxas, onde os artefatos gerados pelos outros métodos são mais danosos (subjetivamente) do que a “*cartoonização*” da imagem.

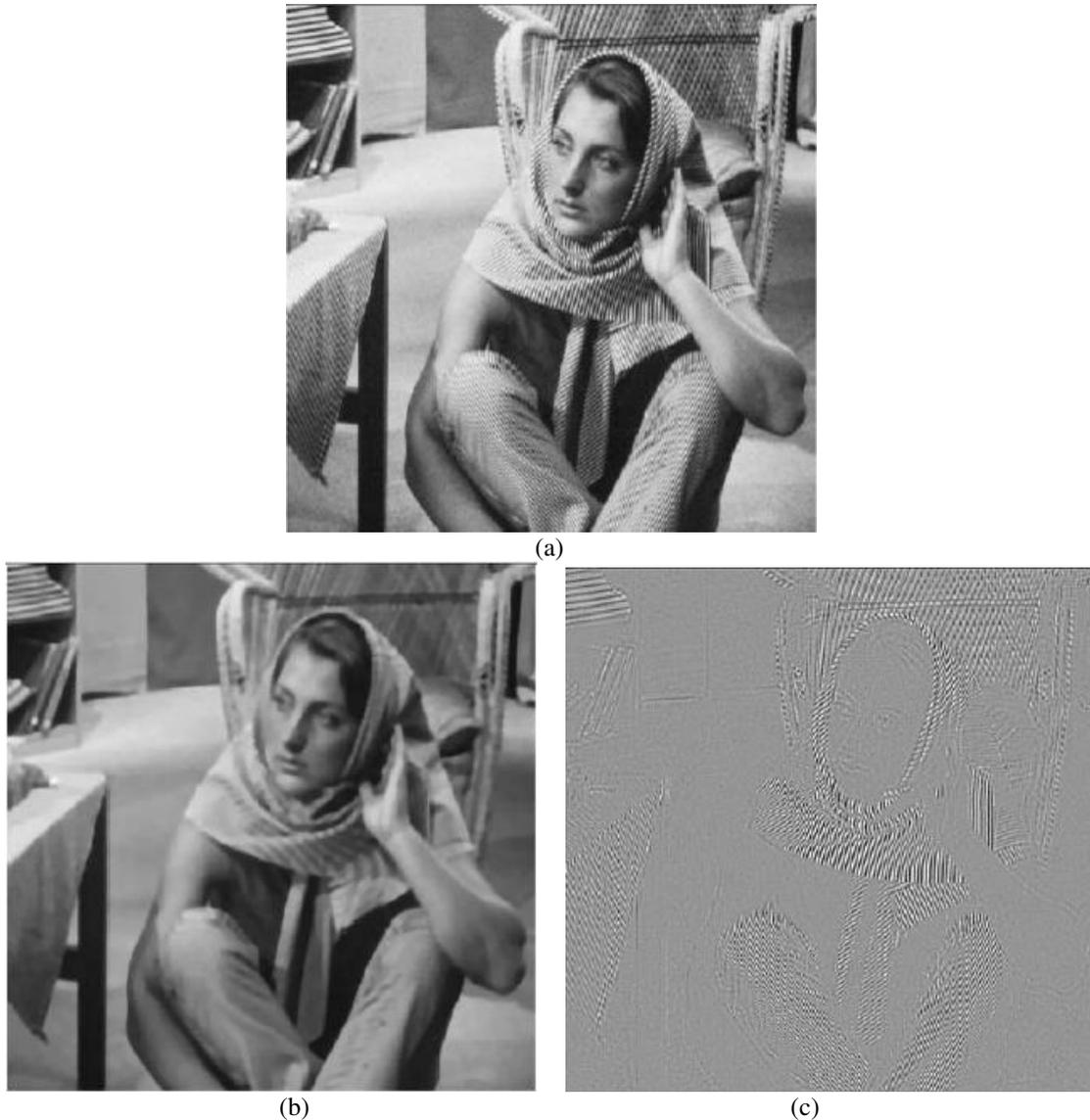


Figura 4.1 – Decomposição *cartoon-textura* de [48] (a) original (b) *cartoon* (c) textura

Sprljan e Izquierdo [47] propuseram um método de compressão híbrido utilizando a metodologia de tratar componentes *cartoon* e textura de forma separada, com métodos diferentes e ajustando a priori o número de *bits* usado para a codificação de cada uma. Seus resultados mostram uma variação pequena em termos de PSNR, mas justifica-se pela ausência de artefatos indesejados na definição das bordas.

A codificação independente *cartoon-textura* passou a ser mais estudada e modelos mais complexos (e mais adaptados para cada componente) foram desenvolvidos, conseguindo ganhos de compressão objetivos além da ausência de artefatos. Alguns desses métodos encontram-se resumidos no capítulo 6. Os maiores ganhos normalmente são conseguidos para altas taxas de compressão (baixa taxa de *bits*), casos em que os artefatos são mais visíveis.

Na Figura 4.2, são mostrados os resultados conseguidos por Sprljan e Izquierdo [47], variando a proporção de *bits* alocados para as componentes *cartoon* e *textura*. Sendo b a taxa de *bits* total da imagem codificada, b_c representa a taxa de *bits* alocada para a componente *cartoon* e b_t a taxa de *bits* alocada para a componente *textura*.

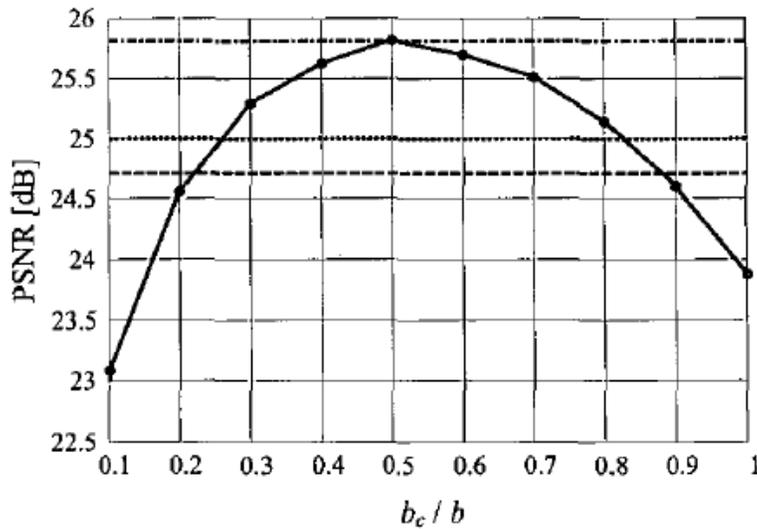


Figura 4.2 – Resultados da codificação híbrida de Sprljan e Izquierdo para a imagem 'Barbara' [47]

Nesse trabalho a decomposição é feita modelando a componente *cartoon* como o resultado de um processo Perona-Malik de difusão não-linear iterativa. Esse processo consiste na seguinte equação diferencial parcial: [52][53]

$$\frac{\partial I}{\partial t} = \text{div}[c(I, \sigma, k)\nabla I] \quad (4.1)$$

onde I é a imagem original e $c(I, \sigma, k)$ é a função de condutividade calculada para cada *pixel* da forma:

$$c(I, \sigma, k) = e^{-\left(\frac{|\nabla(I * G_\sigma)|}{k}\right)^2} \quad (4.2)$$

Denota-se por div o operador divergente ($\text{div}(I) = \nabla \cdot I$) e por ∇ o operador gradiente. O parâmetro k é um limitante para o gradiente da imagem, que define o limiar entre conservação das bordas e difusão de áreas particulares da imagem. Pela convolução da imagem com uma janela gaussiana G_σ de variância σ^2 , ruídos com gradiente impulsivo menor do que k são eliminados. A

cada passo de iteração da resolução de (4.1) a imagem I fica mais cartoon, estabilizando a um determinado ponto de acordo com os parâmetros ajustados.

Alguns autores propuseram minimizar a equação de difusão no domínio *wavelet*, por meio de aproximações não-lineares (somente os coeficientes *wavelets* de maior magnitude ou energia são utilizados na reconstrução, independente da subbanda a que pertencem [54]), num procedimento conhecido como *wavelet shrinkage*. Outra forma, foi utilizar o algoritmo *matching pursuits* (procuras sucessivas) de Mallat [55][56] para aproximar uma função de forma não-linear pelas componentes *wavelets* de maior magnitude e que satisfaçam a equação de minimização, utilizando para isso mais de uma base.

Esses métodos têm sido utilizados principalmente no contexto de recuperação de imagens com ruído aditivo ou embaçamento [57]. Para a codificação, o ganho em baixas taxas de compressão, mesmo com uma pequena perda na qualidade da imagem em relação à original, justifica a utilização da metodologia *cartoon-textura* para codificação de imagens e vídeo no ambiente de transmissão de imagens digitais em dispositivos móveis. Como mostrado no capítulo 2, nesse ambiente a taxa de *bits* disponível é muito baixa, e deseja-se uma imagem com a estrutura preservada, mesmo que em detrimento de detalhes de maior resolução, como texturas.

Em [54] Cohen et al. argumentam que a codificação separada de contornos e texturas usando aproximações não-lineares com dicionários multi-resolução organizados sob a forma de árvore apresentam desempenho mais adaptado a baixas taxas. Curiosamente, essa metodologia ainda não é adotada por nenhum padrão de codificação.

O padrão MPEG-4 Part 20 [58] permite a definição de regiões e camadas com parâmetros de codificação diferentes, mas não define como separar essas camadas nem tampouco como codificá-las, sendo uma especificação muito genérica para que um método verdadeiramente híbrido *cartoon-textura* seja aplicado. Por outro lado, os métodos apresentados na literatura ainda não apresentaram resultados significativos para uma ampla gama de imagens e muitas vezes requerem uma parametrização adequada (como o tamanho do dicionário ou o número de *bits* e a forma de codificação binária de cada componente gerado), não justificando a mudança de padrões.

Trata-se, portanto, de uma metodologia nova, que requer estudo mais aprofundado e testes em diferentes ambientes de aplicação, que é o que se pretende realizar nesta dissertação, focando o ambiente de transmissão móvel de imagens digitais. Não se pretende realizar um estudo exaustivo, mas uma justificativa para a adoção dessa metodologia em futuros codificadores. Pode servir também para uma possível introdução da tecnologia em outras aplicações de eventual interesse.

Além dos resultados inovadores em termos de qualidade versus número de componentes codificadas, faz-se necessário um estudo posterior acerca das melhores práticas para o cálculo e implementação das equações envolvidas na decomposição e codificação *cartoon-textura*. No melhor

dos casos, espera-se uma duplicação na complexidade computacional para tratar duas componentes distintas e operar com métodos de codificação diferentes. Esse aumento pode ser ainda maior considerando os cálculos envolvidos em determinar a melhor alocação de *bits* para cada componente ou os melhores parâmetros *cartoon-textura*.

Apresentam-se neste trabalho as vantagens da metodologia *cartoon-textura* combinada à *wedgelets* em termos de ganho de codificação considerando que o dispositivo onde irá ser implementada (principalmente do lado do codificador) não tem restrições e a implementação otimizada é deixada para futuros trabalhos (capítulo 8).

Capítulo 5

Transformada *Wavelet*

Neste capítulo serão apresentados os fundamentos teóricos da transformada *wavelet* discreta (TWD) e sua relação com a codificação de imagens. A TWD é considerada o método estado-da-arte, apesar de apresentar falhas na representação geométrica quando usada em baixas taxas de dados. O método proposto neste trabalho usa a TWD como método constituinte para a representação de blocos de textura. No capítulo 6, serão mostradas as desvantagens de se usar a TWD para representação de contornos.

Características que remontam algoritmos de rápida implementação, transmissão progressiva, forte capacidade de concentração de energia do sinal em poucos coeficientes, possibilidades de aplicação a técnicas combinadas, ausência de artefatos de blocagem e ampla gama de aplicações possíveis e técnicas beneficiadas com seu uso, posicionaram as transformadas *wavelet* no centro da maioria dos principais sistemas de compressão de imagens, áudio e vídeo comerciais e em estado da arte nos últimos anos. JPEG 2000 [1], MPEG-4 e MPEG-7 [2] são apenas alguns exemplos abrangendo áreas, tais como, codificação de imagens de radar [59], rastreamento geográfico com GeoRaster [60], análise de voz [61], tomografias [62], diagnósticos cardíacos [63] e em restauração de imagens de ultra-som [64], compressão de sinais ECG [65], detecção de câncer por meio de análise espectroscópica [66], mapeamento de genes [67], entre outros.

As transformadas *wavelets* foram desenvolvidas na última década como fruto das contribuições de pesquisadores das mais diversas áreas (matemática, física, estatística e engenharia), tendo como destaque os inúmeros trabalhos de I. Daubechies e S. Mallat e apresentam desempenho nas curvas de custo-benefício consideravelmente superior ao desempenho das demais transformadas, rapidamente ocupando local de destaque.

Neste capítulo será introduzida a transformada *wavelet* contínua (TWC) 1D e 2D, sua interpretação via teoria de banco de filtros, a expansão em série da *wavelet* (SW) 1D e 2D e a introdução à transformada *wavelet* discreta (TWD). Este capítulo restringe-se a funções quadraticamente integráveis, dados os sinais e imagens pertinentes.

5.1 Introdução: Waves x Wavelets

Matemáticos e engenheiros têm explorado aproximações usando transformadas que tenham funções-base de duração limitada chamadas *wavelets* ou ondeletas, ondas que variam em posição e frequência, com alguns exemplos ilustrados na Figura 5.1. As transformadas baseadas nelas são chamadas de transformadas *wavelet* [68], que operam no seu propósito mais básico, como a já previamente conhecida análise tempo-frequência. Essa última consiste em submeter o sinal a várias filtragens diferentes gerando gráficos como os da Figura 5.2 que permite uma análise do conteúdo do sinal, enquanto a TW faz o mesmo ao decompor o sinal ou imagem de entrada em sua base de funções.

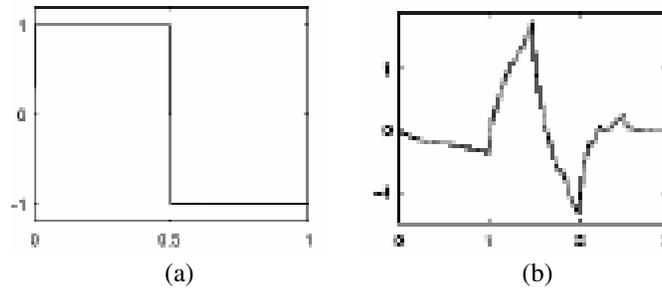


Figura 5.1 – Exemplos de funções *wavelet* (a) Haar (b) Daubechies 2

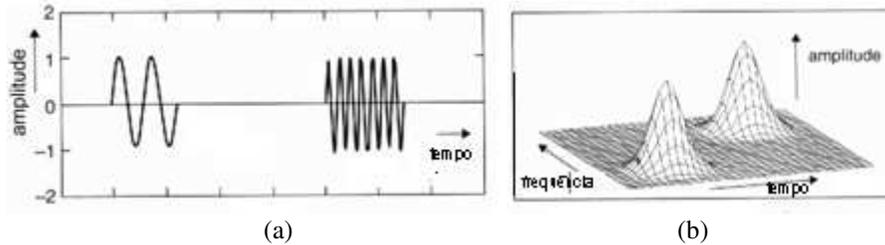


Figura 5.2. Espaço tempo-frequência: (a) sinal original; (b) análise no domínio *wavelet*

5.2 Transformada *Wavelet* Contínua (TWC) 1D

Define-se agora o espaço $L^2(\mathcal{R})$ como sendo a classe de funções $f(x)$ mensuráveis e quadraticamente integráveis (sinais de energia), ou seja, funções tais que:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx < \infty \quad (5.1)$$

Na TWC procura-se uma única função $\psi(x)$ capaz de gerar todo o espaço $L^2(\mathcal{R})$ por meio de dilatações e compressões de si mesma. Uma vez que $\psi(x)$ tem duração finita, característica básica tencionada pelas *wavelets*, para cobrir todo o espaço $L^2(\mathcal{R})$ é necessário utilizar combinações lineares dos deslocamentos de $\psi(x)$ ao longo da reta real. A função $\psi(x)$ é chamada de *wavelet*-básica, oscilatória, usualmente centrada na origem e que cai rapidamente a zero quando $|x| \rightarrow \infty$. Assim, $\psi(x) \in L^2(\mathcal{R})$.

Introduzida por Grossman e Morlet [69], na TWC, as funções-base $\psi_{a,b}(x)$ são comumente chamadas de *wavelets*-filhotes, versões deslocadas e dilatadas ou comprimidas da *wavelet*-básica) $\psi(x)$ e capazes de gerar todo o espaço $L^2(\mathcal{R})$. Assim, sendo $a > 0$ e b números reais, onde b é o parâmetro responsável pelo deslocamento de $\psi(x)$ ao longo do eixo x e a é o parâmetro responsável pela dilatação ou compressão, tem-se:

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x-b}{a}\right) \quad (5.2)$$

Exemplos de *wavelets* de Morlet podem ser observados na Figura 5.3. A constante $a^{-1/2}$ em (5.2) é o termo de normalização da energia da função com relação ao parâmetro a , e assegura que as normas das funções-base sejam iguais, uma vez que:

$$\left\| f\left(\frac{x-b}{a}\right) \right\| = \sqrt{\int_{-\infty}^{+\infty} \left| f\left(\frac{x-b}{a}\right) \right|^2 dx} = \sqrt{a} \|f(x)\| \quad (5.3)$$

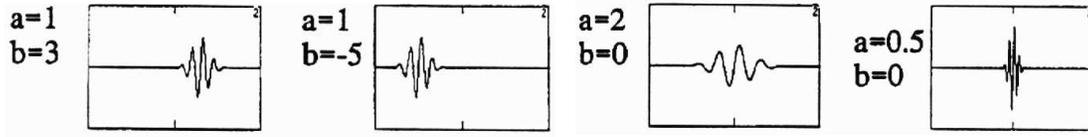


Figura 5.3. *Wavelets* de Morlet deslocadas

A TWC é uma transformada real e é definida pela seguinte expressão [68]:

$$W(a,b) = \langle f, \psi_{a,b} \rangle = \int_{-\infty}^{+\infty} f(x) \cdot \psi_{a,b}(x) dx \quad (5.4)$$

Sendo $\Psi(s)$ o espectro da *wavelet*-básica real $\psi(x)$, para se obter a TWC inversa, é necessário satisfazer o seguinte critério de admissibilidade que restringe a classe de funções que podem ser utilizadas como *wavelets*-básicas. [70] [71]:

$$C_\psi = \int_{-\infty}^{+\infty} \frac{|\Psi(s)|^2}{|s|} ds < \infty \quad (5.5)$$

Como s é denominador da integral, é necessário que:

$$\Psi(0) = 0 = \int_{-\infty}^{+\infty} \psi(x) dx = 0 \quad (5.6)$$

Como, além disso, $\Psi(\infty)=0$, pode-se afirmar que o espectro de amplitude da *wavelet*, *wavelet*-básica é similar à função de transferência de um filtro passa-faixas, resultado extremamente importante considerando toda a teoria básica de filtros que decorre. Se a constante C_ψ na equação (5.5) for finita, então existe a TWC inversa definida por:

$$f(x) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^{+\infty} W(a,b) \psi_{a,b}(x) db \frac{da}{a^2} \quad (5.7)$$

Como a *wavelet*-básica tem média nula, todos os escalonamentos e translações de (5.2) também apresentam média nula.

5.3 Interpretação da TWC por Banco de Filtros

A TWC também pode ser interpretada por meio do conceito de banco de filtros [68]. Define-se a *wavelet* escalonada de a e normalizada por $a^{-1/2}$ e seu complexo conjugado refletido, respectivamente, por:

$$\psi_a(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x}{a}\right) \quad (5.8)$$

$$\tilde{\psi}_a(x) = \psi_a^*(-x) = \frac{1}{\sqrt{a}} \psi^*\left(\frac{-x}{a}\right) \quad (5.9)$$

Pode-se reescrever (5.4) e (5.7), respectivamente, como:

$$W(a,b) = \int_{-\infty}^{+\infty} f(x) \psi_a(x-b) dx = [f * \tilde{\psi}_a](b) \quad (5.10)$$

$$f(x) = \frac{1}{C_\Psi} \int_0^\infty \int_{-\infty}^{+\infty} [f * \tilde{\psi}_a](b) \psi_a(b-x) db \frac{da}{a^2} = \frac{1}{C_\Psi} \int_0^\infty [f * \tilde{\psi}_a * \psi_a](x) \frac{da}{a^2} \quad (5.11)$$

$W(a,b)$ é a convolução de $f(x)$ com o complexo conjugado refletido da *wavelet*-básica escalonada definida em (5.8) para cada a . Conseqüentemente, a TWC pode ser interpretada como um banco de filtros lineares conforme a Figura 5.4. Deve-se frisar que a TW não envolve explicitamente o conceito de freqüência, embora freqüência possua relação implícita com escala. Analogias são utilizadas, contudo, em determinados pontos para evidenciar características.

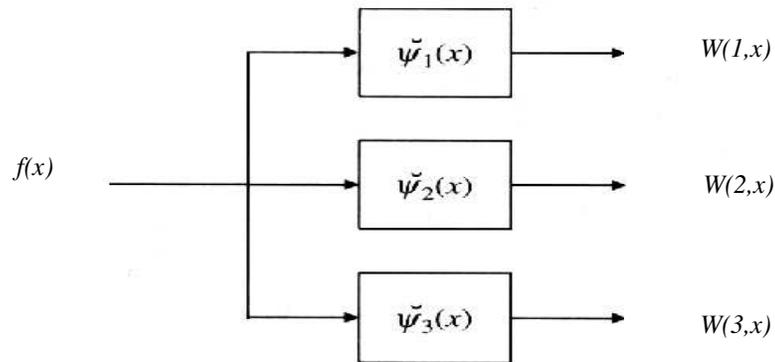


Figura 5.4. Interpretação da TWC por banco de filtros

Os filtros $\tilde{\psi}_1(x)$, $\tilde{\psi}_2(x)$ e $\tilde{\psi}_3(x)$ são, cada um, a *wavelet*-básica escalonada com uma “freqüência” diferente e portanto cada uma é uma família de *wavelets* graças ao deslocamento b . Cada uma é ainda um conjunto de funções-base de mesma “freqüência”. Cada saída da Figura 5.4 é uma convolução de $f(x)$ com uma *wavelet*-básica escalonada. Assim, fixado o valor de a , cada $W(a,b)$

corresponde a uma linha do gráfico da Figura 5.2.b. Além disso, se $\psi(x)$ for real e par, a reflexão e o conjugado não têm efeito, logo $\tilde{\psi}_a(x) = \psi_a(x)$. De (5.11), percebe-se que as saídas dos filtros, cada qual filtrada novamente por $\psi_a(x)$ e corretamente escalonada, se combinam para reconstruir $f(x)$.

Outra consideração importante é que, como:

$$TF\{f(ax)\} = \frac{1}{|a|} F\left(\frac{s}{a}\right) \tag{5.12}$$

tem-se:

$$\Psi_a(s) = TF\{\psi_a(x)\} = \sqrt{a} \Psi(as) \tag{5.13}$$

Logo, para valores altos de a , a *wavelet* definida em (5.8) sofre dilatação no tempo e conseqüente contração em freqüência e o centro da banda passante de filtros passa-faixas é deslocado para valores baixos de freqüência, implicando em analisar as baixas freqüências do sinal através de filtro estreito, como pode ser observado na Figura 5.5.a. Analogamente, para baixos valores de a , podem-se analisar as altas freqüências do sinal através de filtros mais largos. Já a largura da *wavelet* determina a resolução do sinal (Figura 5.5.b): *wavelets* mais largas implicam na utilização de maior número de componentes do sinal original na formação de cada coeficiente transformado (menor resolução no tempo e maior resolução em freqüência) e *wavelets* mais estreitas implicam maior resolução no tempo e menor em freqüência. Essa característica de resoluções da *wavelet* permite a análise de um sinal sob resoluções diferentes simultaneamente.

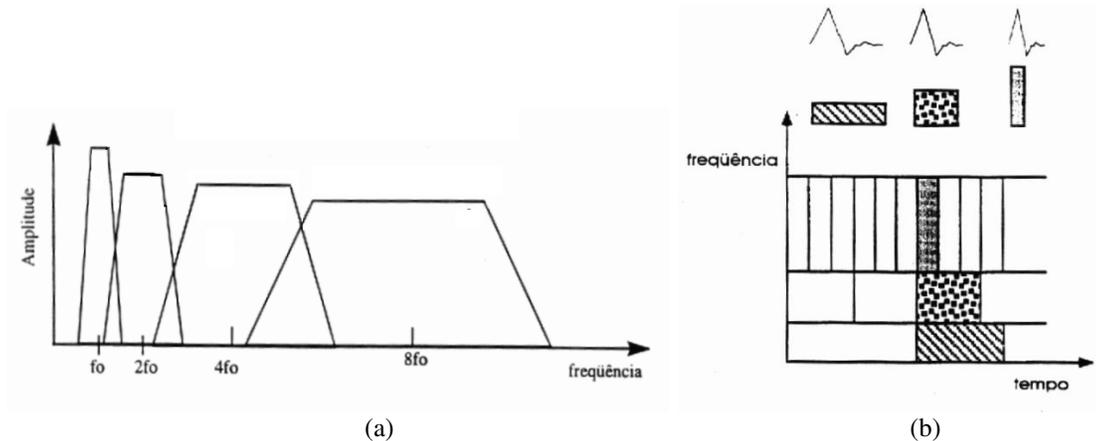


Figura 5.5. TWC: (a) bandas de freqüências das janelas; (b) resolução no plano tempo-freqüência [72]

Suponha que $\psi(x)$ seja uma *wavelet* real. Uma vez que possui média zero, a integral mede a variação de $f(x)$ numa vizinhança de b proporcional a a . Mallat [55] prova que quando a escala vai a zero, o decaimento do coeficiente *wavelet* caracteriza a regularidade da função de entrada na vizinhança de b . Tal característica possui aplicações importantes na detecção de transientes e fractais de análise. Propriedades referentes à completude e redundância das transformadas *wavelets* reais podem ser encontradas em [55].

Na Figura 5.6 é apresentada uma *wavelet*, do tipo conhecido como “chapéu mexicano”, aplicada a um sinal regular por partes à esquerda e singular à direita, com escala máxima menor que 1, dada a normalização do suporte de $f(x)$ de $[0,1]$. A escala mínima é limitada pelo intervalo de amostragem do sinal discretizado. Nota-se que, à medida em que a escala decresce, a *wavelet* cai rapidamente a zero nas regiões onde o sinal é irregular; enquanto as singularidades isoladas à esquerda geram cones nos coeficientes *wavelet* de larga amplitude que convergem para a localização das singularidades.

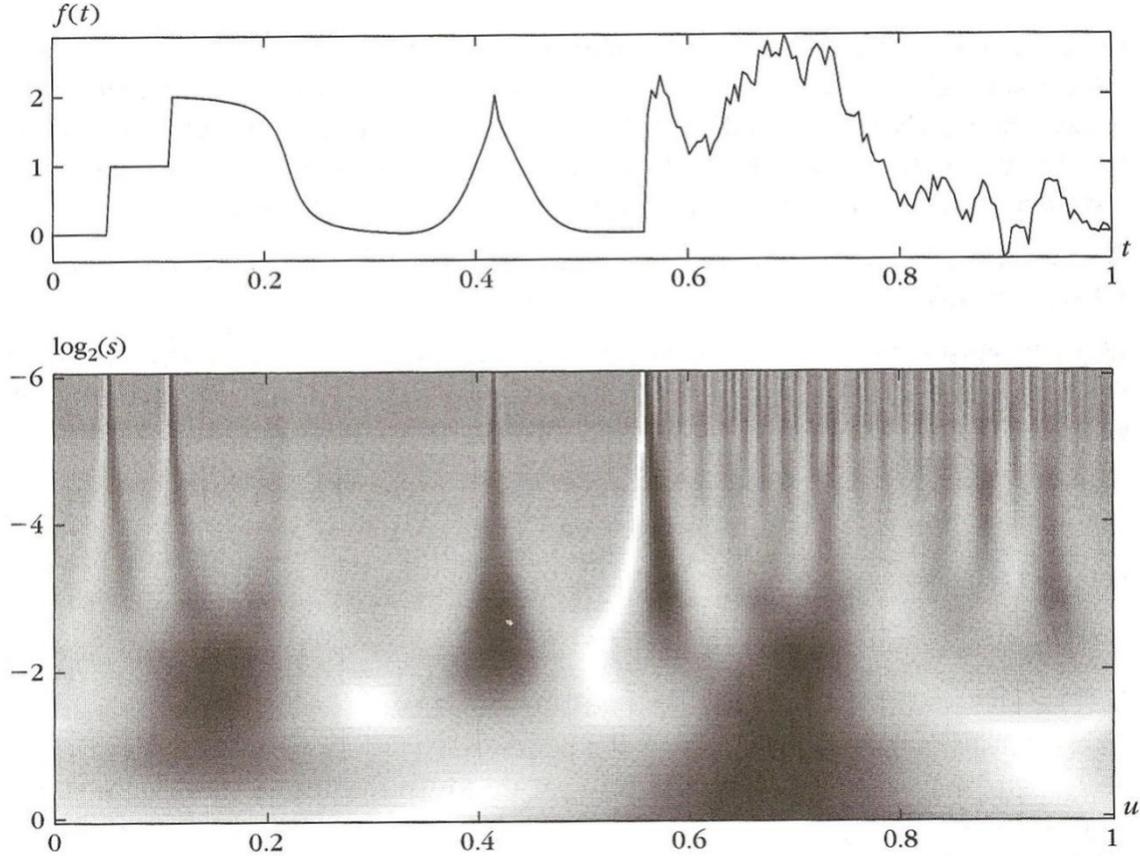


Figura 5.6 - TWC calculada com “chapéu mexicano”. As cores “preta”, “cinza” e “branca” correspondem, respectivamente, aos valores “positivo”, “zero” e “negativo” dos coeficientes [55].

5.4 TWC Bidimensional

Configurando b_x e b_y a translação em duas dimensões, a TWC-2D de $f(x,y)$ é dada por [68]:

$$W(a; b_x; b_y) = \left\langle f(x, y), \psi_{a; b_x; b_y}(x, y) \right\rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \psi_{a; b_x; b_y}(x, y) dx dy \quad (5.14)$$

$\psi_{a; b_x; b_y}(x, y)$ são as *wavelets*-filhotes bidimensionais geradas a partir da *wavelet*-básica bidimensional $\psi(x, y)$ e são dadas por:

$$\psi_{a; b_x; b_y}(x, y) = \frac{1}{a} \psi\left(\frac{x - b_x}{a}, \frac{y - b_y}{a}\right) \quad (5.15)$$

SW

Para obter a TWC 2D inversa, novamente, é necessário que a condição de admissibilidade seja satisfeita [72]. Assim, seja $\Psi(s_x, s_y)$ a TF 2D de $\psi(x, y)$. Logo, tem-se:

$$C_\Psi = \int_0^\infty \int_0^{2\pi} \frac{|\Psi(r \cos \theta, r \sin \theta)|^2}{r} dr d\theta < \infty \quad (5.16)$$

Caso $\psi(x, y)$ seja esfericamente simétrica, sua TF também o é, simplificando a condição de admissibilidade [71]:

$$C_\Psi = \int_0^\infty \frac{|\Psi(a^{-1}s_x, a^{-1}s_y)|^2}{a} da < \infty, \quad \forall (s_x, s_y) \in \mathbb{R}^2 \quad (5.17)$$

Se a constante C_Ψ for finita, então existe a TWC 2D inversa, definida por:

$$f(x, y) = \frac{1}{C_\Psi} \int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty W(a; b_x, b_y) \psi_{a; b_x, b_y}(x, y) db_x db_y \frac{da}{a^3} \quad (5.18)$$

Dado o alto grau de redundância (sobrecompleta), a TWC 2D possui destaque em decomposição e análise, em detrimento da desejável representação compacta. Entretanto, tal característica é considerada extremamente desejável em diversos procedimentos e técnicas de processamento de imagem que requerem acesso isolado a determinados componentes.

5.5 Expansão em Série da Wavelet (SW)

Por meio da discretização do parâmetro escalar a , ou de ambos, escala a e deslocamento b , mantendo-se contínua, a variável tempo x , pode-se obter a expansão em série *wavelet* (SW). Caso, ambos os parâmetros sejam discretizados, a SW compõe um conjunto enumerável de coeficientes, que corresponde a pontos, na grade bidimensional deslocamento-escala [72]. Nesse caso, sendo j e n inteiros e a_0 e b_0 sendo os comprimentos dos passos discretos, tem-se:

$$a = a_0^j \quad (5.19)$$

$$b = k \cdot b_0 \cdot a_0^j = n \cdot a_0^j \quad (5.20)$$

A redundância é evitada pelo fato do parâmetro b sofrer uma discretização proporcional ao parâmetro a : para valores maiores de a , é maior o número de componentes do sinal para compor cada coeficiente transformado. Logo, os passos de deslocamento da *wavelet* devem ser maiores. Da mesma forma, para valores pequenos de a , são necessários passos de deslocamento menores.

Substituindo (5.19), (5.20) e (5.9) em (5.10), tem-se a SW de $f(x)$ [70]. Nesta pesquisa, $W(j, n)$ será referenciado simplesmente como $W_{j, n}$ por questão de simplicidade de referências.

$$W(j, n) = a_0^{-j/2} \int_{-\infty}^{+\infty} f(x) \cdot \psi(a_0^{-j} \cdot x - n) dx \quad (5.21)$$

5.5.1 Wavelets Diádicas Invariantes na Translação (*Translation-Invariant Dyadic Wavelet Transform*)[55]

Caso a discretização seja somente do parâmetro escala, mantendo contínua a variável tempo x , pode-se obter a expansão em série *wavelet* (SW) invariante na translação. Respectivamente, sendo j inteiro e a_0 o comprimento do passo discreto, tem-se:

$$a = a_0^j \quad (5.22)$$

O tamanho do passo a_0 é usualmente $1/2$, uma vez que dilatar um sinal por um fator de dois corresponde simplesmente a tomar uma amostra sim e outra não do sinal, que produzirá as chamadas “*wavelets* diádicas invariantes na translação”. Para esse valor de a_0 são formadas funções-base a partir da *wavelet*-básica por meio de escalonamentos binários de 2^j . Logo, a equação 5.2 torna-se:

$$\psi(j, b) = a^{-1/2} \psi\left(\frac{x-b}{a}\right) = 2^{j/2} \psi\left(\frac{x-b}{2^{-j}}\right) \quad (5.23)$$

A *wavelet* diádica invariante na translação de $f(x)$ é definida por:

$$W(j, b) = \langle f, \psi_{j,b} \rangle = \int_{-\infty}^{\infty} f(x) 2^{j/2} \psi\left(\frac{x-b}{2^{-j}}\right) dx = f * \tilde{\psi}_j(b), \text{ com:} \quad (5.24)$$

$$\psi_j(x) = \psi_j(-x) = 2^{j/2} \psi\left(\frac{-x}{2^{-j}}\right) \quad (5.25)$$

A Figura 5.8 apresenta tal *wavelet* calculada em 5 escalas para a *wavelet spline* quadrática da Figura 5.7.

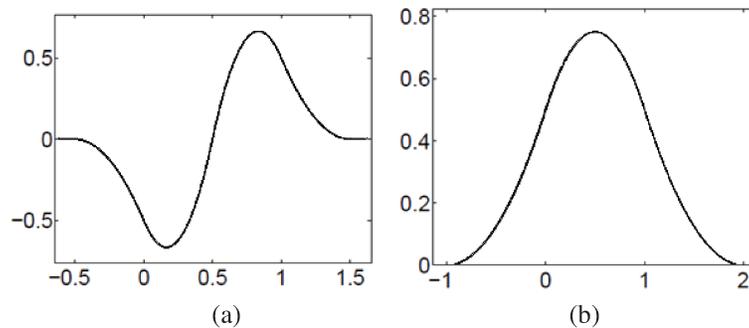


Figura 5.7 - *Spline* quadrática (a) função *wavelet* (b) função-escala [55]

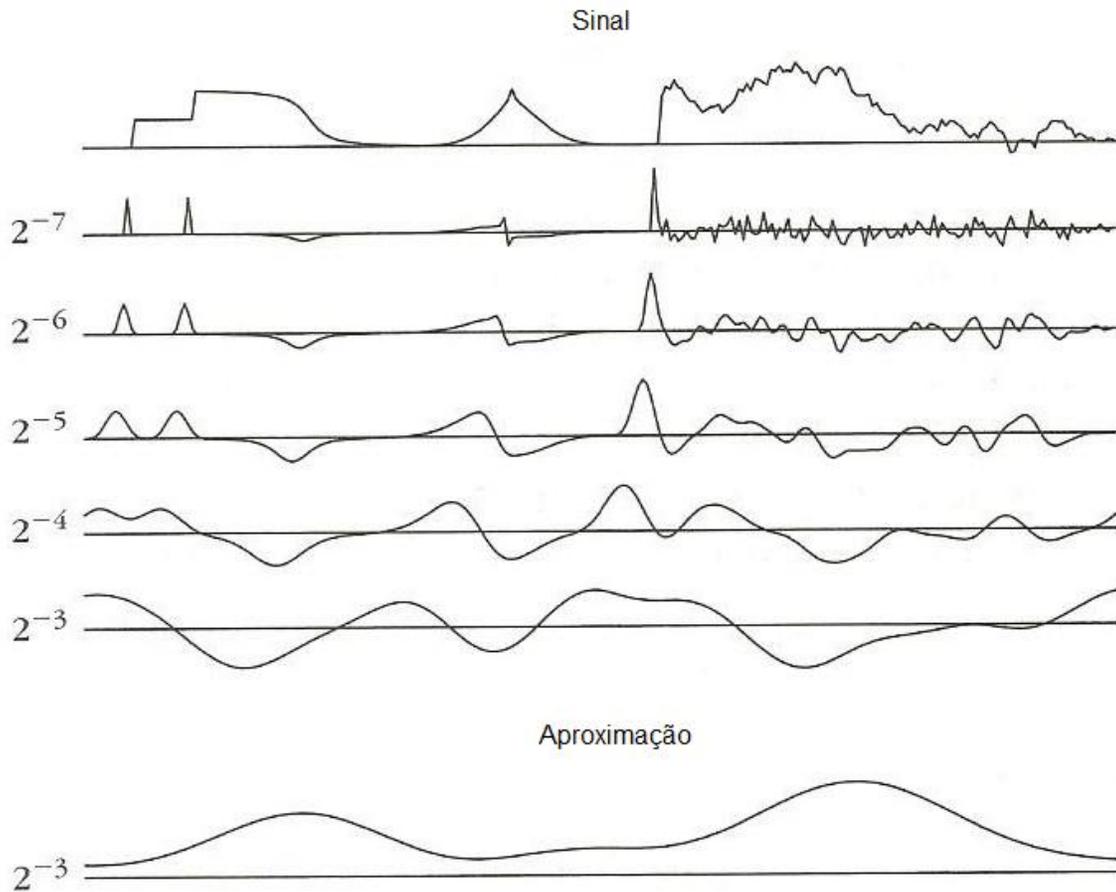


Figura 5.8 - Wavelet diádica nas escalas $2^{-7} \leq 2^{-j} \leq 2^{-3}$ para sinal definido sobre $[0, 1]$. A curva inferior carrega frequências mais baixas (escalas maiores que escala 2^{-3}) [55].

5.5.2 Wavelets Diádicas

Caso ambos os parâmetros, deslocamento e escala sejam discretizados e o tamanho do passo a_0 for $\frac{1}{2}$, mantendo contínua a variável tempo x , pode-se obter a expansão em série *wavelet* (SW) a partir das chamadas simplesmente “*wavelets* diádicas”. Para esse valor de a_0 são formadas funções-base a partir da *wavelet*-básica por meio de escalonamentos binários de 2^j e deslocamentos diádicos de $k/2^j$. As *wavelets* diádicas são definidas por [68]:

$$\psi_{j,n}(x) = s^{j/2} \cdot \psi(2^j \cdot x - n) \quad j, n \in \mathbb{Z} \quad (5.26)$$

Caso seja escolhida uma *wavelet*-básica de norma unitária, as *wavelets*-filhotes $\psi_{j,n}(x)$ também apresentarão normas unitárias, logo, há conservação de energia de $f(x)$ em $W_{j,n}$. A Figura 5.9 mostra a grade de amostragem no plano deslocamento-escala para *wavelets* diádicas onde cada nó é um coeficiente-*wavelet* $W_{j,n}$ associado a uma *wavelet* $\psi_{j,n}(x)$.

Uma grade muito esparsa verticalmente indica uma diferença muito grande entre escalas consecutivas de duas *wavelets*, o que pode vir a provocar perda de informação durante o processo

de filtragem e, eventualmente, a invalidar a equação de reconstrução. Já uma grade muito densa verticalmente poderia até mesmo, a despeito de uma reconstrução fiel do sinal, promover considerável redundância. Desse compromisso entre fidelidade e eficiência na escolha da *wavelet*-básica e sua grade de operação decorre que o ideal é que as grades sejam apenas densas, o suficiente, de modo a promover um mínimo de perda e um máximo de eficiência, daí a busca por funções-base ortonormais.

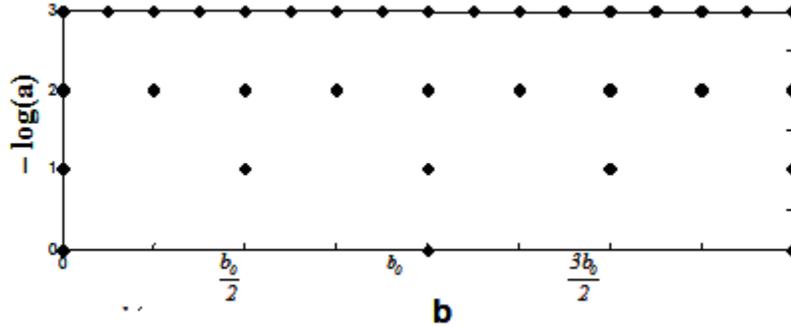


Figura 5.9 - Grade de amostragem no plano deslocamento-escala para *wavelets* diádicas ($a_0=1/2$)

A função $f(x) \in L^2(\mathbb{R})$ pode ser reconstruída a partir dos coeficientes-*wavelet* $W_{j,n}$ por [70]:

$$f(x) = \sum_{j=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} W_{j,n} \cdot \psi_{j,n}(x) \quad (5.27)$$

onde as funções-base $\psi_{j,n}(x)$, $-\infty < j, n < \infty$, formam uma base ortonormal de $L^2(\mathbb{R})$, ou seja:

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l} \cdot \delta_{k,m} = \begin{cases} 1, & \text{se } j = l, k = m \\ 0, & \text{cc} \end{cases} \quad (5.28)$$

sendo l, m inteiros e $\delta_{j,n}$ sendo a função delta de Kronecker. Os coeficientes da série são dados por:

$$W_{j,n} = \langle f, \psi_{j,n} \rangle \quad (5.29)$$

A série em (5.27) converge em $L^2(\mathbb{R})$, ou seja:

$$\lim_{M_1, N_1, M_2, N_2 \rightarrow \infty} \left\| f(x) - \sum_{j=-M_2}^{N_2} \sum_{n=-M_1}^{N_1} W_{j,n} \cdot \psi_{j,n} \right\| = 0 \quad (5.30)$$

Assim, para as *wavelets* diádicas, os coeficientes da SW são dados por [70]:

$$W_{j,n} = \langle f, \psi_{j,n} \rangle = 2^{j/2} \int_{-\infty}^{\infty} f(x) \cdot \psi(2^j x - n) dx \quad (5.31)$$

Em (5.27), a função contínua é representada por uma dupla seqüência infinita, mas se $\psi(x)$ for escolhida apropriadamente, pode-se truncar a série sem que haja erro significativo de aproximação. Se $f(x)$ tiver suporte compacto e se a *wavelet*-básica for bem localizada (i.e., decaia rapidamente a

zero à medida que se afasta da origem), então muitos dos coeficientes com altos valores de $|n|$ serão de pequena magnitude, podendo ser negligenciados. Já para altos valores de $|j|$ a área das funções-base tende a zero e portanto, os coeficientes relacionados a ela podem ser negligenciados, promovendo redução da complexidade computacional.

5.5.3 SW Bidimensional

Analogamente ao caso unidimensional, fixando os valores dos passos a_0 e b_0 como: $a = a_0^j$; $b_x = n_x \cdot a_0^j$ e $b_y = n_y \cdot a_0^j$,

A SW-2D é dada por:

$$W_{j;n_x,n_y} = a^{-j} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot \psi(a_0^{-j}x - n_x, a_0^{-j}y - n_y) dx dy \quad (5.32)$$

e as *wavelets* diádicas ($a_0 = 1/2$, $b_0 = 1$) 2D, são dadas por:

$$\psi_{j;n_x,n_y}(x, y) = 2^j \psi(2^j x - n_x, 2^j y - n_y) \quad (5.33)$$

5.6 Princípios da Transformada Wavelet Discreta (TWD)

A transformada *wavelet* discreta (TWD), que na presente pesquisa, terá seu desempenho na codificação de imagens analisado comparativamente com as transformadas *wedgelets*, apresenta um ótimo desempenho na compressão, processamento e análise de imagens. Dado o forte envolvimento da teoria de aproximações por multiresolução na TWD, se torna mais interessante abordar brevemente a codificação por subbanda [68] e o algoritmo conhecido como *Mallat's herringbone algorithm* [73]; uma vez que, de ambos, foi derivada a aplicação prática dessa transformada.

5.6.1 Codificação por Subbanda

Técnicas de codificação por subbanda foram primeiramente aplicadas na literatura no campo da codificação de voz por Crochiere [74] e ganharam espaço nesse campo, muito antes que seu potencial para a codificação de imagens fosse explorado. Essa técnica decompõe o sinal 1D ou 2D em componentes limitados em faixas de frequência e os representa sem redundância e de forma a possibilitar a reconstrução do sinal sem erros [68][75][76].

A codificação por subbanda invoca uma espécie de “análise tempo-frequência” do sinal na qual o sinal é particionado em um número de diferentes bandas de frequência. Logo, há a partição do eixo de frequências em intervalos disjuntos de modo a permitir a implementação de filtro passa-faixas, ao qual o sinal será submetido, conforme a Figura 5.10. Podem ser adotados M intervalos

disjuntos de comprimento $2s_N / M$ produzindo M sinais com subbanda s_N / M cada, onde para um sinal limitado em banda com TF nula $F(s) = 0$ para $|s| \geq s_N$:

$$S_{Nyquist} = \frac{1}{\Delta x} = 2 \cdot s_N \quad (5.34)$$

Dessa forma, componentes de frequências diferentes aparecerão em diferentes canais de subbanda.

Deve ser notado que sinais 2D podem ser escaneados segundo uma varredura por linhas ou mesmo zigzag (priorizando a correlação de coeficientes) e serem processados analogamente. Essas subbandas podem ainda ser quantizadas independentemente, de acordo com a priorização de aplicação. Tal procedimento permite que *codecs* identifiquem as subbandas de maior concentração de energia, facilitando uma alocação de *bits* apropriada ou otimizando codificadores entrópicos.

Supondo dois canais de subbanda, o intervalo de frequências é particionado no ponto $s_N/2$ submetendo o sinal a dois filtros que gerarão dois sinais de subbanda: um sinal meia-banda baixa e um sinal meia-banda alta.

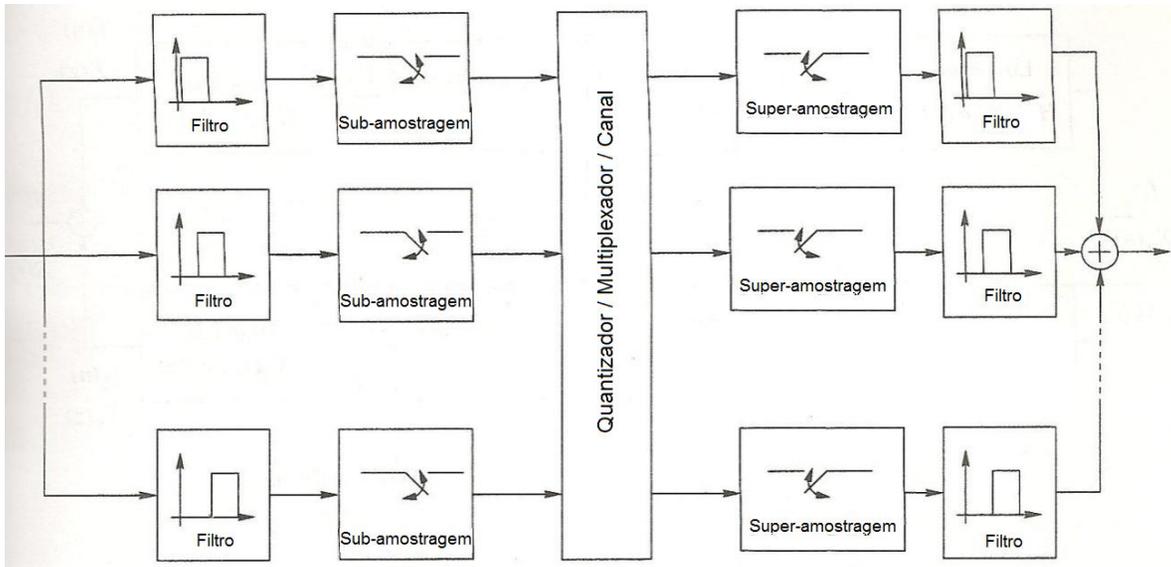


Figura 5.10 – Esquema de codificação por subbanda [74].

Desse modo, observando a Figura 5.11 e sabendo que $y_1(i \cdot \Delta x)$ contém exatamente a informação de alta frequência que foi eliminada de $f(i \cdot \Delta x)$ durante a geração de $y_0(i \cdot \Delta x)$, pode-se dizer que $y_1(i \cdot \Delta x)$ e $y_0(i \cdot \Delta x)$ contém juntos toda a informação presente no sinal original $f(i \cdot \Delta x)$ [68]. Assim, tem-se:

$$f(i \cdot \Delta x) = y_0(i \cdot \Delta x) + y_1(i \cdot \Delta x) = f(i \cdot \Delta x) * h(i \cdot \Delta x) + f(i \cdot \Delta x) * g(i \cdot \Delta x), \text{ pois: } (5.35)$$

$$H(s) + G(s) = 1 \quad (5.36)$$

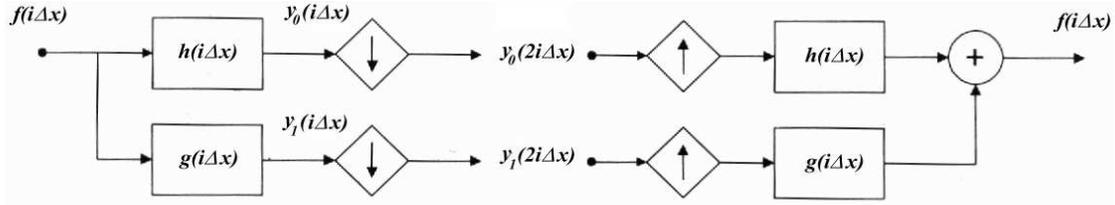


Figura 5.11 – Codificação por subbanda de duas bandas e reconstrução.

Logo, o sinal original $f(i \cdot \Delta x)$ de N pontos pode ser codificado sem redundância em dois sinais de $N/2$ pontos. Como $y_0(i \cdot \Delta x)$ e $y_1(i \cdot \Delta x)$ podem ser completamente recuperados a partir dos sinais codificados em subbanda, $f(i \cdot \Delta x)$ pode ser reconstruído sem erros no decodificador.

Logo, a codificação por subbanda em dois canais requer somente a filtragem de $f(i \cdot \Delta x)$ com $h(i \cdot \Delta x)$ e $g(i \cdot \Delta x)$, seguidas pela subamostragem de cada saída:

$$y_0(2i \cdot \Delta x) = \sum_k f(k \cdot \Delta x) \cdot h((-k + 2i)\Delta x) \quad (5.37)$$

$$y_1(2i \cdot \Delta x) = \sum_k f(k \cdot \Delta x) \cdot g((-k + 2i)\Delta x) \quad (5.38)$$

Para a reconstrução do sinal original basta superamostrar os dois sinais codificados em subbanda, interpolá-los com $2h(i \cdot \Delta x)$ e $2g(i \cdot \Delta x)$, e em seguida, somá-los, ou seja:

$$f(i \cdot \Delta x) = 2 \sum_k [y_0(2k \cdot \Delta x) \cdot h((-i + 2k) \cdot \Delta x) + y_1(2k \cdot \Delta x) \cdot g((-i + 2k) \cdot \Delta x)] \quad (5.39)$$

O processo completo encontra-se esquematizado na Figura 5.11. O *design* dos filtros adotados é crucial uma vez que os lobos do espectro sobrepostos resultam na distorção bem conhecida como *aliasing*, que não permite recuperação.

5.6.2 Introdução ao Algoritmo Rápido para TWD

Mallat [73] definiu um algoritmo iterativo para a TWD chamado de FWT (*Fast Wavelet Transform* - Transformada Wavelet Rápida) ou *Mallat's herringbone algorithm*, mais eficiente do que calcular o conjunto completo de produtos internos.

No algoritmo, após aplicar a codificação por subbanda em dois canais ao sinal $f(i \cdot \Delta x)$, o sinal meia-banda baixa $y_0(i \cdot \Delta x)$ é novamente sujeito à mesma codificação por subbanda em dois canais. Desse procedimento deriva um sinal meia-banda alta, com $N/2$ coeficientes e dois sinais subbanda com $N/4$ coeficientes. A cada passo é retido o sinal meia-banda alta e processado o sinal meia-banda baixa, até que reste apenas um único coeficiente (Figura 5.12). Os coeficientes transformados são então o coeficiente banda-baixa e o conjunto de coeficientes meia-banda alta codificados. No final, tem-se um total de N coeficientes.

Cada conjunto de coeficientes transformados é obtido por meio da convolução de $f(i \cdot \Delta x)$

repetidamente com $h(i.\Delta x)$ e somente uma vez com $g(i.\Delta x)$. Daí, as funções-base da TWD, $\psi_{j,n}(x)$ serem a reflexão de $g(i.\Delta x)$ e de outras funções derivadas da convolução de $g(i.\Delta x)$ repetidamente com $h(i.\Delta x)$. [76]. O processo de reconstrução de $f(i.\Delta x)$ encontra-se esquematizado na Figura 5.13.

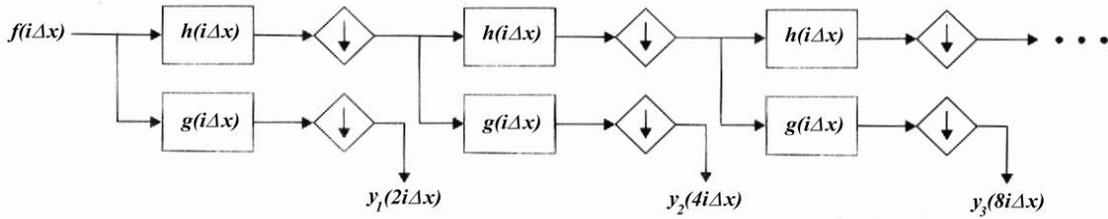


Figura 5.12 - Algoritmo de Mallat (FWT)

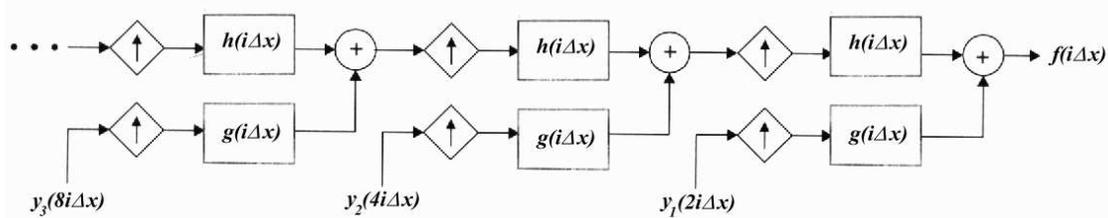


Figura 5.13 - Algoritmo inverso de Mallat

5.7 Resolução e Aproximação por Multiresolução em $L^2(\mathbb{R})$

A resolução r_j de um sinal $f(x) \in L^2(\mathbb{R})$ é a quantidade r_j de pontos para os quais ele é definido [72][73][76]. Funções contínuas, por exemplo, possuem resolução infinita, uma vez que são conhecidas para todos os pontos do intervalo.

Uma aproximação de $f(x)$ é gerada a partir da alteração da resolução do sinal $f(x)$ original. A análise multiresolução é uma técnica que permite analisar o sinal por meio da decomposição de seu conteúdo em um conjunto de detalhes em diferentes resoluções. As aplicações e usos são claros considerando que adaptar a resolução de um sinal permite o acesso e processamento somente de detalhes relevantes para uma aplicação particular [55]. Em visão computacional, Burt e Adelson [77] introduziram uma pirâmide multiresolução que pode ser usada para processar uma imagem de baixa resolução primeiro e então, seletivamente, aumentar a resolução quando necessário.

Dada uma seqüência crescente de resoluções $\{r_j, j \in \mathbb{Z}\}$ 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 resolução r_j e a aproximação com resolução r_{j-1} , mais baixa [73].

5.8 Aproximação por Multiresolução em $L^2(\mathbb{R})$: propriedades

Aproximações multiresolução representam o sinal em várias resoluções por meio de projeções

ortogonais em diferentes espaços métricos $\{V_{2^j}\}$, $j \in \mathbb{Z}$. Aproximações multiresolução são inteiramente caracterizadas por filtros discretos particulares que governam a perda de informação que ocorre entre as resoluções. Esses filtros providenciam uma forma simples de projetar e sintetizar bases *wavelets* ortogonais, e portanto, de realizar a TWD [55].

A aproximação de uma função $f(x)$ numa resolução 2^j é especificada por uma grade discreta de amostras que provêm médias locais de $f(x)$ ao longo de vizinhanças de tamanho proporcional a 2^j . Logo, a aproximação por multiresolução é composta por grades de aproximação “embutidos”. Mais formalmente, a aproximação de uma função na resolução 2^j é definida como a projeção ortogonal na resolução 2^j . [55]

5.8.1 O Operador de Projeção

Seja o sinal original $f(x) \in L^2(\mathbb{R})$. O operador A_{2^j} é um operador de projeção de $f(x)$ no subespaço vetorial $V_{2^j} \subset L^2(\mathbb{R})$, gerando sua aproximação com resolução 2^j , portanto, provocando a mudança na resolução de $f(x)$.

O subespaço vetorial V_{2^j} pode ser interpretado como o conjunto de todas as possíveis aproximações de todas as funções $L^2(\mathbb{R})$ no nível de resolução 2^j . O conjunto dos subespaços vetoriais V_{2^j} forma o espaço vetorial $\{V_{2^j}, j \in \mathbb{Z}\}$ que será referenciado ao longo do texto como $\{V_{2^j}\}$ de forma a diferenciá-lo de cada V_{2^j} em particular.

A aproximação com resolução 2^j é denotada por $A_{2^j} \cdot f(x)$ e o operador apresenta as seguintes propriedades [73]:

5.8.2 Propriedades [55]

Propriedade 5.1 $A_{2^j} \cdot f(x)$ não é modificada se a operação for repetida no nível de resolução 2^j , ou seja,

$$A_{2^j} \cdot A_{2^j} = A_{2^j} \quad (5.40)$$

Propriedade 5.2 Há várias aproximações da função $f(x)$ no nível de resolução 2^j . Se A_{2^j} for encarado como um pente de Dirac com 2^j amostras no intervalo $]0, x_0[$, basta pensar que as diferentes aproximações seriam geradas por diferentes deslocamentos desse pente, resultando numa espécie de mudança na taxa de amostragem. Dentre todas as aproximações de $f(x)$ no nível de resolução 2^j , $A_{2^j} \cdot f(x)$ é considerada a função que mais se aproxima de $f(x)$, ou seja:

$$\forall y(x) \in V_{2^j}, \quad |y(x) - f(x)| \geq |A_{2^j} \cdot f(x) - f(x)| \quad (5.41)$$

e, por consequência, o operador A_{2^j} calcula a projeção ortogonal da função no espaço vetorial V_{2^j} .

Propriedade 5.3 Causalidade: A aproximação de $f(x)$ no nível de resolução 2^{j+1} contém toda a informação necessária para calcular a sua aproximação mais “grosseira” no nível de resolução 2^j .

Sendo A_{2^j} operador de projeção em V_{2^j} :

$$\forall j \in \mathbb{Z}, \quad V_{2^j} \subset V_{2^{j+1}} \quad (5.42)$$

e, recursivamente, obtém-se a seguinte hierarquia de subespaços:

$$\dots \subset V_{2^{-1}} \subset V_{2^0} \subset V_{2^1} \subset V_{2^2} \subset \dots \quad (5.43)$$

Propriedade 5.4 A operação aproximação não depende do nível de resolução. Os subespaços das aproximações podem ser obtidos a partir de outros subespaços, por meio da dilatação ou compressão das aproximações:

$$\forall j \in \mathbb{Z}, \quad A_{2^j} \cdot f(x) \in V_{2^j} \Rightarrow A_{2^j} \cdot f\left(\frac{x}{2}\right) \in V_{2^{j+1}} \quad (5.44)$$

Propriedade 5.5 Caso o intervalo $0 \leq x \leq x_0$ no qual a $f(x)$ é definida seja normalizado por x_0 , 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, logo, invariante com o deslocamento.

$$f(x) \in V_{2^j} \Leftrightarrow f(x - 2^{-2}k) \in V_{2^j} \quad (5.45)$$

Propriedade 5.6 Ao ser feita uma aproximação, alguma informação da função original $f(x)$ é perdida, e essa perda é inversamente proporcional à resolução. Assim:

$$\lim_{j \rightarrow \infty} V_{2^j} = \bigcup_{j=-\infty}^{+\infty} V_{2^j} = L^2(\mathbb{R}) \quad (5.46)$$

$$\lim_{j \rightarrow -\infty} V_{2^j} = \bigcap_{j=-\infty}^{+\infty} V_{2^j} = \{0\} \quad (5.47)$$

De (5.46) e (5.47), percebe-se que à medida que a resolução aumenta tendendo a $+\infty$, a aproximação converge para o sinal original. Similarmente, à medida que a resolução diminui a aproximação passa a conter cada vez menos informação, convergindo para zero e perdendo os detalhes.

5.9 Aproximação por Multiresolução e Implementação da Transformada TWD (Decomposição ou Análise)

Definição Qualquer conjunto de subespaços vetoriais $\{V_{2^j}\}$ que satisfaça as propriedades 5.1-5.6 é chamado de "aproximação por multiresolução em $L^2(\mathbb{R})$ ". O conjunto de operadores $\{A_{2^j}\}$ associados a este conjunto de espaços vetoriais gera as aproximações de qualquer função $f(x) \in L^2(\mathbb{R})$ em todos os níveis de resolução 2^j , para $j \in \mathbb{Z}$.

Teorema 5.1 Seja $\{V_{2^j}, j \in \mathbb{Z}\}$ uma aproximação por multiresolução em $L^2(\mathbb{R})$. 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 $\phi(x) \in L^2(\mathbb{R})$ responsável pela formação de todas as bases ortonormais de todos os subespaços, denominada **função-escala** [73]:

$$\phi_{j,n}(x) = 2^{j/2} \phi(2^j x - n) \quad , \quad n \in \mathbb{Z} \tag{5.48}$$

e essa família, por meio da variação de n , é uma base ortonormal para cada V_{2^j} . Essa família será referenciada ao longo do texto simplesmente como $\{\phi_{j,n}\}$.

O Teorema 5.1 mostra que $\{V_{2^j}, j \in \mathbb{Z}\}$ é completamente caracterizada pela função-escala $\phi(x)$ que deve ser continuamente diferenciável e cujo decaimento assintótico e de sua derivada $\phi'(x)$ no infinito deve satisfazer as seguintes relações:

$$\phi(x) = O(x^{-2}) \quad \text{e} \quad |\phi'(x)| = O(x^{-2}) \tag{5.49}$$

A relação entre $\{\phi_{j,n}\}$ e V_{2^j} pode ser exemplificada na Figura 5.14 e combinando cada valor de j com alguns valores de n obtêm-se duas bases ortonormais B_1, B_2 . As duas bases respectivamente, dos dois subespaços vetoriais do exemplo, são geradas pela mesma função-escala $\phi(x)$. [76]

$$\bullet \quad j = 0, \left\{ \begin{array}{c} \vdots \\ n_1 = 0 \Rightarrow \phi_{0,0} = \phi(x) \\ n_2 = 1 \Rightarrow \phi_{0,1} = \phi(x-1) \\ n_3 = 2 \Rightarrow \phi_{0,2} = \phi(x-2) \\ \vdots \\ B_2 \end{array} \right. \quad \text{e} \quad \bullet \quad j = -1, \left\{ \begin{array}{c} \vdots \\ n_1 = 0 \Rightarrow \phi_{-1,0} = 1/\sqrt{2} \cdot \phi(2^{-1}x) \\ n_2 = 1 \Rightarrow \phi_{-1,1} = 1/\sqrt{2} \cdot \phi(2^{-1}x - 1) \\ n_3 = 2 \Rightarrow \phi_{-1,2} = 1/\sqrt{2} \cdot \phi(2^{-1}x - 2) \\ \vdots \\ B_3 \end{array} \right. \tag{5.50}$$

5.9.1 O Sinal de Aproximação ($A_{2^j}^d \cdot f$)

A aproximação (projeção ortogonal de $f(x)$) no subespaço vetorial V_{2^j} pode ser obtida por meio da decomposição dessa função na base ortonormal definida no Teorema 5.1:

$$\forall f(x) \in L^2(\mathbb{R}), \quad A_{2^j} \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f, \phi_{j,n} \rangle \cdot \phi_{j,n}(x) \quad (5.51)$$

e coeficientes dessa aproximação podem ser obtidos em função dos seguintes produtos internos:

$$A_{2^j}^d \cdot f = \left\{ \langle f, \phi_{j,n} \rangle, n \in \mathbb{Z} \right\} \quad (5.52)$$

onde $A_{2^j}^d \cdot f$ é a aproximação discreta de $f(x)$ na resolução 2^j . [73]

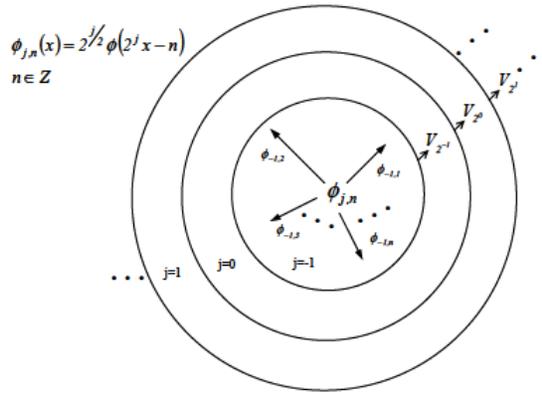


Figura 5.14 – Esquema ilustrativo da relação entre $\phi_{j,n}(x)$ e V_{2^j} para $j \in [-1, 1]$. [76]

A aproximação discreta de $f(x)$, $A_{2^j}^d \cdot f$, possui resolução máxima finita que, por questão de normalização, será considerada igual a 1. A propriedade 5.3 diz que a partir de $A_1^d \cdot f$ pode-se calcular todas as aproximações discretas $A_{2^j}^d \cdot f$ para $j < 0$.

Seja $\{V_{2^j}, j \in \mathbb{Z}\}$ uma aproximação por multiresolução e sua função-escala $\phi(x)$. De acordo com o teorema 5.1, a família $\{\phi_{j,n}\}$ é uma base ortonormal para cada V_{2^j} [73][78]. Sabe-se também que $V_{2^j} \subset V_{2^{j+1}}$. Pode-se então expandir $\{\phi_{j,n}\}$ na base ortonormal de $V_{2^{j+1}}$:

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{j,n}, \phi_{j+1,k} \rangle \phi_{j+1,k}(x) \quad (5.53)$$

Fazendo a mudança de variáveis $u = 2^{j-1}y + n/2^j$ no produto interno de (5.53), tem-se:

$$\begin{aligned} \langle \phi_{j,n}, \phi_{j+1,k} \rangle &= \langle 2^{j/2} \phi(2^j u - n), 2^{(j+1)/2} \phi(2^{j+1} u - k) \rangle = \\ &= 2^{j/2 + (j+1)/2} \int_{-\infty}^{\infty} \phi(2^{-1} y) \cdot \phi(y - (k - 2n)) 2^{-j-1} dy = 2^{-1/2} \langle \phi(2^{-1} u), \phi(u - (k - 2n)) \rangle \end{aligned} \quad (5.54)$$

Assim, retornando à (5.53) tem-se:

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \phi_{j+1,n}(x) \quad (5.55)$$

Sabe-se que todas as aproximações do sinal discreto original podem ser obtidas por meio de diferentes taxas de subamostragem do mesmo. A análise multiresolução foca em obter aproximações do sinal discreto original a partir de outras aproximações. Assim, calculando cada coeficiente resultante da projeção de $f(x)$ por meio da aplicação do produto interno de $f(x)$ em (5.53), tem-se:

$$\langle f, \phi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} \langle \phi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \langle f(u), \phi_{j+1,k}(u) \rangle \quad (5.56)$$

Definindo $h(m)$ como sendo o filtro discreto de resposta impulsiva em (5.56):

$$h(m) = \langle \phi_{-1,0}(u), \phi_{0,m}(u) \rangle \quad \forall n \in \mathbb{Z} \quad (5.57)$$

Reescrevendo (5.53):

$$\phi_{j,n}(x) = \sum_{k=-\infty}^{\infty} h(k-2n) \cdot \phi_{j+1,k}(x) \quad (5.58)$$

os produtos internos de (5.58) em (5.56) provêm a aproximação discreta na escala 2^{-1} , esquematizada na Figura 5.15.a:

$$\langle f, \phi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} h(k-2n) \cdot \langle f, \phi_{j+1,k} \rangle \quad (5.59)$$

Para n fixo, (5.59) gera um coeficiente da projeção de $f(x)$ na resolução 2^j a partir de todos os coeficientes da projeção de $f(x)$ na resolução 2^{j+1} . Logo, (5.58) mostra que $A_{2^j}^d \cdot f$ pode ser calculada por meio da convolução de $A_{2^{j+1}}^d \cdot f$ com $h(-n)$ e subamostrando o resultado por 2 (Figura 5.15.a) [76].

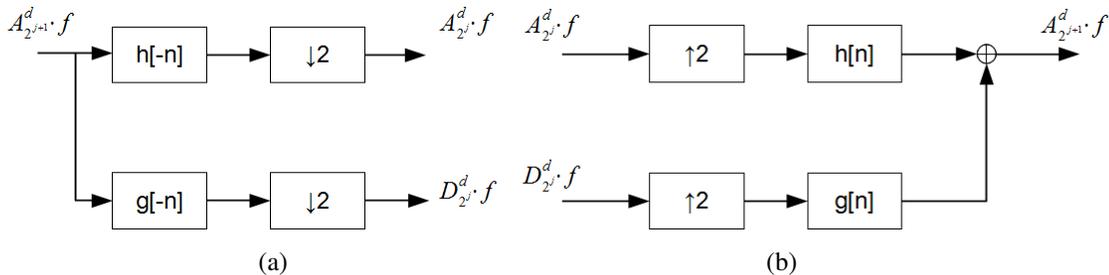


Figura 5.15 – Diagrama em blocos: (a) decomposição; (b) síntese [76]

Com essa codificação piramidal [79], todas as aproximações $A_{2^j}^d \cdot f$ para $j < 0$ podem ser

obtidas recursivamente a partir de $A_1^d \cdot f$. Se a aproximação discreta de $f(x)$ no maior nível de resolução tiver N coeficientes: $A_1^d \cdot f = \alpha_n$ para $1 \leq n \leq N$; então, cada aproximação discreta $A_{2^j}^d \cdot f$ para $j < 0$ terá $2^j \cdot N$ amostras (coeficientes).

A Figura 5.16 apresenta um exemplo da aplicação de (5.59) gerando os coeficientes da aproximação para várias escalas diferentes a partir de um mesmo sinal contínuo. Essas aproximações podem ser calculadas a partir de $f(x)$ por (5.59) ou uma a partir da outra (sempre no sentido da maior para a menor resolução) aplicando (5.58).

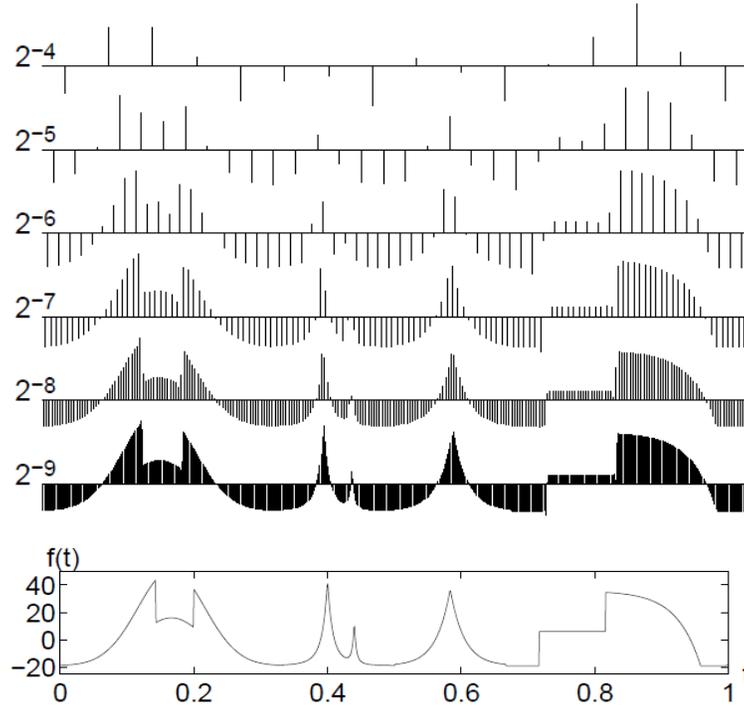


Figura 5.16 – Aproximações multiresolução discretas $A_{2^j}^d \cdot f$ nas escalas 2^{-j} calculadas com *wavelets splines* cúbicas.

Para obter a função-escala $\phi(x)$ basta fazer $j = n = 0$ em (5.58):

$$\phi(x) = \sum_{k=-\infty}^{\infty} h(k) \cdot 2^{1/2} \phi(2x - k) \quad (5.60)$$

A TF de (5.60) é obtida recursivamente da seguinte forma:

$$\phi(\mathbf{w}) = \sum_{k=-\infty}^{\infty} h(k) \frac{\phi(0,5\mathbf{w})}{2^{1/2}} e^{-i0,5\mathbf{w}k} = \frac{\phi(0,5\mathbf{w})}{2^{1/2}} H(0,5\mathbf{w}) \quad (5.61)$$

$$\phi(\mathbf{w}) = \phi(0) \cdot \prod_{m=1}^{\infty} \frac{H(\mathbf{w} \cdot 2^{-m})}{2^{1/2}} \quad (5.62)$$

A família de funções $\{\phi_{j,n}\}$ não são *wavelets*, uma vez que não obedecem à condição de

admissibilidade de (5.6), pois para qualquer análise multiresolução, $\phi(0)=1$ [80]. Em [77], também se mostra que se $h(k)$ é não-nulo apenas para $0 \leq k \leq N-1$ (filtro FIR), então $\phi(x)$ é não-nula somente para $0 \leq x \leq N-1$. Para obter $\phi(x)$ bem localizada deve-se impor um conjunto de restrições ao filtro $h(k)$.

$H(w)$, (TF de $h(n)$) de (5.57) deve satisfazer as propriedades 5.1 a 5.6 [78][80], configurando-se como um filtro passa-baixas.

$$H(w) = \sum_{n=-\infty}^{\infty} h(n) e^{-inw} \quad (5.61)$$

$$H(0) = \sum_{k=-\infty}^{\infty} h(k) = \sqrt{2}. \quad (5.62)$$

$$h(n) = O(x^{-2}) \quad x \rightarrow \infty \quad (5.65)$$

$$|H(w)|^2 + |H(w + \pi)|^2 = 2, \quad \forall w \quad (5.66)$$

$$|H(w)| \neq 0 \quad \text{para } w \in \left[0, \frac{\pi}{2}\right] \quad (5.67)$$

5.9.2 O Sinal de Detalhamento ($D_{2^j}^d \cdot f$)

As aproximações de $f(x)$ nas escalas 2^j e 2^{j-1} são, respectivamente iguais às suas projeções ortogonais em V_{2^j} e $V_{2^{j+1}}$ [55]. Sabendo que V_{2^j} está contido em $V_{2^{j+1}}$, seja W_{2^j} o subespaço vetorial que é o complemento ortogonal do subespaço vetorial V_{2^j} em $V_{2^{j+1}}$, ou seja:

$$V_{2^j} \oplus W_{2^j} = V_{2^{j+1}} \quad (5.68)$$

a projeção ortogonal de $f(x)$ em $V_{2^{j+1}}$ pode ser decomposta como a soma das projeções ortogonais em V_{2^j} e W_{2^j} :

$$A_{2^{j+1}} f(x) = A_{2^j} f(x) + D_{2^j} f(x) \quad (5.69)$$

O complemento $D_{2^j} f(x)$ provê os detalhes de $f(x)$ que aparecem na escala 2^{j-1} (resolução e 2^{j+1}), mas que desaparecem na escala mais “grosseira” e 2^j (resolução 2^j) [55]. O sinal que contém a diferença de informação entre os níveis de resolução 2^j e 2^{j+1} é chamado de “detalhe” no nível de resolução 2^j . O sinal de detalhamento em 2^j é obtido por meio da projeção ortogonal de $f(x)$ em W_{2^j} .

Para tanto, precisa-se encontrar, uma função $\psi(x)$ capaz de compor uma base ortonormal para

gerar o subespaço W_{2^j} . Assumindo a existência de uma família $\{\psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n), n \in \mathbb{Z}\}$, referenciada como $\{\psi_{j,n}\}$, verifica-se se ela é uma base ortonormal para W_{2^j} [72][73].

Para qualquer $n \in \mathbb{Z}$, $\psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n)$ pertence a $W_{2^j} \subset V_{2^{j+1}}$. Pode-se então expandir $\psi_{j,n}$ na base ortonormal de $V_{2^{j+1}}$:

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \psi_{j,n}, \phi_{j+1,k} \rangle \cdot \phi_{j+1,k}(x) \quad (5.70)$$

Fazendo a mudança de variáveis no produto interno de (5.70), tem-se analogamente à subseção anterior:

$$\begin{aligned} \langle \psi_{j,n}, \phi_{j+1,k} \rangle &= 2^{j/2-1} \int_{-\infty}^{\infty} \psi(2^{-j} y) \cdot \phi(y - (k - 2n)) 2^{-j-1} dy \\ &= 2^{-1/2} \langle \psi(2^{-1} u), \phi(u - (k - 2n)) \rangle \end{aligned} \quad (5.71)$$

Assim, retornando à (5.70) tem-se:

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} \langle \psi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \phi_{j+1,k}(x) \quad (5.72)$$

Calculando-se cada coeficiente resultante da projeção de $f(x)$ através da aplicação do produto interno de $f(x)$ com ambos os lados de (5.70), tem-se:

$$\langle f, \psi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} \langle \psi_{-1,0}(u), \phi_{0,k-2n}(u) \rangle \cdot \langle f(u), \phi_{j+1,k}(u) \rangle \quad (5.73)$$

Definindo $g(n)$ como sendo o filtro de resposta impulsiva em (5.74):

$$g(m) = \langle \psi_{-1,0}(u), \phi_{0,m}(u) \rangle \quad \forall m \in \mathbb{Z} \quad (5.74)$$

Reescrevendo (5.72):

$$\psi_{j,n}(x) = \sum_{k=-\infty}^{\infty} g(k - 2n) \cdot \phi_{j+1,k}(x) \quad (5.75)$$

e os produtos internos de (5.73) como (5.75) providenciam os detalhes na escala 2^{-j} esquematizado no segundo ramo da Figura 5.15.a.

$$\langle f, \psi_{j,n} \rangle = \sum_{k=-\infty}^{\infty} g(k - 2n) \cdot \langle f, \phi_{j+1,k} \rangle \quad (5.76)$$

Para se obter $\psi(x)$ basta fazer $j = n = 0$ em (5.75), resultando em:

$$\psi(x) = \sum_{k=-\infty}^{\infty} g(k) \cdot 2^{1/2} \phi(2x - k) \quad (5.77)$$

Tomando-se a TF de ambos os lados de (5.77), tem-se:

$$\psi(w) = \sum_{k=-\infty}^{\infty} g(k) \cdot \frac{\phi(0,5w)}{2^{1/2}} e^{-i0,5wk} = \frac{\phi(0,5w)}{2^{1/2}} G(0,5w) \quad (5.78)$$

Deve-se notar que, diferentemente de $\phi(w)$, não há recursão para $\psi(w)$. $G(w)$ deve satisfazer as condições (5.79), (5.80) e (5.82):

$$|G(w)|^2 + |G(w + \pi)|^2 = 2, \quad \forall w \quad [80] \quad (5.79)$$

Porat [80] também coloca que $G(w)$ e $H(w)$ satisfazem à seguinte relação:

$$H(w)G^*(w) + H(w + \pi)G^*(w + \pi) = 0 \quad (5.80)$$

Testando a condição de admissibilidade para $\psi(w)$, substituindo (5.64) em (5.66) e (5.64) e (5.81) em (5.80):

$$|H(0)|^2 + |H(\pi)|^2 = 2 + |H(\pi)|^2 = 2 \Rightarrow |H(\pi)|^2 = 0 \quad (5.81)$$

$$G(0) = 0 \Rightarrow \psi(0) = \frac{\phi(0)}{2^{1/2}} \cdot G(0) \Rightarrow \psi(0) = 0 \quad (5.82)$$

Logo, $\psi(w)$ satisfaz a condição de admissibilidade. Assim, a família $\{\psi_{j,n}\}$ é uma base *wavelet*.

Para se obter a relação entre os filtros $g(n)$ e $h(n)$, define-se:

$$H_a = H(w), \quad G_a = G(w), \quad H_b = H(w + \pi), \quad G_b = G(w + \pi) \quad (5.83)$$

Substituindo (5.83) em (5.80):

$$G_a / G_b = - \left(H_b / H_a \right)^* \quad (5.84)$$

Após manipulações matemáticas, chega-se a:

$$G_a = - \frac{|H_a|G_b}{|G_b|H_a^*} \cdot H_b^* = -A(w) \cdot H_b^* \quad (5.85)$$

onde $A(w)$ é uma função passa-tudo, uma vez que $|A(w)| = 1, \forall w$.

Conseqüentemente, a primeira relação entre os filtros $|G_a|=|H_b|$ nos diz que a magnitude do espectro de $g(n)$ é igual ao espectro de $h(n)$ deslocado de π . Logo, $G(w)$ é um filtro passa-altas.

Substituindo (5.85) em (5.80):

$$\begin{aligned} H(w) \left[-A(w)H^*(w + \pi) \right]^* + H(w + \pi) \left[-A(w + \pi)H^*(w + 2\pi) \right]^* &= 0 \\ A(w) + A(w + \pi) &= 0 \end{aligned} \quad (5.86)$$

A condição de admissibilidade está embutida em (5.83) (5.86). Frequentemente se escolhe $A(w)=e^{-iw}$, escolha que obedece à $|A(w)|=1$ e à (5.86). Substituindo essa escolha em (5.85) tem-se:

$$\begin{aligned} G(w) &= e^{-iw} \cdot H^*(w + \pi) \\ g(n) &= (-1)^{n-1} \cdot h^*(1-n) \end{aligned} \quad (5.87)$$

que é a relação entre os filtros $g(n)$ e $h(n)$ utilizados no processo de decomposição e síntese da TWD [72]. Qualquer par de filtros que obedeça às condições apresentadas nesta seção pode ser utilizado, mas é usual escolher $h(n)$ como sendo passa-baixas e $g(n)$ automaticamente um passa-altas. Os filtros espelho são peça chave no algoritmo rápido da TWD (FWT - *Fast Wavelet Transform*) [55].

Note-se que determinando $g(n)$ e $h(n)$, são determinadas $\psi(x)$ e $\phi(x)$.

Teorema 5.2: Seja $\{V_{2^j}\}$, $\phi(x)$, $h(n)$ e seja $\psi(x)$ a função cuja TF é dada por:

$$\Psi(w) = G(0,5w)\phi(0,5w)2^{-1/2} \quad (5.88)$$

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

$$\psi_{j,n}(x) = 2^{j/2} \cdot \psi(2^j x - n), \quad n \in \mathbb{Z} \quad (5.89)$$

chamada no texto de $\{\psi_{j,n}\}$ é uma base ortonormal em W_{2^j} [72].

A partir de um par de filtros que satisfaçam (5.64)-(5.67), pode-se encontrar a função-escala e a *wavelet*-mãe a partir de (5.64) e (5.88), respectivamente. Normalmente é mais fácil começar com $h(n)$. Se $h(n)$ tem somente um número finito de entradas não-nulas, então $\phi(x)$, $\psi(x)$ e as *wavelets*-filhotes resultantes terão todas elas, suporte compacto [68].

Seja D_{2^j} o operador de projeção ortogonal no subespaço vetorial W_{2^j} , o sinal de detalhamento de $f(x)$, $D_{2^j} \cdot f(x)$, com resolução 2^j é dado por:

$$D_{2^j} \cdot f(x) = \sum_{n=-\infty}^{\infty} \langle f(u), \psi_{j,n}(u) \rangle \cdot \psi_{j,n}(x) \quad (5.90)$$

e coeficientes desse sinal de detalhamento são obtidos em função dos seguintes produtos internos:

$$D_{2^j}^d \cdot f = \left\{ \langle f, \psi_{j,n} \rangle, n \in \mathbb{Z} \right\} \quad (5.91)$$

onde $D_{2^j}^d \cdot f$ contém a diferença de informação entre $A_{2^{j+1}}^d \cdot f$ e $A_{2^j}^d \cdot f$

De (5.76) nota-se que é possível calcular o sinal de detalhamento $D_{2^j}^d \cdot f$ por meio da

convolução de $A_{2^{j+1}}^d \cdot f$ com o filtro discreto $g(-n)$, e aplicando dizimação por um fator de dois no resultado, conforme esquematizado no ramo inferior da Figura 5.15.a. O processo completo é definido por (5.52) e (5.91).

5.9.3 A Decomposição Completa

Assim, a expansão de um sinal $f(x)$ numa base *wavelet* ortogonal pode ser visto como uma agregação de detalhes nas escalas 2^j que vão de 1 a ∞ .

$$f(x) = \sum_{j=-\infty}^{\infty} A_{2^j}^d f(x) = \sum_{j=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \langle f, \psi_{j,n} \rangle \psi_{j,n} \quad (5.92)$$

A Figura 5.17 apresenta como exemplo os coeficientes do sinal decomposto nas bases ortogonais *wavelet spline* cúbicas. Observa-se o sinal de aproximação $A_{2^j}^d \cdot f$ para $j=5$ ao topo nessa figura e os sinais de detalhes $D_{2^j}^d \cdot f$ para várias escalas diferentes nas demais curvas; todos para o mesmo sinal contínuo. Os detalhes podem ser calculados a partir de $f(x)$ por (5.91) ou uma a partir da outra (sempre no sentido da maior para a menor resolução) aplicando (5.76).

A representação *wavelet* ortogonal do sinal discreto $A_j^d \cdot f$ pode ser calculada por meio de sucessivas decomposições de $A_{2^{j+1}}^d \cdot f$ em $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$, sendo $-J \leq j \leq -1$:

$$\left(A_{2^{-J}}^d \cdot f, \{ D_{2^j}^d \cdot f \}_{-J \leq j \leq -1} \right) \quad (5.93)$$

ou seja, composta pela aproximação $A_{2^{-J}}^d \cdot f$ com resolução mais “grosseira” (“DC”) e pelos detalhes $D_{2^j}^d \cdot f$ com resoluções 2^j (“ACs”), sendo $-J \leq j \leq -1$.

Se o sinal original possui N amostras, então os sinais discretos $D_{2^j}^d \cdot f$ e $A_{2^j}^d \cdot f$ possuem $2^j \cdot N$ amostras cada. Assim, a representação *wavelet* (5.95) apresenta o mesmo número de amostras que $A_j^d \cdot f$, graças à ortogonalidade da representação. Observa-se, portanto, o algoritmo *herringbone* de Mallat, introduzido na seção 5.6.2.

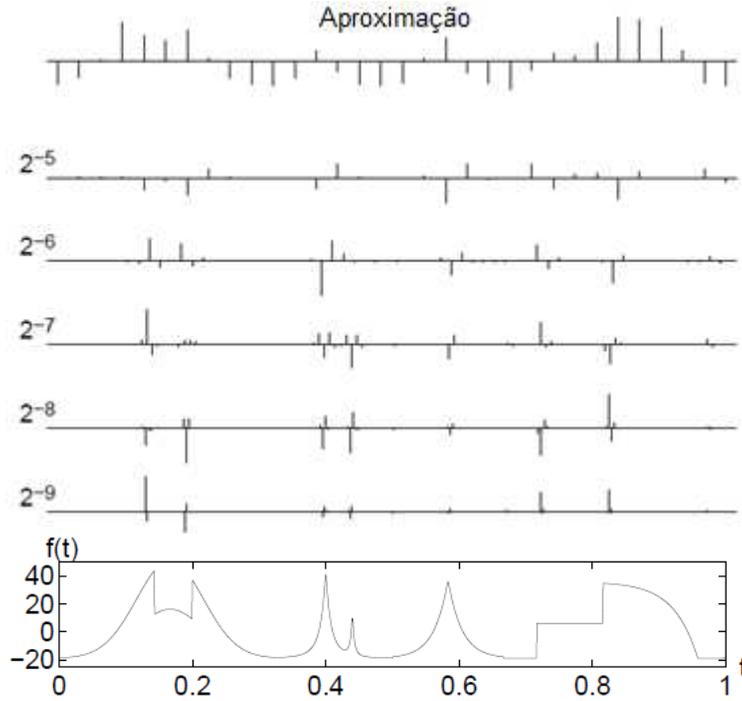


Figura 5.17 – Coeficientes *wavelet* calculados nas escalas 2^{-j} com *wavelets splines* cúbicas (aproximação $A_{2^j}^d \cdot f$ para $j=5$ e detalhes $D_{2^j}^d \cdot f$ [55])

5.10 Reconstrução ou Síntese

Como W_{2^j} é o complemento ortogonal de V_{2^j} em $V_{2^{j+1}}$, $(\{\phi_{j,n}(x)\}, \{\psi_{j,n}(x)\})_{n \in \mathbb{Z}}$ é uma base ortonormal em $V_{2^{j+1}}$. Logo, tem-se:

$$\phi_{j+1,n}(x) = \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \phi_{j,k} \rangle \cdot \phi_{j,k}(x) + \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \psi_{j,k} \rangle \cdot \psi_{j,k}(x) \quad (5.94)$$

$$\langle f(x), \phi_{j+1,n}(x) \rangle = \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \phi_{j,k} \rangle \cdot \langle f(x), \phi_{j,k}(x) \rangle + \sum_{k=-\infty}^{\infty} \langle \phi_{j+1,n}, \psi_{j,k} \rangle \cdot \langle f(x), \psi_{j,k}(x) \rangle \quad (5.95)$$

Fazendo-se uma mudança de variáveis em (5.95), obtém-se:

$$\langle \phi_{j+1,n}(u), \phi_{j,k}(u) \rangle = \langle \phi_{-1,0}(u), \phi_{0,n-2k}(u) \rangle = h(n-2k) \quad (5.96)$$

$$\langle \phi_{j+1,n}(u), \psi_{j,k}(u) \rangle = \langle \psi_{-1,0}(u), \phi_{0,n-2k}(u) \rangle = g(n-2k) \quad (5.97)$$

$$\underbrace{\langle f(x), \phi_{j+1,n}(x) \rangle}_{\text{componente de } A_{2^{j+1}}^d \cdot f} = \sum_{k=-\infty}^{\infty} h(n-2k) \cdot \underbrace{\langle f(x), \phi_{j,k}(x) \rangle}_{\text{componente de } A_{2^j}^d \cdot f} + \sum_{k=-\infty}^{\infty} g(n-2k) \cdot \underbrace{\langle f(x), \psi_{j,k}(x) \rangle}_{\text{componente de } D_{2^j}^d \cdot f} \quad (5.98)$$

$A_{2^{j+1}}^d \cdot f$ pode ser construída por superamostragem de $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$ e convolução dos

resultados, respectivamente, com $h(n)$ e $g(n)$. Cada coeficiente de $A_{2^{j+1}}^d \cdot f$ é formado por *todos* os coeficientes de $A_{2^j}^d \cdot f$ e $D_{2^j}^d \cdot f$ (Figura 5.15.b).

Comparando as Figuras. 5.15.a e 5.15.b, pode-se observar que no algoritmo de decomposição utiliza-se $h(-n)$ e $g(-n)$, enquanto na síntese utilizam-se filtros de resposta impulsiva $h(n)$ e $g(n)$.

5.11 Possibilidades de Implementações da TWD [76]

Colocando a *wavelet*-básica (que obedeça a condição de admissibilidade) e variando j e n , pode-se implementar a TWD diretamente [68], não havendo necessidade de se fazer o cálculo de $h(n)$, $g(n)$ ou da função-escala.:

$$W_{j,n} = \sum_i f(i\Delta t) \cdot \psi_{j,n}(i\Delta t) \quad (5.99)$$

$$f(i\Delta t) = \sum_{j,n} W_{j,n} \cdot \psi_{j,n}(i\Delta t) \quad (5.100)$$

Contudo, uma escolha arbitrária pode levar às *wavelets* sem o comportamento desejado, não sendo possível obter a representação desejada ou clara o suficiente para dada aplicação. 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 achando $\phi(x)$, $g(n)$ e $\psi(x)$, direcionando as características.

Uma vez determinada a *wavelet*-básica do sistema, pode-se alternativamente implementar a TWD através do algoritmo *herringbone* (FWT) da seção 5.6.2, que é computacionalmente mais eficiente [68]. O algoritmo *herringbone* não manipula as *wavelets* e funções-escala explicitamente, mas sim, as projeções de $f(x)$ na hierarquia de subespaços. O filtro $h(-n)$ via equação (5.58) mapeia a projeção de $f(x)$ no subespaço $V_{2^{j+1}}$ para a projeção no subespaço V_{2^j} . Similarmente, o filtro $g(-n)$ via (5.76) mapeia a projeção de $f(x)$ no subespaço $V_{2^{j+1}}$ para a projeção no subespaço W_{2^j} . Desse modo, as saídas do esquema da Figura. 5.12 podem ser vistas em (5.93).

5.11.1 Bases Wavelet Separáveis

A qualquer base *wavelet* ortonormal $\{ \psi_{j,n}(x, y) \}_{(j,n) \in \mathbb{Z}^2}$ de $L^2(\mathbb{R})$, pode-se associar uma base *wavelet* ortonormal separável de $L^2(\mathbb{R}^2)$:

$$\{ \psi_{j_1:n_x}(x) \psi_{j_2:n_y}(y) \}_{(j_1, j_2, n_x, n_y) \in \mathbb{Z}^4} \quad (5.101)$$

As funções $\psi_{j_1:n_x}(x) \psi_{j_2:n_y}(y)$ misturam informações em duas diferentes escalas 2^{j_1} e 2^{j_2} ao

longo de x e y , aspecto freqüentemente desejável de ser evitado [55]. A multiresolução separável leva a outra construção de bases *wavelet* separáveis com *wavelets* que são produtos de funções dilatadas de mesma escala. Essas aproximações multiresolução também têm importantes aplicações em visão computacional, onde são usadas para processar imagens em diferentes níveis de detalhes. Imagens de resolução mais baixa são representadas por um número menor de *pixels* e podem portar informação suficiente para a realização da tarefa de reconhecimento [55].

Decomposições do sinal em bases *wavelet* separáveis são calculadas com uma extensão separável de algoritmos de bancos de filtros. Bases *wavelet* não-separáveis também podem ser construídas [81][82], mas são menos utilizadas no processamento de imagens, dada a elevação na complexidade computacional. Em [55], exemplos de bases *wavelet* biortogonais *quincux* não-separáveis são apresentados.

5.12 Aproximação por Multiresolução e Implementação da Transformada TWD 2D (Decomposição ou Análise): Multiresoluções Separáveis

Analogamente ao caso 1D, a noção de resolução é formalizada com projeções ortogonais em espaços de várias dimensões. A aproximação de uma imagem $f(x,y)$ na resolução 2^j é definida como a projeção ortogonal de $f(x,y)$ no subespaço vetorial $V_{2^j}^2$ que está incluído no $L^2(R^2)$. O espaço $V_{2^j}^2$ é o conjunto de todas as aproximações na resolução 2^j . Quando a resolução decresce, a dimensão de $V_{2^j}^2$ também decresce.

A definição formal da aproximação por multiresolução $\{V_{2^j}^2\}_{j \in \mathbb{Z}}$ de $L^2(R^2)$ é uma extensão da multiresolução de $L^2(R)$. A mesma causalidade, completude, e propriedades de escalonamento devem ser satisfeitas [55].

Considere-se agora, o caso de multiresolução separável, seja $\{V_{2^j}^1\}, j \in \mathbb{Z}$, ou simplesmente $\{V_{2^j}\}$ a multiresolução de $L^2(R)$ dada na seção 5.9. Uma multiresolução 2D separável é composta pelo conjunto de subespaços vetoriais $\{V_{2^j}^2\}_{j \in \mathbb{Z}}$, sendo que cada $V_{2^j}^2$ é definido como o produto tensorial dado por:

$$V_{2^j}^2 = V_{2^j}^1 \otimes V_{2^j}^1 \quad (5.102)$$

Qualquer conjunto de subespaços vetoriais $\{V_{2^j}^2, j \in \mathbb{Z}\}$ que satisfaçam as propriedades 1-6 da subseção 5.8.2 para o caso bidimensional é chamado de "aproximação por multiresolução em

$L^2(\mathbb{R}^2)$ ". Ao longo do texto, esse conjunto será referenciado simplesmente por $\{V_{2^j}^2\}$, de forma a diferenciá-lo de cada $V_{2^j}^2$ em particular.

As funções $f(x,y)$ são expansões lineares de funções separáveis:

$$f(x, y) = \sum_{m=-\infty}^{\infty} a[m] f_m(x) g_m(y), \text{ com } f_m \in V_j, g_m \in V_j \quad (5.103)$$

Sendo $\phi(x)$ a função-escala 1D, pode-se mostrar que há uma única função-escala $\phi(x,y) \in L^2(\mathbb{R}^2)$ responsável pela formação de todas as bases ortogonais de todos os subespaços em questão [73]. Assim, uma base ortonormal para cada $V_{2^j}^2$ é formada pela família $\{\phi_{j;n_x;n_y}(x, y)\}$:

$$\phi_{j;n_x;n_y}(x, y) = 2^j \phi(2^j x - n_x, 2^j y - n_y), \quad (n_x, n_y) \in \mathbb{Z}^2 \quad (5.104)$$

Mallat [83] mostra o caso em que $\phi(x,y)$ ortogonal é separável:

$$\phi(x, y) = \phi(x) \cdot \phi(y) \quad (5.105)$$

O mesmo ocorre para a base ortogonal em $V_{2^j}^2$:

$$\phi_{j;n_x;n_y}(x, y) = \phi_{j;n_x}(x) \cdot \phi_{j;n_y}(y), \quad (n_x, n_y) \in \mathbb{Z}^2 \quad (5.106)$$

5.12.1 O Sinal de Aproximação 2D ($A_{2^j}^d \cdot f$)

Logo, a aproximação (projeção ortogonal de uma imagem $f(x,y)$) no subespaço vetorial $V_{2^j}^2$ pode ser obtida por meio da decomposição :

$$A_{2^j} \cdot f(x, y) = \sum_{n_x=-\infty}^{\infty} \sum_{n_y=-\infty}^{\infty} \langle f(u, v), \phi_{j;n_x;n_y}(u, v) \rangle \cdot \phi_{j;n_x;n_y}(x, y) \quad (5.107)$$

e os coeficientes dessa aproximação podem ser obtidos em função dos seguintes produtos internos:

$$A_{2^j} \cdot f = \left\{ \langle f(u, v), \phi_{j;n_x}(u) \cdot \phi_{j;n_y}(v) \rangle, \quad (n_x, n_y) \in \mathbb{Z}^2 \right\} \quad (5.108)$$

onde (5.108) é a aproximação discreta de $f(x,y)$ na resolução 2^j .

Considerem-se imagens discretas, com resolução máxima finita igual a 1. Analogamente ao caso 1D, de acordo com a extensão do teorema 5.1, a família $\{\phi_{j;n_x;n_y}(x, y)\}$ é uma base ortonormal para cada $V_{2^j}^2$. Como $\{\phi_{j;n_x;n_y}(x, y)\} \in V_{2^j}^2 \subset V_{2^{j+1}}^2$, pode-se expandir $\{\phi_{j;n_x;n_y}(x, y)\}$ na base ortonormal de $V_{2^{j+1}}^2$:

$$\phi_{j;n_x;n_y}(x, y) = \sum_{k_x=-\infty}^{\infty} \sum_{k_y=-\infty}^{\infty} \langle \phi_{j;n_x;n_y}(u, v), \phi_{j+1;k_x;k_y}(u, v) \rangle \cdot \phi_{j+1;k_x;k_y}(x, y) \quad (5.109)$$

Fazendo uma mudança de variáveis, tem-se:

$$\langle f(x, y) \cdot \phi_{j;n_x;n_y}(x, y) \rangle = \sum_{k_x=-\infty}^{\infty} \sum_{k_y=-\infty}^{\infty} \langle \phi_{-1;0;0}(u, v), \phi_{0;(k_x-2n_x);(k_y-2n_y)}(u, v) \rangle \cdot \langle f(x, y) \cdot \phi_{j+1;k_x;k_y}(x, y) \rangle \quad (5.110)$$

Define-se $h(l, m)$ como sendo o filtro discreto de resposta impulsiva em (5.111):

$$h(l, m) = \langle \phi_{-1;0;0}(u, v), \phi_{-1;l;m}(u, v) \rangle = \langle \phi_{-1;0;0}(u), \phi_{0;l}(u) \rangle \cdot \langle \phi_{-1;0}(v), \phi_{0;m}(v) \rangle = h(l) \cdot h(m) \quad (5.111)$$

$$\langle f(x, y) \cdot \phi_{j;n_x;n_y}(x, y) \rangle = \sum_{k_x=-\infty}^{\infty} h(k_x - 2n_x) \cdot \sum_{k_y=-\infty}^{\infty} h(k_y - 2n_y) \cdot \langle f(x, y) \cdot \phi_{j+1;k_x;k_y}(x, y) \rangle \quad (5.112)$$

Para (n_x, n_y) fixo, (5.114) gera cada coeficiente da projeção de $f(x, y)$ na resolução 2^j a partir de todos os coeficientes da projeção de $f(x, y)$ na resolução 2^{j+1} (ramo superior da Figura 5.18). $A_{2^j}^d \cdot f$ pode ser calculada por meio de filtragens sucessivas de $A_{2^{j+1}}^d \cdot f$ em ambas as direções. Primeiramente, $A_{2^{j+1}}^d \cdot f$ passa por filtragem unidimensional horizontal por $h(-n)$ (direção das linhas), e subamostragem do resultado por um fator de 2 nessa direção.

A seguir, o resultado passa por filtragem unidimensional vertical por $h(-n)$ (direção das colunas) com o mesmo $h(-n)$, seguido por subamostragem por um fator de 2 na direção vertical. Alternativamente, pode-se realizar primeiramente o processamento por colunas e depois por linhas.

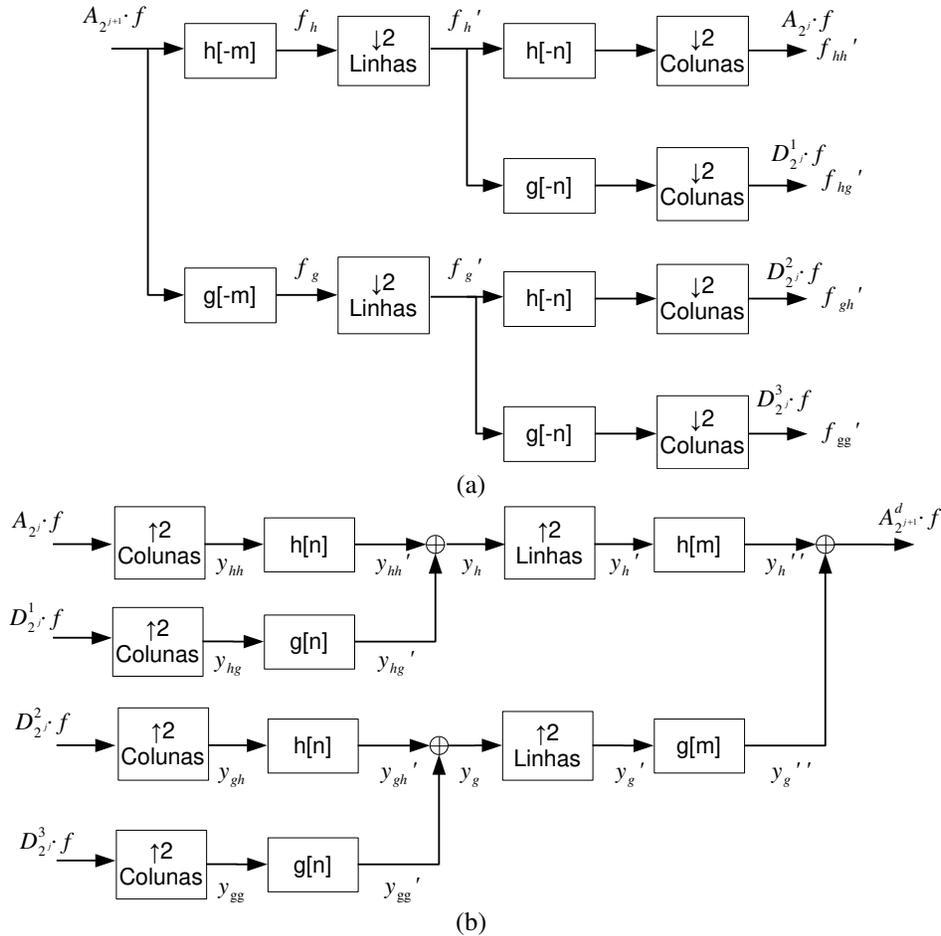


Figura 5.18 – Diagrama em blocos para imagem de: (a) um estágio de decomposição wavelet; (b) um estágio de reconstrução wavelet.

5.12.2 O Sinal de Detalhamento 2D ($D_{2^j}^d \cdot f$)

Sendo $\{V_{2^j}^2\}$ a aproximação por multiresolução separável em $L^2(\mathbb{R}^2)$, o detalhe no nível de resolução 2^j é a projeção de $f(x,y)$ no subespaço vetorial $W_{2^j}^2$ que, analogamente ao caso 1D, é o complemento ortogonal de $V_{2^j}^2$ em $V_{2^{j+1}}^2$.

Teorema 5.3: Sendo a função-escala 2D separável $\phi(x,y)$ e sendo $\psi(x)$ a wavelet associada à $\phi(x)$, há 3 wavelets responsáveis pela formação das bases ortonormais [68][73]:

$$\begin{aligned} \psi^1(x, y) &= \phi(x) \cdot \psi(y) \\ \psi^2(x, y) &= \psi(x) \cdot \phi(y) \\ \psi^3(x, y) &= \psi(x) \cdot \psi(y) \end{aligned} \tag{5.113}$$

onde i indica o índice da wavelet utilizada. Assim, a família $\{\psi_{j;n_x,n_y}^i(x, y), 1 \leq i \leq 3\}$ dada em (5.114) é uma base ortonormal para cada $W_{2^j}^2$.

$$\left\{ \begin{array}{l} \psi_{j;n_x;n_y}^1(x, y) = 2^j \psi^1(2^j x - n_x, 2^j y - n_y) \\ \psi_{j;n_x;n_y}^2(x, y) = 2^j \psi^2(2^j x - n_x, 2^j y - n_y) \\ \psi_{j;n_x;n_y}^3(x, y) = 2^j \psi^3(2^j x - n_x, 2^j y - n_y) \end{array} \right\}, \quad (n_x, n_y) \in \mathbb{Z}^2 \quad (5.114)$$

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 $W_{2^j}^2$ [72], gerando as subimagens de detalhes:

$$\begin{aligned} D_{2^j}^1 \cdot f &= \left\langle f(x, y), \psi_{j;n_x;n_y}^1(x, y) \right\rangle, \quad (n_x, n_y) \in \mathbb{Z}^2 \\ D_{2^j}^2 \cdot f &= \left\langle f(x, y), \psi_{j;n_x;n_y}^2(x, y) \right\rangle, \quad (n_x, n_y) \in \mathbb{Z}^2 \\ D_{2^j}^3 \cdot f &= \left\langle f(x, y), \psi_{j;n_x;n_y}^3(x, y) \right\rangle, \quad (n_x, n_y) \in \mathbb{Z}^2 \end{aligned} \quad (5.115)$$

Assumindo a existência do conjunto de *wavelets* $\left\{ \psi_{j;n_x;n_y}^i(x, y), 1 \leq i \leq 3 \right\}$ separáveis, cada $\left\{ \psi_{j;n_x;n_y}^i(x, y) \right\}$ pertence ao espaço $W_{2^j}^2 \subset V_{2^{j+1}}^2$. Logo, fazendo a expansão:

$$\psi_{j;n_x;n_y}^i(x, y) = \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \left\langle \psi_{j;n_x;n_y}^i(u, v), \phi_{j+1;k_x;k_y}(u, v) \right\rangle \cdot \phi_{j+1;k_x;k_y}(x, y) \quad (5.116)$$

Fazendo uma mudança de variáveis, tem-se:

$$\psi_{j;n_x;n_y}^i(x, y) = \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \left\langle \psi_{-1;0;0}^i(u, v), \phi_{0;(k_x-2n_x),(k_y-2n_y)}(u, v) \right\rangle \cdot \phi_{j+1;k_x;k_y}(x, y) \quad (5.117)$$

Definindo o filtro de resposta impulsiva:

$$g^i(l, m) = \left\langle \psi_{-1;0;0}^i(u, v), \phi_{0;l;m}(u, v) \right\rangle \quad (5.118)$$

para cada *wavelet* i em (5.113), existe um filtro separável correspondente que pode ser decomposto no produto de dois filtros 1D definidos em (5.57) e (5.74). Logo:

$$g^1(l, m) = \left\langle \psi_{-1,0}(u) \psi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \right\rangle = \left\langle \psi_{-1,0}(u), \phi_{0,l}(u) \right\rangle \cdot \left\langle \psi_{-1,0}(v), \phi_{0,m}(v) \right\rangle = h(l) \cdot g(m)$$

$$g^2(l, m) = \left\langle \psi_{-1,0}(u) \phi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \right\rangle = \left\langle \psi_{-1,0}(u), \phi_{0,l}(u) \right\rangle \cdot \left\langle \phi_{-1,0}(v), \phi_{0,m}(v) \right\rangle = g(l) \cdot h(m) \quad (5.119)$$

$$g^3(l, m) = \left\langle \psi_{-1,0}(u) \psi_{-1,0}(v), \phi_{0,l}(u) \phi_{0,m}(v) \right\rangle = \left\langle \psi_{-1,0}(u), \phi_{0,l}(u) \right\rangle \cdot \left\langle \psi_{-1,0}(v), \phi_{0,m}(v) \right\rangle = g(l) \cdot g(m)$$

Obtendo-se o processo de filtragem em (5.120-5.122), tem-se:

$$\left\langle f(x, y) \cdot \psi_{j;n_x;n_y}^1(x, y) \right\rangle = \sum_{k_y=-\infty}^{\infty} g(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} h(k_x - 2n_x) \cdot \left\langle f(x, y) \cdot \phi_{j+1;n_x;n_y}(x, y) \right\rangle \quad (5.120)$$

$$\langle f(x, y) \cdot \psi_{j;n_x,n_y}^2(x, y) \rangle = \sum_{k_y=-\infty}^{\infty} h(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} g(k_x - 2n_x) \cdot \langle f(x, y) \cdot \phi_{j+1;n_x,n_y}(x, y) \rangle \quad (5.121)$$

$$\langle f(x, y) \cdot \psi_{j;n_x,n_y}^3(x, y) \rangle = \sum_{k_y=-\infty}^{\infty} g(k_y - 2n_y) \cdot \sum_{k_x=-\infty}^{\infty} g(k_x - 2n_x) \cdot \langle f(x, y) \cdot \phi_{j+1;n_x,n_y}(x, y) \rangle \quad (5.122)$$

As subimagens de detalhes $D_{2^j}^d \cdot f$ podem ser calculadas por meio de filtragens sucessivas de $A_{2^{j+1}}^d \cdot f$ em ambas as direções. Primeiramente $A_{2^{j+1}}^d \cdot f$ passa por filtragem horizontal unidimensional (linhas) e subamostragem por um fator de 2 nessa direção.

O resultado passa então por filtragem unidimensional vertical (colunas), seguido por subamostragem por 2. O filtro utilizado em cada direção depende do índice i da *wavelet* em questão (demais ramos da Figura 5.18)

5.12.3 A Decomposição 2D Completa

A representação *wavelet* ortogonal 2D de uma imagem original $A_1^d \cdot f$ pode ser calculada por meio 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 $D_{2^j}^3 \cdot f$, sendo $-J \leq j \leq -1$. Um estágio completo desse algoritmo encontra-se na Figura 5.19. A representação *wavelet* 2D pode, então, ser calculada por meio do cascadeamento de dois algoritmos piramidais 1D: o primeiro aplicado nas linhas da imagem, e o segundo aplicado às colunas [76].

Para qualquer $J > 0$, a representação *wavelet* 2D ortogonal de uma imagem $A_1^d \cdot f(x, y)$ é dada pelas $3J+1$ imagens discretas em (5.123), exemplificada para $j=3$ na Figura 5.20:

$$A_{2^{-j}}^d \cdot f, \left\{ D_{2^j}^1 \cdot f \right\}_{-J \leq j \leq -1}, \left\{ D_{2^j}^2 \cdot f \right\}_{-J \leq j \leq -1}, \left\{ D_{2^j}^3 \cdot f \right\}_{-J \leq j \leq -1} \quad (5.123)$$

A imagem $A_{2^{-j}}^d \cdot f$ é a aproximação de $A_1^d \cdot f$ com nível de resolução 2^{-j} , correspondendo às frequências mais baixas de $f(x, y)$. $\left\{ D_{2^j}^i \cdot f \right\}_{1 \leq i \leq 3}$ são as imagens de detalhamento para diferentes níveis de resoluções onde: $D_{2^j}^1 \cdot f$ corresponde às frequências verticais altas (linhas horizontais na imagem), $D_{2^j}^2 \cdot f$ corresponde às frequências horizontais altas (linhas verticais na imagem) e $D_{2^j}^3 \cdot f$ corresponde às frequências altas em ambas as direções. Se a imagem original possui N pixels, cada imagem $A_{2^{-j}}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$ e $D_{2^j}^3 \cdot f$ da representação possui $2^{2j} \cdot N$ pixels, totalizando os mesmos N pixels da imagem original [76].

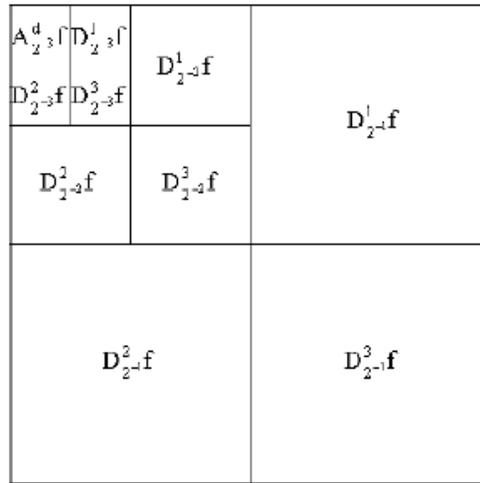


Figura 5.19 – Decomposição *wavelet* ortogonal 2D para $J = 3$ [76].

Na Figura 5.20 é apresentado um exemplo evidenciando essa característica das frequências verticais, horizontais e diagonais aparecendo nas subimagens separadamente.

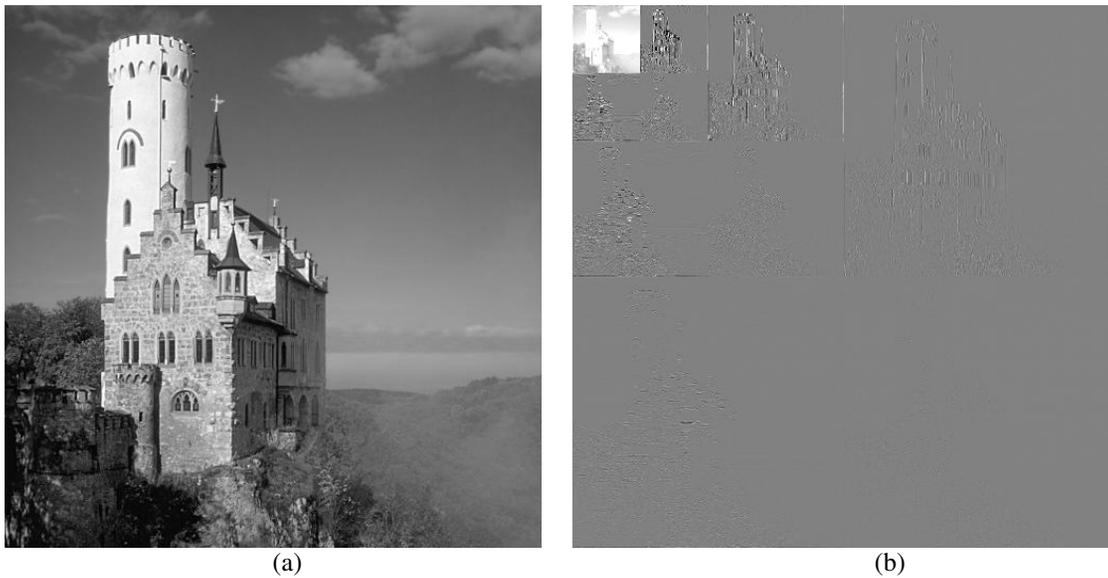


Figura 5.20 - Decomposição *wavelet* Daubechies separável (passa-alta foi ressaltada para comparação) (a) original (b) decomposição *wavelet*

A Figura 5.21 também evidencia essa característica das frequências verticais, horizontais e diagonais por meio da apresentação da TF das *wavelets* separáveis e funções escala calculadas de uma *wavelet* Daubechies separável.

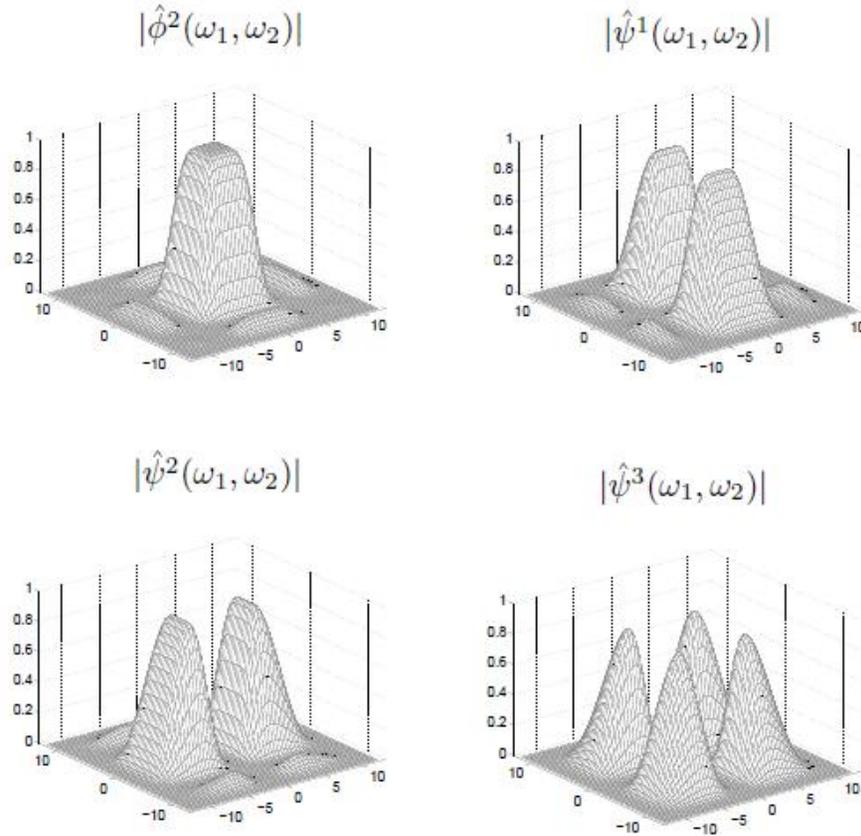


Figura 5.21 - TF de uma função escala separável e das 3 wavelets separáveis correspondentes calculadas para uma wavelet de Daubechies [55].

5.13 Reconstrução ou Síntese Ortogonal 2D

Como $W_{2^j}^2$ é o complemento ortogonal de $V_{2^j}^2$ em $V_{2^{j+1}}^2$, o conjunto de funções $(\{\phi_{j;n_x,n_y}(x,y)\} \cup \{\psi_{j;n_x,n_y}^i(x,y), 1 \leq i \leq 3\})_{(n_x,n_y) \in \mathbb{Z}}$ é a base ortonormal em $V_{2^{j+1}}^2$. Logo:

$$\begin{aligned} \phi_{j+1;n_x,n_y}(x,y) &= \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \phi_{j;k_x,k_y}(u,v) \rangle \cdot \phi_{j;k_x,k_y}(x,y) + \\ &+ \sum_{i=1}^3 \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \psi_{j;k_x,k_y}^i(u,v) \rangle \cdot \psi_{j;k_x,k_y}^i(x,y) \end{aligned} \quad (5.124)$$

$$\begin{aligned} \langle f(x,y), \phi_{j+1;n_x,n_y}(x,y) \rangle &= \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \phi_{j;k_x,k_y}(u,v) \rangle \cdot \langle f(x,y), \phi_{j;k_x,k_y}(x,y) \rangle + \\ &+ \sum_{i=1}^3 \sum_{k_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \langle \phi_{j+1;n_x,n_y}(u,v), \psi_{j;k_x,k_y}^i(u,v) \rangle \cdot \langle f(x,y), \psi_{j;k_x,k_y}^i(x,y) \rangle \end{aligned} \quad (5.125)$$

e:

$$\begin{aligned}
\langle f(x, y) \cdot \phi_{j+1; n_x, n_y}(x, y) \rangle &= \sum_{k_y=-\infty}^{\infty} h(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} h(n_x - 2k_x) \cdot \langle f(x, y), \phi_{j; k_x, k_y}(x, y) \rangle + \\
&+ \sum_{k_y=-\infty}^{\infty} g(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} h(n_x - 2k_x) \cdot \langle f(x, y), \psi_{j; k_x, k_y}^1(x, y) \rangle + \\
&+ \sum_{k_y=-\infty}^{\infty} h(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} g(n_x - 2k_x) \cdot \langle f(x, y), \psi_{j; k_x, k_y}^2(x, y) \rangle + \\
&+ \sum_{k_y=-\infty}^{\infty} g(n_y - 2k_y) \sum_{k_x=-\infty}^{\infty} g(n_x - 2k_x) \cdot \langle f(x, y), \psi_{j; k_x, k_y}^3(x, y) \rangle
\end{aligned} \tag{5.126}$$

$A_{2^{j+1}} \cdot f$ pode ser reconstruído fazendo a inserção de uma coluna de zeros entre cada coluna de $A_{2^j}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$, $D_{2^j}^3 \cdot f$ seguida pela convolução das linhas resultantes com um filtro 1D. A seguir, há a inserção de uma linha de zeros entre cada linha, seguida pela convolução das colunas resultantes com outro filtro 1D. Os filtros utilizados foram previamente definidos em (5.57) e (5.74). Cada coeficiente de $A_{2^{j+1}}^d \cdot f$ é formado por todos os coeficientes de $A_{2^j}^d \cdot f$, $D_{2^j}^1 \cdot f$, $D_{2^j}^2 \cdot f$ e $D_{2^j}^3 \cdot f$ (Figura 5.18.b). A imagem $A_1^d \cdot f$ é reconstruída repetindo-se esse processo para $-1 \leq j \leq -J$.

5.14 Comentários

Este capítulo apresentou a implementação da TWD ortogonal 2D, por meio da análise multiresolução, gerando os sinais passa-baixas (aproximação) e de passa-altas (detalhes) bidimensionais. No capítulo 6 serão mostradas desvantagens da TWD em representar contornos das imagens e novas formas de aproximação são apresentadas, com um intuito de se obter representações esparsas (poucos coeficientes para armazenar a informação).

Capítulo 6

Novas Formas de Representação Esparsa

Em processamento ou compressão de imagens digitais, é fundamental converter o sinal original para uma nova forma de representação, que destaque suas características e isole componentes a serem processadas separadamente. Em geral, deseja-se obter uma boa representação de um sinal com relação a um determinado funcional ou função de custo, que relacione a qualidade do sinal com o número de componentes necessárias para sua reconstrução. Representações esparsas são aquelas em que a maior parte das componentes é nula ou insignificante. Essas componentes podem ser desprezadas na reconstrução sem grandes perdas.

Para compressão de imagens com mínimas perdas, são desejáveis aproximações que representem bem, aspectos visuais e consigam boa reconstrução em termos de erro quadrático médio (*MSE – Mean Square Error*) ou norma L^2 :

$$\|f\|^2 = \sum_{k=1}^K |f[k]|^2 \quad (6.1)$$

$$MSE = \|erro\|^2 = \|f - \hat{f}\|^2 = \frac{1}{K} \sum_{k=1}^K |f[k] - \hat{f}[k]|^2 \quad (6.2)$$

onde $\hat{f}[k]$ representam as amostras aproximadas do sinal f .

Com esse objetivo, as transformadas *wavelet* têm sido muito estudadas em teorias de aproximações. Uma aproximação linear clássica é a estabelecida pelo teorema da amostragem (Nyquist-Shannon), representando um sinal analógico por vetores de valores bem definidos em frequência que formam uma base escolhida a priori, desde que a frequência de amostragem seja maior do que 2 vezes a maior frequência presente no sinal original.

$$\begin{aligned} x[n] &= x(nT) \\ x(t) &= \sum_{n=-\infty}^{+\infty} x[n] \operatorname{sinc}\left(\frac{t - nT}{T}\right) \end{aligned} \quad (6.3)$$

onde $T = 1/f_s$ é o intervalo de amostragem, f_s é a frequência de amostragem e $\text{sinc}(x)$ é a função sinc normalizada:

$$\text{sinc}(x) = \frac{\text{sen}(\pi x)}{\pi x}$$

A representação discreta do sinal pode ser ainda mais reduzida do que a amostragem definida acima, escolhendo os vetores de acordo com o sinal. Tradicionalmente são utilizadas transformadas ortogonais ou biortogonais para obter a representação esparsa, como no caso da TWD apresentada no capítulo 5.

No entanto, as transformadas ortogonais têm limitações na capacidade de concentrar dados em poucas componentes para sinais com descontinuidades, como imagens do tipo *cartoon*. O erro de aproximação depende da regularidade do sinal tratado. Para sinais suaves, aproximações lineares e não-lineares têm desempenho similar [55]. Quando a regularidade do sinal não é uniforme, aproximações não-lineares em bases *wavelet* são mais apropriadas. Para aproximar funções suaves por partes e com geometria não-separável, autores têm recorrido a esquemas de quantização vetorial e dicionários redundantes, que garantem representação mais esparsa, como será visto a seguir.

Essas novas formas de representação esparsa serão apresentadas neste capítulo juntamente com a motivação para usá-las considerando as limitações das transformadas *wavelet* ortogonais isotrópicas.

6.1 Aproximações Lineares

Seja $\{\varphi_n(x) = \varphi_s(x - ns)\}_{0 \leq n < N}$ uma base ortonormal de um espaço $A_N \subset L^2[0,1]$. Considerando um sinal analógico $f(x)$ de suporte compacto normalizado em $[0, 1]$, a melhor aproximação de f em A_N é a projeção ortogonal f_N de f em A_N , recuperado pela base $\{\varphi_n(x)\}_{0 \leq n < N}$.

$$f_N(x) = \sum_{n=0}^{N-1} \langle f, \varphi_n \rangle \varphi_n(x) \quad (6.4)$$

A questão central é determinar quantos coeficientes *wavelet* são necessários para representar um determinado sinal até um dado erro de aproximação tolerado. Em aproximações lineares, as N *wavelets* em A_N são escolhidas a priori representando a N -ésima escala $A_N = \text{span}\{\varphi_{n,-N}\}_{n=0}^{N-1}$.

Para estimar o erro de aproximação $\|f - f_N\|$, define-se uma base ortonormal $B = \{g_m\}_{m \in \mathbb{N}}$ de $L^2[0,1]$ com N vetores $\{g_m\}_{0 \leq m < N}$ formando uma base ortogonal do mesmo espaço A_N . Como mostrado em [55], aproximações utilizando bases de Fourier ou *wavelets* são ótimas para sinais que pertençam a espaços definidos por:

$$\mathbf{W}_{B,s} = \left\{ f \in \mathbf{H} : \sum_{m=0}^{+\infty} m^{2s} |\langle f, g_m \rangle|^2 < +\infty \right\} \quad (6.5)$$

Quando B é uma base de Fourier ou *wavelets*, a equação acima define $\mathbf{W}_{B,s}$ como espaços de Sobolev:

$$\mathbf{W}_s = \left\{ f \in \mathbf{H} : \sum_{m=0}^{+\infty} \left| \frac{\partial^k f}{\partial x^k} \right|^2 < +\infty, 0 \leq k \leq s \right\} \quad (6.6)$$

Como mostrado no Teorema 6.1 [55], o erro provocado por essas aproximações segue $\varepsilon_l(N, f) = O(N^{-2s})$, onde N é o número de componentes utilizadas.

Teorema 6.1: Para algum $s > 1/2$, existem $A, B > 0$ tais que $\sum_{m=0}^{+\infty} |m|^{2s} |\langle f, g_m \rangle|^2 < +\infty$, então

$$A \sum_{m=0}^{+\infty} m^{2s} |\langle f, g_m \rangle|^2 \leq \sum_{N=0}^{+\infty} N^{2s-1} \varepsilon_l(N, f) \leq B \sum_{m=0}^{+\infty} m^{2s} |\langle f, g_m \rangle|^2 \quad (6.7)$$

Para sinais mais complexos, que possuam descontinuidades bem localizadas, a aproximação linear de f pelas primeiras N componentes não é necessariamente a melhor aproximação. Nesse caso, $f \notin \mathbf{W}^2[0,1]$ para qualquer $s > 1/2$, portanto $\varepsilon_l(N, f)$ decai com $O(N^{-1})$. A Figura 6.1 [55] mostra um exemplo de sinal descontínuo com variação limitada aproximado pelas componentes de alta escala da decomposição *wavelet*. Nota-se a existência de erros nas proximidades das descontinuidades, devido à insuficiência de componentes para representá-las. Na Figura 6.1.a, a aproximação é dada pelas $N = 128$ componentes Fourier de mais baixa frequência com relação sinal-ruído de $8,63 \cdot 10^{-2}$. Na Figura 6.1.b, a aproximação é dada pelos coeficientes *wavelet* Daubechies 4 de maior escala com $N = 128$ e a relação sinal-ruído é de $8,58 \cdot 10^{-2}$.

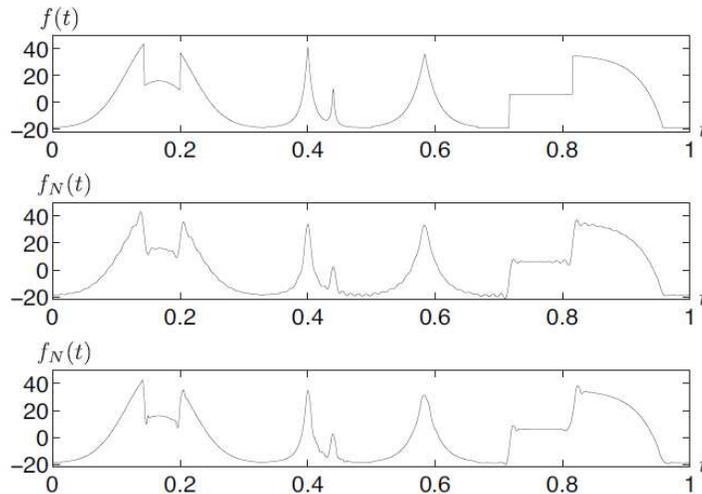


Figura 6.1 – Reconstrução de um sinal unidimensional de variação limitada (a) sinal original (b) aproximação Fourier (c) aproximação por *wavelet* Daubechies 4 [55]

Para sinais estacionários, modelados por um processo aleatório de média nula, a base que minimiza o erro quadrático médio para aproximações lineares é a transformada de Karhunen-Loève (KLT). Caso o sinal f não tenha média nula, subtrai-se o valor da média amostral $E[f]$ antes da transformação e adiciona novamente após a transformação inversa. Essa transformada concentra toda a energia de um sinal de dimensões finitas em poucos coeficientes bem localizados (correspondente a uma matriz diagonal), de forma a diagonalizar a matriz de covariância do sinal.

$$\text{Cov}(F) = \Gamma^{*T} \text{Cov}(f) \Gamma^* \quad (6.8)$$

onde $()^T$ indica a matriz transposta e $()^*$ o operador complexo conjugado.

A transformação Γ que satisfaz a relação (6.8) é a transformada cujos vetores-base (amostras da função-base) são os autovetores da matriz de covariância e, por consequência, os elementos da diagonal da matriz de covariância transformada são os autovalores da matriz de dados original. Em termos práticos, essa transformada é difícil de ser implementada, pois necessita que sejam calculados os autovetores da matriz $\text{Cov}(f)$ a cada nova amostra de imagem e, dependendo da dimensão dos dados, pode consumir muito tempo computacional. Transformadas com coeficientes fixos (como a DCT) são muito mais fáceis de serem implementadas e atingem resultados muito próximos à KLT. Além disso, quando f não é um processo estacionário gaussiano (ou não é em todo domínio e possui descontinuidades), esse tipo de transformada (bem como outras funções estacionárias como bases de Fourier e transformadas de cossenos) não é eficiente, pois os autovetores da matriz de covariância têm a mesma ordem de magnitude e f não terá a energia concentrada em poucos coeficientes. Nesses casos, são mais adequadas aproximações não-lineares, como é apresentado na seção 6.2.

6.2 Aproximações Não-Lineares

Em aproximações não-lineares, A_N consiste de N *wavelets* em diferentes escalas e localizações. Considerando uma função de valores reais f sobre R^n que satisfaz a condição de Hölder quando existem constantes reais não-negativas C, α , tais que $\forall x, y \in \mathbb{R}^n$,

$$|f(x) - f(y)| \leq C|x - y|^\alpha, \quad (6.9)$$

o erro de uma aproximação não-linear é calculado a partir dos coeficientes não utilizados na aproximação:

$$\hat{f}(x) = \sum_{m \in A_n} \langle f, g_m \rangle g_m(x) \quad (6.10)$$

$$\|f - \hat{f}\|^2 = \sum_{m \in A_n} |\langle f, g_m \rangle|^2 \quad (6.11)$$

Como visto em (6.10), a não-linearidade está no método de escolha dos coeficientes wavelet pelos maiores, sendo portanto adaptativo ao sinal analisado. Modelando o sinal como uma função suave por partes, pode-se mostrar que *wavelets* oferecem a melhor taxa de aproximações não-lineares. Aproximando funções que são localmente Hölder- α com descontinuidades pelos N maiores coeficientes, obtém-se uma taxa de erro $O(N^{-2\alpha})$ e esse é um resultado ótimo [84].

Considerando um espaço parametrizado por k^{-1} , então o erro $\varepsilon_{nl}(N, f)$ da aproximação não-linear segue [55]:

$$y = x + k^{-1}$$

$$|f(x) - f_k(x)| \leq Ck^{-\alpha} \quad (6.12)$$

$$\varepsilon_{nl}(N, f) = \|f - f_k\|^2 \leq \frac{C^2}{2\alpha - 1} N^{1-2\alpha} \quad (6.13)$$

$$\varepsilon_{nl}(N, f) = O(N^{-2\alpha}) \quad (6.14)$$

A Figura 6.2 mostra a aproximação não-linear *wavelet* de um sinal regular por partes. Impulsos de Dirac representam coeficientes *wavelet* com coeficientes positivos ou negativos de amplitude acima de um dado limiar T . Os maiores coeficientes estão nas proximidades das descontinuidades. Nesse caso, o erro de aproximação é 17 vezes menor do que para aproximações lineares. [55] Na Figura 6.4, é mostrada uma comparação para um sinal de imagem.

Funções de variação limitada (*bounded variation*) são exemplos de sinais nos quais uma aproximação não-linear leva a um erro muito menor do que uma aproximação linear. A norma de variação total é definida por

$$\|f\|_V = \int_0^1 |f'(x)| dx \quad (6.15)$$

A derivada f' deve ser tomada como uma distribuição, de modo a incluir funções descontínuas. Assim, garante-se que um sinal de variação limitada satisfaz a condição de Hölder e pode ser bem representado por uma aproximação não-linear. Quando $f \in \mathbf{BV}[0,1]$ contém descontinuidades, o erro assintótico segue N^{-1} e N^{-2} para a aproximação linear e não-linear, respectivamente. A Figura 6.3, ilustra a aproximação não-linear de um sinal de variação limitada com descontinuidades.

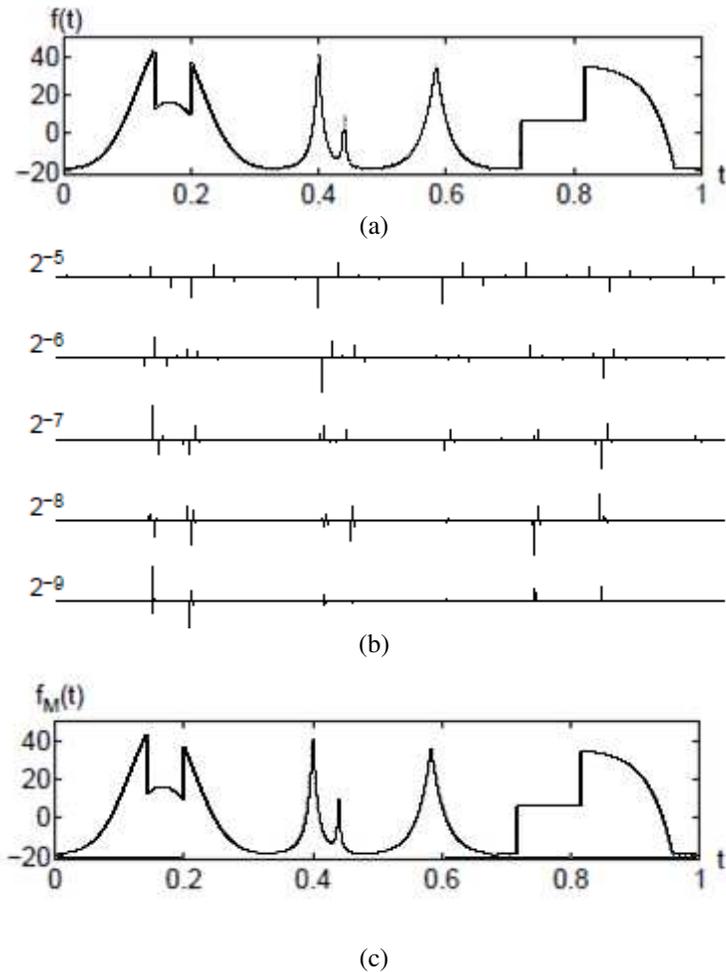


Figura 6.2 – (a) sinal original f (b) decomposição *wavelet*, com os $N/6$ maiores coeficientes *wavelet* Symmetlet 4 (c) aproximação não-linear

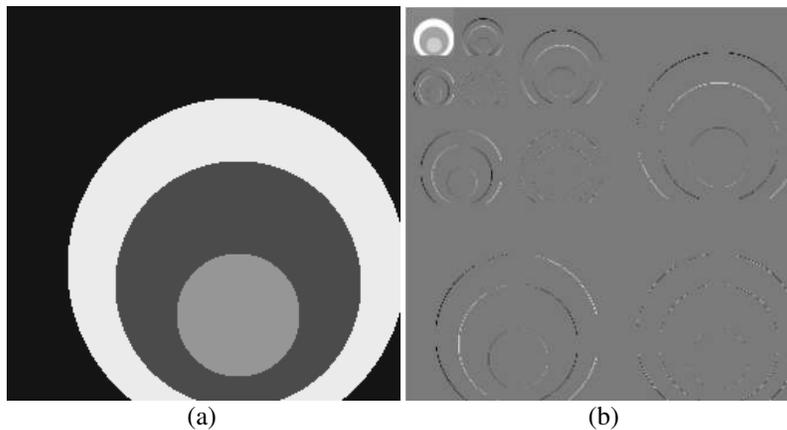


Figura 6.3 – Coeficientes *wavelet* de uma imagem suave com contornos bem definidos. As componentes de alta frequência (detalhes) foram normalizadas para melhor visualização. [45]

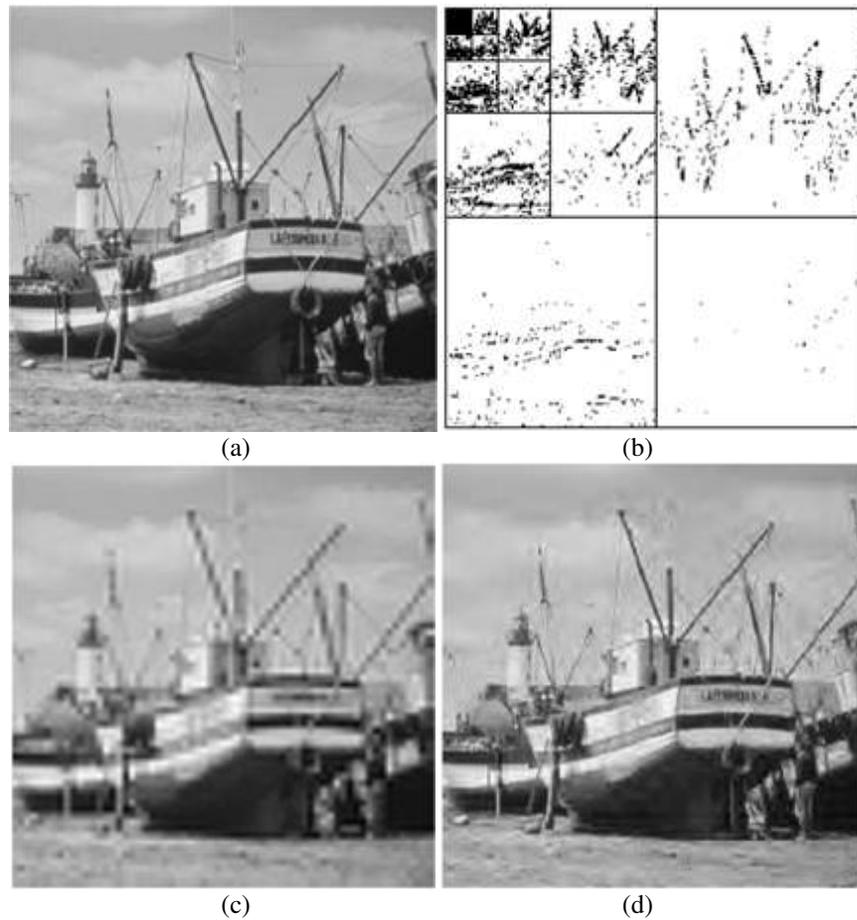


Figura 6.4 – Esquemas de aproximação (a) original ($N = 2562$) (b) decomposição *wavelet*, pontos escuros correspondem a $\left| \langle f, \psi_{j,n}^k \rangle \right| > T$ (c) aproximação linear com $N/16$ (d) aproximação não-linear com $N/16$ [55]

6.3 O Problema das Transformadas Isotrópicas e o Uso de Representações Esparsas

Embora essa situação funcione bem para sinais unidimensionais, problemas surgem quando se tratam duas ou mais dimensões, devido à importância da geometria na qualidade visual. Essas aproximações são ótimas para imagens de variação limitada, mas não para imagens naturais contendo bordas geometricamente regulares. Além disso, *wavelets* d -dimensionais são normalmente construídas com $2^d - 1$ produtos tensoriais de *wavelets* unidimensionais (separáveis). Singularidades multidimensionais podem estar organizadas ao longo de subconjuntos de acordo com a geometria, como acontece com contornos de uma imagem no caso 2D, e portanto podem não ser representadas de maneira eficiente por *wavelets* separáveis. A Figura 6.5 mostra a ineficiência da decomposição *wavelet* em representar contornos por não explorar a regularidade dos mesmos.

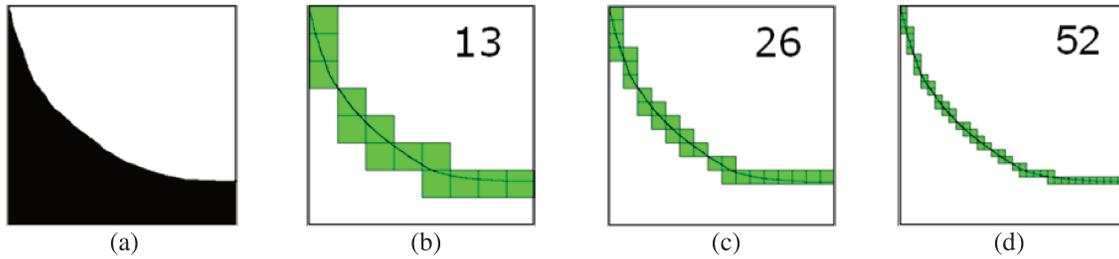


Figura 6.5 Ineficiência da *wavelet* isotrópica em representar a geometria de uma imagem. Número de coeficientes: (a) 13 (b) 26 (c) 52 [85]

Um grande número de coeficientes é necessário para representar uma descontinuidade, que no caso de imagens pode ser reduzido conhecendo-se a geometria. Existe coerência nesses coeficientes que precisa ser preservada quando aplicada a quantização, de forma a evitar artefatos do tipo *ringing* (Figura 6.6).



Figura 6.6 – Artefatos do tipo *ringing* gerados pela quantização inadequada dos coeficientes *wavelet* ao longo de um contorno.

Para mostrar a baixa taxa de decaimento do erro de aproximações em sinais bidimensionais, tem-se na Figura 6.7.a uma imagem simples de variação limitada. A imagem é definida como $f = C \cdot \mathbf{1}_\Omega$ e a parte escura identifica um subconjunto Ω . A Figura 6.7.b mostra os suportes das bases *wavelet* quando é feita a projeção da imagem, com o tamanho dos pontos de acordo com a magnitude dos respectivos coeficientes *wavelet*. Se o suporte da *wavelet* $\psi_{j,n}^l$ não intercepta a borda $\partial\Omega$, então $\langle f, \psi_{j,n}^l \rangle = 0$. As *wavelets* $\psi_{j,n}^l$ têm um suporte quadrado de tamanho proporcional a 2^j , que é mostrado na Figura por uma grade de intervalo 2^j . A borda $\partial\Omega$ tem comprimento finito L e portanto existem da ordem de $L \cdot 2^{-j}$ *wavelets* interceptando $\partial\Omega$. Como f tem variação limitada, $\langle f, \psi_{j,n}^l \rangle = O(C \cdot 2^j)$, sendo $\langle f, \psi_{j,n}^l \rangle \sim C \cdot 2^j$ ao longo da borda. Assim, os M maiores coeficientes estão nas escalas $2^j \geq L/M$, resultando em um erro de aproximação não-linear de:

$$\varepsilon_{nl}(M, f) \sim \sum_{j=-\infty}^{\log_2(L/M)-1} L2^{-j} C^2 2^{2j} = (CL)^2 M^{-1} \quad (6.16)$$

Assim, o número excessivo de coeficientes *wavelet* produzidos ao longo dos contornos limita o decaimento do erro em M^{-1} .

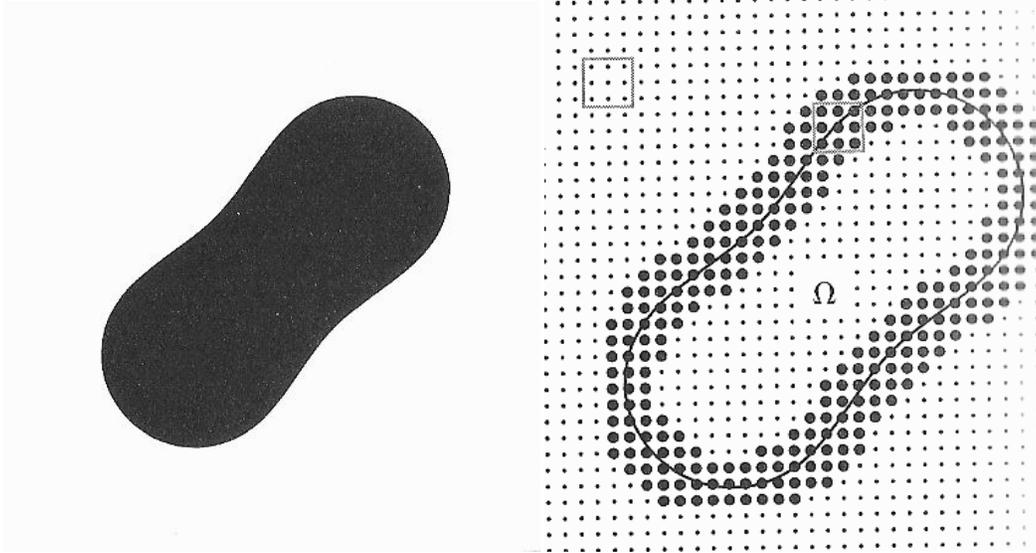


Figura 6.7 – Projeção *wavelet* (a) imagem $f = \mathbf{1}_{\Omega}$ (b) na escala 2^j , as *wavelets* $\psi_{j,n}^l$ têm um suporte quadrado de largura proporcional a 2^j . Esse suporte corresponde a uma grade de intervalo 2^j , indicada pelos pequenos pontos. Pontos maiores indicam *wavelets* com suporte que cruza a fronteira de Ω para $\langle f, \psi_{j,n}^l \rangle \neq \mathbf{0}$

Por essas razões e para tratar a deficiência da TWD pura em explorar a geometria, outros tipos de representação de sinais, incluindo dicionários redundantes, tem sido o foco de novos avanços em codificação de imagem, além do desempenho das *wavelets* ortogonais. Esses avanços se justificam quando são aplicadas etapas severas de quantização (necessárias para altas taxas de compressão) que normalmente geram artefatos pelos algoritmos tradicionais.

6.4 Representações Esparsas e Dicionários Geométricos

Anisotropia é a propriedade de um sistema de transformadas em apresentar variação com a direcionalidade do sinal analisado. Tradicionalmente, em algoritmos de codificação de imagens tem-se usado um produto tensorial entre duas bases unidimensionais para analisar a direção vertical e a horizontal da imagem. Autores mais recentes têm explorado a característica taxa-distorção de sistemas *anisotrópicos*, que consideram bases variantes com a direção e portanto não podem ser decompostas, de forma a representar imagens com contornos bem definidos [45][86][87].

Nikol'skii [88] define espaços de Sobolev anisotrópicos, que indicam um número de derivadas limitado ao longo das direções, em contraste com o espaço de Sobolev tradicional \mathbf{W}_s , que é isotrópico e portanto caracterizam funções com a mesma suavidade (γ e α) em ambas as direções. O

espaço de Sobolev anisotrópico \mathbf{W}_{s_1, s_2} para $d = 2$ é definido como:

$$\mathbf{W}_{s_1, s_2} = \left\{ f(x_1, x_2) : \sum_{x_1=0}^{+\infty} \sum_{x_2=0}^{+\infty} \left| \frac{\partial^{k+m} f}{\partial x_1^k \partial x_2^m} \right|^2 < +\infty, 0 \leq k \leq s_1, 0 \leq m \leq s_2 \right\} \quad (6.17)$$

Espaços de Sobolev anisotrópicos d -dimensionais podem ser definidos de forma similar. Aplicações envolvendo esses espaços são mostradas em [89].

Assim como a *wavelet* tradicional isotrópica analisa blocos quadrados de tamanho diádico (Figura 6.8.c), novas transformadas anisotrópicas foram definidas em domínios diferentes. Para imagens suaves por partes, é provado que aproximações lineares sobre partições de N triângulos, como na Figura 6.8.b [86], podem atingir taxa de decaimento de erro ótima $O(N^{-2})$ como se a imagem não tivesse discontinuidades. No entanto, não existe um algoritmo que garanta a melhor triangulação para todas as imagens suaves por partes, porque a geometria dos contornos e o contraste de embaçamento (*blur*) são desconhecidos. Algoritmos propostos na literatura constroem iterativamente a triangulação de forma a reduzir progressivamente o erro com o aumento do número de triângulos, ou utilizam triangulações densas que podem ser decimadas (reduzidas por fator de dois).

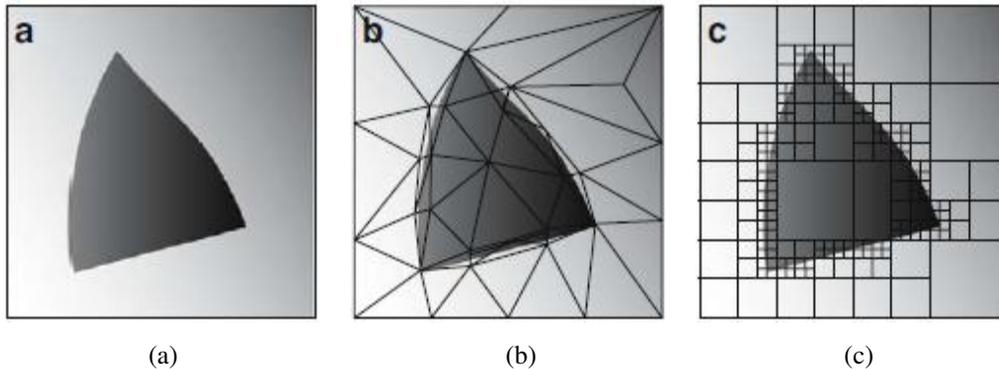


Figura 6.8 – Geometrias de aproximação (a) original (b) aproximação com triangulação (c) aproximação com uma base *wavelet* [86].

Em [87], são também apresentadas transformadas com reconstrução perfeita com momentos nulos direcionais, impostos por funções base bidimensionais ao longo de diferentes direções, batizadas de *directionlets* (Figura 6.9). Nesse esquema, mais componentes de detalhe são obtidas para uma determinada direção. Para que essa aproximação seja efetiva, é necessário um bom algoritmo de codificação que agrupe de forma adequada os coeficientes gerados nas diferentes resoluções e direções, e preserve a regularidade.

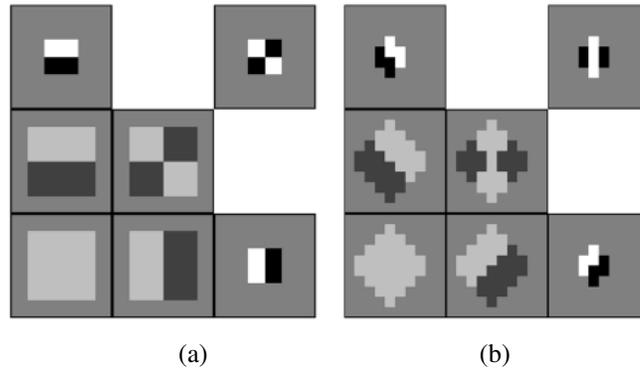


Figura 6.9 – Funções base 2D (a) *wavelet* Haar isotrópica (b) *wavelet* Haar anisotrópica com orientação de 45°.

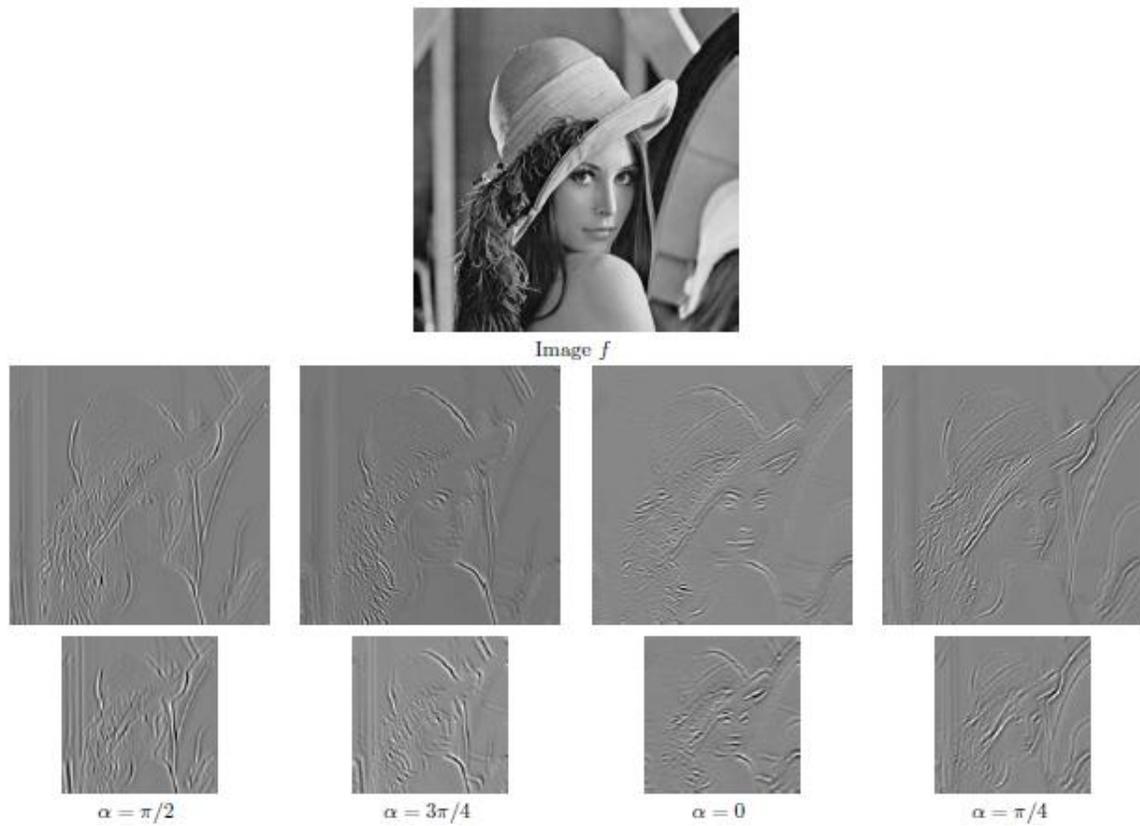


Figura 6.10 – Decomposição *wavelet* direcional com $\alpha = 0, \pi/4, \pi/2, 3\pi/4$ e escalas 2^j e 2^{j+1} [55].

Candes e Donoho [90] propuseram as *curvelets*, definidas como bases (ou frames) em $L^2(\mathbb{R}^2)$ com suporte multi-escala, alongado e rotacionado, garantindo uma taxa de aproximação não-linear $N^{-2} \log(N)$, melhor do que as *wavelets*, para imagens com descontinuidades ao longo de curvas C^2 . Da mesma forma que as *wavelets* direcionais, *curvelets* são obtidas pela rotação, dilatação e translação de formas de onda elementares (bases). No entanto, *curvelet* possui suporte

mais alongado, obtido pelo escalonamento dos parâmetros de largura e altura (Figura 6.11).

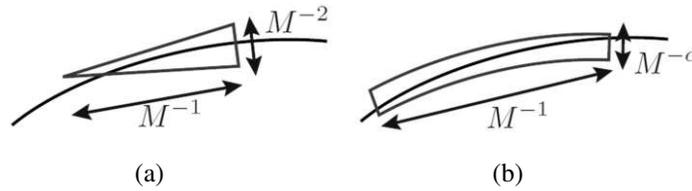


Figura 6.11 – Aproximações em blocos não retangulares (a) triângulo adaptado para aproximar uma curva (b) elemento *curvelet* de aproximação [86]

Le Pennec et al. [91][92][93] propuseram o dicionário de bases *bandelet* ortogonais, tomando vantagem da regularidade geométrica. Esse método retransforma os coeficientes *wavelet* de forma direcional, criando novos momentos nulos nas direções apropriadas, com decaimento do erro em $O(N^{-\alpha})$. Os coeficientes *bandelet* de um sinal f são escritos como:

$$\langle f, \varphi_p \rangle = \sum_n \tilde{\psi}_{i,l,m}[k] \langle f, \psi_{j,n}^k \rangle \quad (6.18)$$

onde $\tilde{\psi}_{i,l,m}[k]$ é uma *wavelet* direcional de comprimento 2^i e largura 2^l , com posição m no vetor de coeficientes *wavelet* e $p = (k, j, l, i, m)$ é a parametrização em posição, escala e tamanho do *bandelet*.

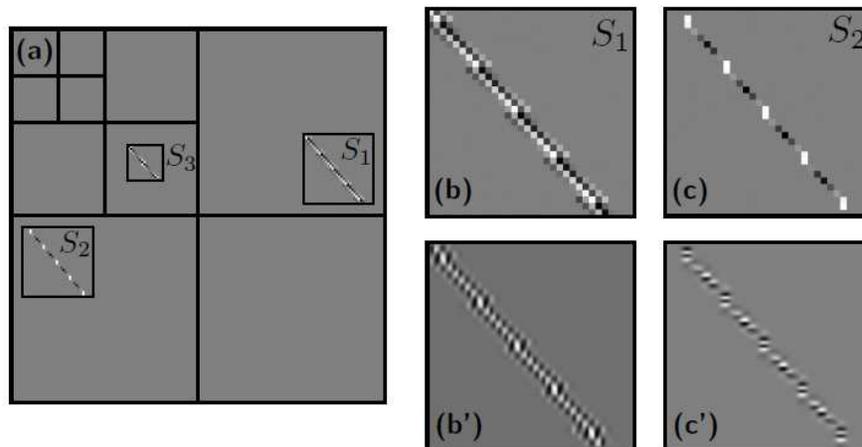


Figura 6.12 – Decomposição *wavelet* direcional (a) decomposição *wavelet* (b-c) *wavelets* direcionais Alpert de diferentes comprimento e largura. (b'-c') funções *bandelet* [55]

Por agregar as vantagens de representação multi-resolução das transformadas *wavelet* isotrópicas e a localização dos contornos das *wavelets* direcionais, as *bandeleets* têm sido consideradas estado-da-arte em várias aplicações de processamento de imagens. No entanto, para aplicações de compressão ainda não tem desempenho satisfatório quando comparadas com algoritmos padronizados como o JPEG 2000, devido ao alto custo computacional e a taxa adicional necessária para codificar os parâmetros do dicionário.

Os modelos direcionados apresentados, embora contornem as deficiências de representação

geométrica, não têm encontrado muito uso em aplicações de compressão devido ao seu alto custo. Por isso, para aplicações onde se deseja uma alta taxa de compressão (em detrimento de qualidade), a metodologia de representações esparsas tem sido usada. Nesse caso, as imagens são representadas por um número pequeno de coeficientes que identificam entradas de um dicionário bem adaptado a geometria e a outras características da imagem. Trata-se de um dicionário redundante, isto é, não ortogonal, que não apresenta bom desempenho quando se deseja representação perfeita, mas garante boa taxa de aproximação quando são usados poucos coeficientes.

Este tipo de codificação via dicionários relaciona-se às técnicas de quantização vetorial (QV), nas quais também parte-se do processamento por blocos da imagem original, sendo que tais blocos que são vistos como vetores, a serem representados por *codevectors* num *codebook* que deve estar disponível no codificador e no decodificador. Cada bloco é então comparado a cada *codevector* do *codebook* através da minimização de uma medida taxa-distorção e, após a definição do melhor *matching*, os parâmetros que caracterizam esse *codevector* como único dentro do *codebook*, são enviados. A QV apresenta vantagens consideráveis sobre a quantização escalar por operar em blocos de amostras, permitindo explorar dependências estatísticas lineares e não-lineares dentro dos blocos, com impacto positivo tanto na eficiência quanto na maior dimensão dos vetores [39].

Para desenvolver o melhor dicionário adaptado à geometria, devem-se adaptar as bases para características mais comuns nas imagens. Com isso, adequa-se essa representação a metodologia de representação *cartoon*-textura apresentada no capítulo 4: bases com funções suaves por partes com descontinuidades bem localizadas representam bem estruturas do tipo ‘*cartoon*’ e bases com funções oscilatórias, localmente estacionárias podem representar estruturas do tipo ‘*textura*’. Neste trabalho serão usadas *wedgelets* para a representação *cartoon* e as transformadas *wavelet* para a representação de textura.

6.5 Dicionário *Wedgelet* de 1ª Ordem

No contexto desses trabalhos sobre *wavelets* e comportamento geométrico dos contornos, surgiu o dicionário de *wedgelets* proposto por Donoho [4], com desempenho superior às próprias transformadas *wavelet* para altas taxas de compressão, em termos de análise objetiva e subjetiva (redução de *ringing*).

Um *wedgelet* é definido como uma função constante por partes em um quadrado diádico S contendo uma descontinuidade ao longo de uma linha com orientação θ e deslocamento d , como mostrado na Figura 6.13.c. Uma representação *wedgelet* de uma imagem pode ser realizada em blocos de tamanho pré-determinado (Figura 6.14.a) ou iterativamente em blocos de diferentes resoluções, organizados na forma de uma árvore *quadtrees* (Figuras 6.13.b e 6.14.b). Os nós da *quadtrees* definem a partição diádica. Nas folhas da *quadtrees* são armazenados os parâmetros do

wedgelet que aproxima o segmento de imagem no correspondente quadrado diádico. Da mesma forma que nos algoritmos de aproximações não-lineares em *quadtree wavelet* pode-se cortar a descendência da árvore em escalas mais baixas em regiões onde a imagem é suave na *quadtree wedgelet* pode-se cortar a árvore em regiões onde a imagem é bem aproximada por um *wedgelet* em termos de erro quadrático. Assim, imagens suaves por partes (*cartoon*) podem ser bem aproximadas usando um pequeno número de componentes.

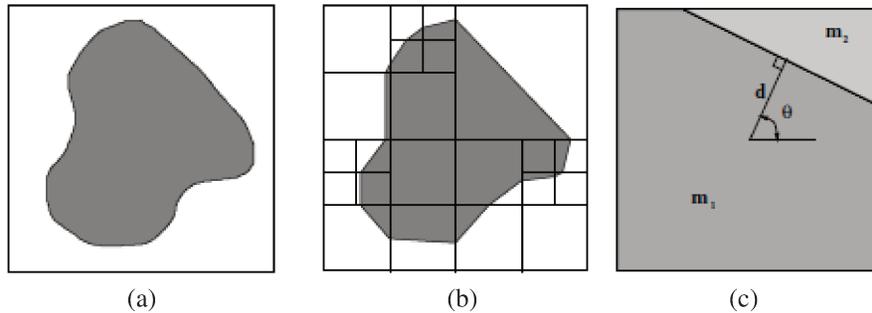


Figura 6.13 – Decomposição *wedgelet* (a) imagem artificial (b) decomposição *wedgelet* multi-escala (c) parametrização de um *wedgelet* em um quadrado $N \times N$ diádico: θ é a orientação angular do contorno, d é a distância normalizada do contorno ao centro do bloco e h é a diferença de intensidades entre cada lado do contorno.

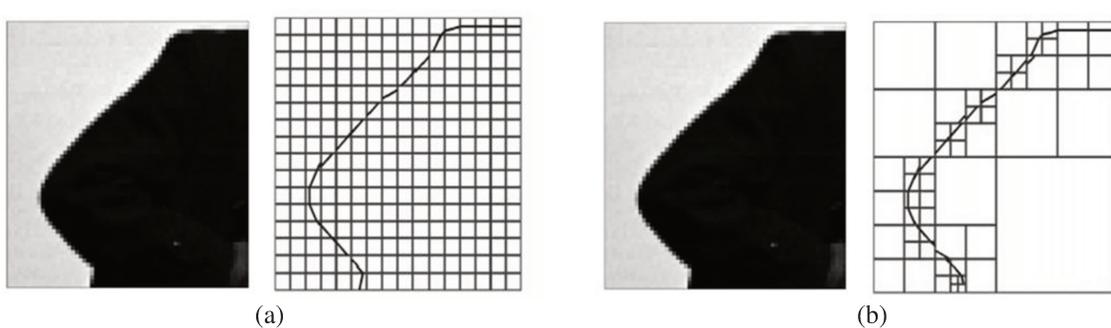


Figura 6.14 – Decomposição *wedgelet* de um trecho de imagem *cartoon* (a) escala única (tamanho dos blocos fixo) (b) multi-escala [94].

Donoho [4] introduziu o conceito de funções *horizon* para descrever um tipo de sinal hipotético (na verdade um tipo de sinal *cartoon*) que seria bem representado usando *wedgelets*. Considerando $H(x)$ uma função unidimensional chamada de *horizon* (horizonte), definida no intervalo $[0,1]$ e que a imagem é da forma:

$$f(x_1, x_2) = \mathbf{1}_{\{x_2 \geq H(x_1)\}}, \quad 0 \leq x_1, x_2 \leq 1 \quad (6.19)$$

Nesse modelo, tem-se uma imagem que é “branca” ($f = 1$) para coordenadas acima do *horizon* e “preta” ($f = 0$) para coordenadas abaixo. Deseja-se estudar casos em que a função *horizon* é regular, ou seja, segue a condição de Hölder (equação 6.9). Nesses casos, a função *horizon* é denotada por $Hor^\alpha([0,1]^2)$. A Figura 6.14.a ilustra um exemplo de função *horizon*.

Considerando apenas contornos não degenerados, isto é, que não estejam na fronteira de um

quadrado, define-se o *wedgelet* $w(x_1, x_2)$ da partição como:

$$w(x_1, x_2) = m_1 \cdot H_c + m_2 \cdot (1 - H_c) \quad (6.20)$$

onde

$$H_c = \mathbf{1}_{\{x_2 \leq b(x_1)\}}, \quad (x_1, x_2) \in S \quad (6.21)$$

O cálculo da projeção *wedgelet* normalmente é feito com um algoritmo exaustivo (*greedy*). Definido um conjunto discreto de possíveis orientações e distâncias (θ, d) , é calculado o erro quadrático de cada aproximação possível $w_{\theta,d}(x_1, x_2)$ e armazena o mínimo valor encontrado. Os valores de m_1 e m_2 para cada *wedgelet* são tomados como a média do segmento da imagem na região acima ou abaixo da linha.

$$m_1 = E[f(x_1, x_2) \cdot H_c], \quad m_2 = E[f(x_1, x_2) \cdot (1 - H_c)] \quad (6.22)$$

$$w(x_1, x_2) = \min_{\theta,d} \|f - f_w\|^2 \quad (6.23)$$

onde f denota a imagem original e f_w é a aproximação *wedgelet*.

Muitos outros métodos poderiam ser usados para a obtenção da melhor projeção *wedgelet* de um bloco de imagem $N \times N$ (que minimize o MSE). A busca exaustiva no espaço \mathbb{R}^4 para todos os parâmetros θ, d, m_1 e m_2 é o método mais pesado computacionalmente. Estimativas do melhor *wedgelet* também podem ser obtidas pela análise da transformada de Radon [95] ou por uma transformada *wavelet* complexa [96]. Técnicas rápidas podem ser desenvolvidas usando dicionários discretos e com cálculos parciais específicos para a geometria considerada. Em [97][98] são apontadas as sugestões mais recentes para otimizar a projeção.

O dicionário *wedgelet* é definido como o conjunto

$$W = \{w_{i,j,m} : i = 0, \dots, \log_2 N, j = 0, \dots, 4^i - 1, m = 0, \dots, M_w(S_{i,j}) - 1\} \quad (6.24)$$

A representação *wedgelet* é definida pela seguinte fórmula:

$$f_w(x_1, x_2) = \sum_{i,j,m} w_{i,j,m}(x_1, x_2) \quad (6.25)$$

Como W é um dicionário, não uma base, alguns coeficientes $w_{i,j,m}$ da representação acima podem ser nulos. A decomposição é feita basicamente em dois passos. No primeiro passo é feita uma decomposição completa da imagem, isto é, são encontradas as melhores representações *wedgelet* para todos os níveis de resolução em cada quadrado $s_{i,j}, 0 \leq j \leq 4^i, 0 \leq i \leq \log_2 N$ de acordo com o erro quadrático. Todos os coeficientes *wedgelet* são armazenados como nós da *quadtrees*. Em seguida, no segundo passo é aplicado um algoritmo de otimização, removendo nós

em níveis avançados da árvore, de forma a obter um número mínimo de nós na representação, mas garantindo a qualidade desejada. O algoritmo busca minimizar a seguinte função de custo [45]:

$$R = \min_P \left\{ \|f - f_w\|^2 + \lambda \cdot |P| \cdot K \right\} \quad (6.26)$$

onde P denota a partição da imagem (nós armazenados na *quadtree* durante o primeiro passo), K é o número de *bits* necessário para codificar os nós e λ é o parâmetro de taxa-distorção, também chamado de fator de penalização. Para reconstrução exata, $\lambda = 0$ e a aproximação *wedgelet* é exatamente a imagem original. Usualmente, a minimização é feita por um algoritmo de corte de baixo para cima na *quadtree* (*bottom-up*), cortando da árvore os nós com um custo maior do que o custo do nó pai.

Além disso, pode-se explorar a regularidade dos contornos na imagem, incorporando modelos de contornos multiescala ao algoritmo de corte da *quadtree*, favorecendo *wedgelets* com orientações alinhadas entre os blocos diádicos, ou com pequenos desvios sobre as previsões de níveis mais altos [85].

Para a análise do erro produzido por aproximações *wedgelets*, define-se o conjunto \mathfrak{M} de todas as possíveis partições com um seguimento de contorno (*wedge*) em um quadrado unitário. Para uma dada partição *wedgelet* $\mathcal{W} \in \mathfrak{M}$, seja $V(\mathcal{W})$ o espaço linear de funções lineares por partes, relativo a \mathcal{W} :

$$V(\mathcal{W}) := \left\{ f \mid f(x) = \sum_{W \in \mathcal{W}} P_W(x) \chi_W(x), \text{ para alguma } P_W \in \Pi_1 \right\} \quad (6.27)$$

onde Π_1 denota o conjunto de polinômios de duas variáveis de grau no máximo 1.

Considerando um sinal $f \in L^2[0,1]^2$ e denotando $\hat{f}(\mathcal{W})$ a projeção ortogonal de f no espaço $V(\mathcal{W})$, $\hat{f}(\mathcal{W})$ é a única função que minimiza a o erro quadrático L^2 :

$$\|f - \hat{f}(\mathcal{W})\|_{L^2[0,1]^2} = \inf_{g \in V(\mathcal{W})} \|f - g\|_{L^2[0,1]^2} \quad (6.28)$$

A partir disso, pode-se definir uma aproximação não-linear de n termos como nos casos anteriores escolhendo uma partição \mathcal{W} tal que $|\mathcal{W}| = n$:

$$\|f - \hat{f}(\widehat{\mathcal{W}}_n)\|_{L^2[0,1]^2} = \inf_{\mathcal{W} \in \mathfrak{M}_n} \|f - \hat{f}(\mathcal{W})\|_{L^2[0,1]^2} \quad (6.29)$$

onde $\mathfrak{M}_n := \{\mathcal{W} \in \mathfrak{M} \mid |\mathcal{W}| \leq n\}$

Para determinar a eficiência dessa aproximação, define-se o conjunto W_α :

$$W_\alpha = \left\{ f \in L^2[0,1]^2 \mid |f|_{W,\alpha} < \infty \right\} \quad (6.30)$$

$$|f|_{W,\alpha} := \sup_{n \in \mathbb{N}} \left(n^{\frac{\alpha}{2}} \|f - \hat{f}(\mathcal{W}_n)\|_{L^2[0,1]^2} \right) \quad (6.31)$$

O conjunto W_α é chamado classe de aproximação de ordem α e atua de forma similar a norma

do espaço de Sobolev (eq. 6.5). Um resultado interessante é quando o sinal possui descontinuidades e $1 \leq \alpha \leq 2$, onde a aproximação por *wedgelets* tem uma taxa de decaimento mais rápida do que a aproximação não-linear por coeficientes *wavelet*, como apontado pela proposição a seguir: [4]

Seja $\alpha \in [1,2]$, $f \in \text{Hor}^\alpha [0,1]^2$ e h a função que define o contorno de f . Existem constantes $C, C' > 0$ independentes de n e para cada n existe uma partição *wedgelet* \mathcal{W}_n sobre o quadrado unitário $[0,1]^2$, tal que $|\mathcal{W}_n| \leq C'n$ e satisfaz:

$$\|f - \hat{f}_n(\mathcal{W}_n)\|_{L^2[0,1]^2} \leq \frac{C}{n^{\frac{\alpha}{2}}} \quad (6.32)$$

onde $\hat{f}_n := \hat{f}_n(\mathcal{W}_n)$ é a projeção L^2 de f no espaço de funções *wedgelets* sobre a partição \mathcal{W}_n .

Para imagens *horizon*, a aproximação não-linear por *wavelets* tem uma taxa de decaimento de erro mais lenta:

$$\|f - \hat{f}_n\|_{L^2[0,1]^2} = O\left(\frac{C}{n^{\frac{1}{2}}}\right) \quad (6.33)$$

A Figura 6.15 mostra uma comparação simplificada da aproximação de uma imagem tipo *horizon* por árvores *quadtree* compostas por nós quadrados constantes (a) e por nós *wedgelet* (b). Para uma reconstrução com qualidade comparável a aproximação *wedgelet* requer aproximadamente metade do número de componentes em relação à aproximação por *quadtrees* constantes.

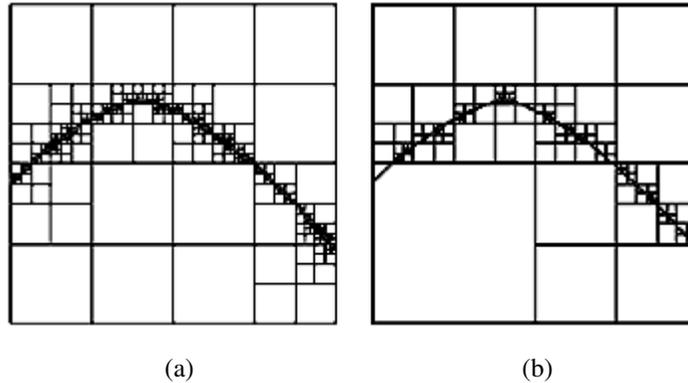


Figura 6.15 – Aproximações em *quadtree* (a) *quadtree* com 967 nós constantes, PSNR: 33.69 dB. (b) *quadtree* com 248 nós *wedgelet* (496 coeficientes *wedgelet*), PSNR: 33.71 dB [99]

Existem diferentes formas de armazenar os coeficientes *wedgelet* de cada nó da estrutura *quadtree*. A forma mais simples consiste em usar o mesmo número de *bits* em cada nó e armazenar todos os parâmetros (d, θ, m_1, m_2) de cada nó. Uma forma mais sofisticada é armazenar apenas as diferenças de parâmetros em relação a parâmetros estimados pelos níveis superiores da árvore. Para imagens com geometria regular, pode-se estimar o ângulo do *wedgelet* em um nível de resolução mais detalhado a partir do nó pai, com um pequeno desvio máximo permitido, codificando apenas

as diferenças (Figura 6.16). O conjunto de diferença nos coeficientes $(\Delta d, \Delta\theta, \Delta m_1, \Delta m_2)$ é codificado por meio de um dos métodos em estado-da-arte de codificação de entropia, de acordo com as probabilidades de ocorrência.

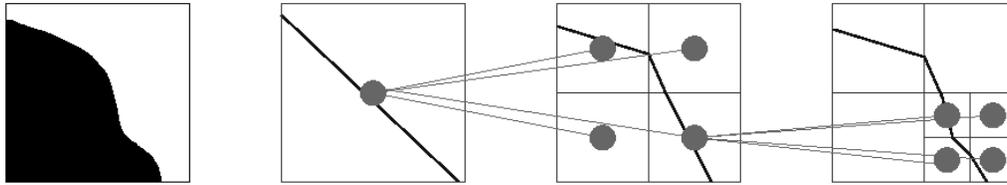


Figura 6.16 – Estimativa do parâmetro angular do *wedgelet* a partir de níveis superiores [85]

6.6 Extensões ao Modelo *Wedgelets* – *Surflets* e *Platelets*

Na seção 6.5, foi considerada em cada nó uma aproximação por regiões divididas por uma reta com um parâmetro angular que a define. Como mostrado por Lisowska [100], esse é apenas um caso especial de função suave por partes para a aproximação. É possível utilizar para a separação entre regiões qualquer função unidimensional contínua, por exemplo, polinômios. O requisito principal recai sobre a forma de codificar os parâmetros da função de aproximação e definir como se dá a relação de estimativa entre cada nível de resolução. Em geral, quanto maior a ordem da função de aproximação, maior o número de parâmetros para representá-la. Além disso, o tempo computacional para determinar a melhor projeção também aumenta.

Como apontado em [101], o aumento de um parâmetro a representação *wedgelet* permite melhor aproximação *wedgelet* com um número comparável de *bits*. Foram definidas *wedgelets* de 2ª ordem, ou *arcllets* [102], que possuem um parâmetro adicional para indicar linhas curvas na fronteira entre as regiões *wedgelet*. A parametrização das *arcllets* é feita por funções cônicas, como parábolas, elipses e hipérbolas, alinhadas segundo um ângulo e com apenas um parâmetro a mais que indica a abertura da curva (Figura 6.17.a).

Aproximações por regiões suaves, mas não constantes, são propostas por [103]. São definidas *wedgelets* contendo regiões com valores linearmente variáveis, divididas por uma linha reta, chamada de *platelet* (Figura 6.17.b).

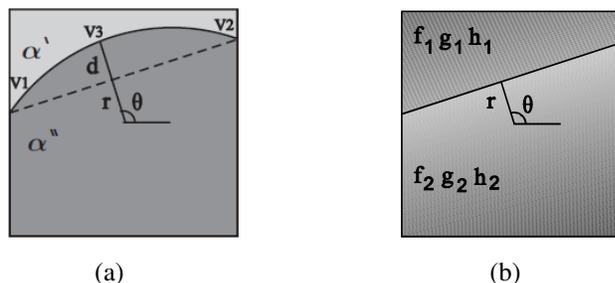


Figura 6.17 – Extensões das *wedgelets* (a) *arcllet* [101][102] (b) *platelet* [103]

Em [104] o conceito de *wedgelets*, *arcllets* e *platelets* é generalizado pelo dicionário de *surflets*. Consideram-se funções $f(M, K_d, K_s)$ M -dimensionais, constantes por partes ou suaves por partes, que contenham uma descontinuidade C^{K_d} -suave $(M-1)$ -dimensional, separando duas regiões C^{K_s} -suaves em M dimensões (Figura 6.18). É demonstrado que um algoritmo de projeção *surflet* consegue uma curva de taxa-distorção ainda mais abrupta. Tem-se:

$$\|f - \hat{f}\|^2 \lesssim \left(\frac{\log R}{R}\right)^{\min\left(\frac{K_d}{M-1}, \frac{2K_s}{M}\right)} \quad (6.32)$$

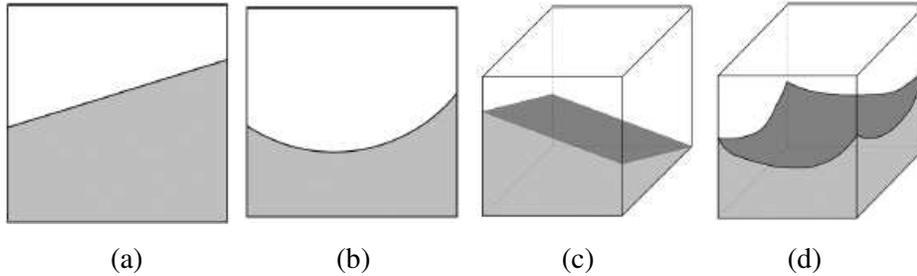


Figura 6.18 – Exemplo de *surflets* (a) $M = 2, K_d = [1,2]$ (b) $M = 2, K_d = [2,3]$ (c) $M = 3, K_d = [1,2]$ (d) $M = 3, K_d = [2,3]$ [104]

6.7 Modelos Híbridos – Decomposição *Cartoon-Textura*

Os modelos *wedgelets* apresentados garantem uma reconstrução da estrutura geométrica (contornos) com taxa de aproximação superior às transformadas *wavelets* e outros modelos ortogonais e têm reconstrução perfeita para sinais do tipo *horizon (cartoon)*, desde que o modelo seja bem parametrizado a priori (daí, a grande variedade de *wedgelets* disponível, além da parametrização dos ângulos e do processo de decomposição em árvores). Para imagens naturais no entanto, modelos *horizon* não representam com perfeição, embora a informação primordial de contornos seja preservada. Para garantir a reconstrução perfeita é preciso completar o modelo com informação não geométrica, de acordo com a metodologia *cartoon-textura* (capítulo 4).

Alguns modelos híbridos têm sido usados na literatura, geralmente com mais sucesso do que os modelos geométricos únicos. A metodologia consiste em determinar um método que funcione muito bem para codificar a estrutura geométrica dos contornos, como os métodos apresentados nas seções 6.4 a 6.6 e idealmente toda a informação não-codificada (resíduo) terá uma estrutura do tipo textura e pode ser codificada por meio das aproximações *wavelets* não-lineares com taxa de erro $(N^{-2\alpha})$.

Wakin et al. [46] propuseram um codificador de dois passos que utiliza aproximações *wedgelets* preditivas e quantização dos parâmetros dos nós para identificar e o resíduo (diferença entre imagem aproximada e a imagem original) é codificado usando *wavelets*. Para eliminar efeitos de bloqueio indesejados produzidos pela decomposição *wedgelet* com resolução limitada, a componente *cartoon* passa por um estágio de filtragem passa-baixas para eliminar bordas. Ao final

do processo de decodificação as duas componentes são somadas para compor novamente a imagem, como mostrado na Figura 6.19.

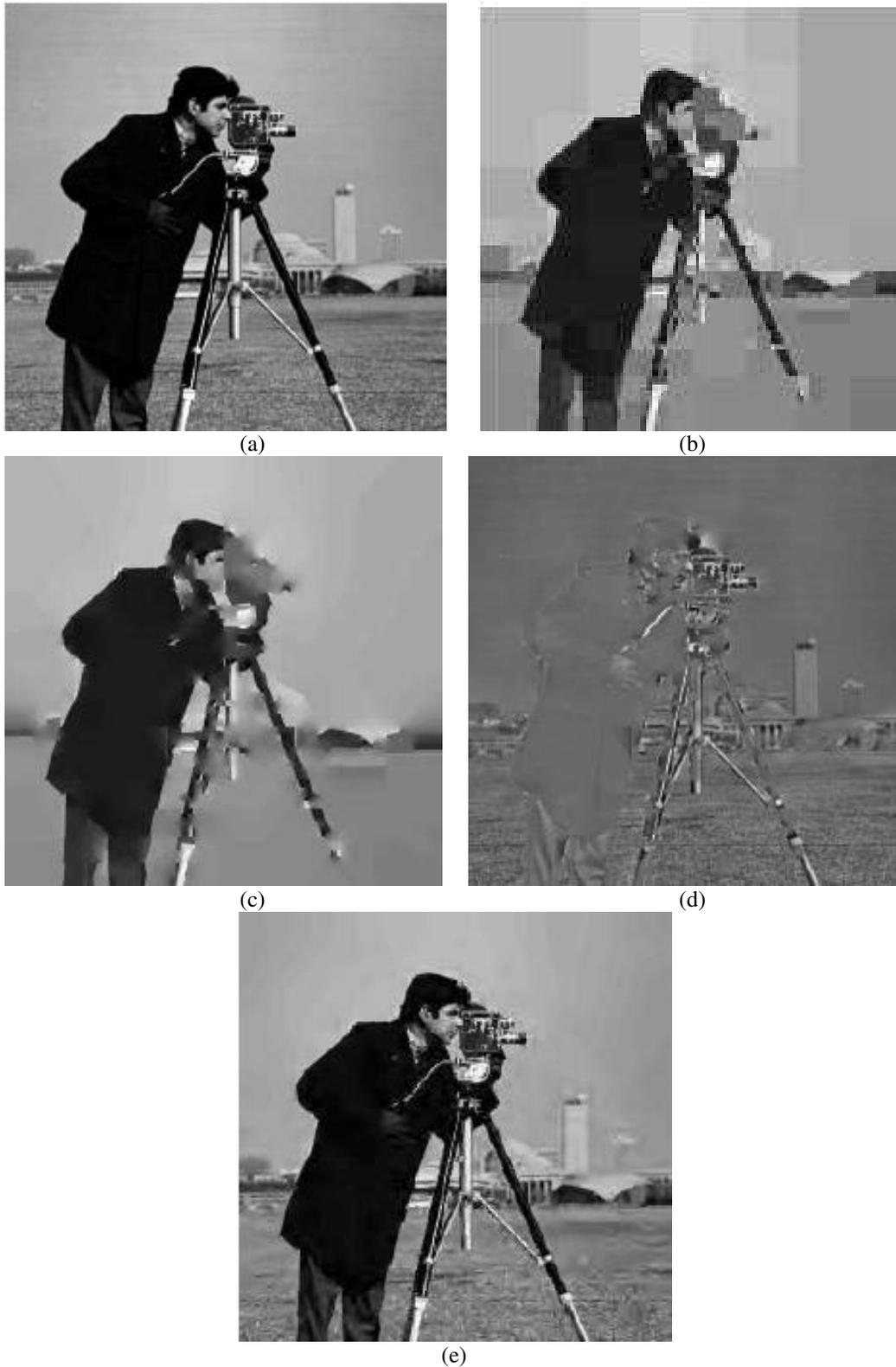


Figura 6.19 – Codificação *wedgelet-wavelet* de [46] (a) original (b) aproximação *wedgelets* (*cartoon*) (c) *cartoon* após filtragem (d) resíduo codificado por *wavelets* (*textura*) (e) imagem reconstruída (0,4 bpp, PSNR = 27,35 dB).

O algoritmo proposto em [46] não atinge resultados superiores a algoritmos mais recentes usando *wavelets*, como SPIHT e EZW, em termos de PSNR, apesar de reduzir efeitos de *ringing*. O baixo desempenho se explica pela perda de precisão das bordas com a filtragem passa-baixas da componente *cartoon* e o surgimento de bordas artificiais na componente textura, por causa de uma má aproximação da componente *cartoon*. Esses resultados mostraram, inclusive para os próprios autores do método, que o resíduo da aproximação *wedgelet* precisa ser devidamente tratado antes de ser aproximado por *wavelets*, ou então o resultado pode ser ainda pior que um método usando somente *wavelets*.

Em [105] Wakin et al. propõem um método de codificação híbrida *wedgelets-wavelet* que otimiza a relação taxa-distorção para cada subregião da imagem, ao invés de usar o resíduo. Basicamente o método é uma modificação do algoritmo SFQ (*Space Frequency Quantization*), que codifica os coeficientes *wavelet* buscando minimizar o erro da aproximação não-linear a cada passo e a cada resolução observada. Trata-se de um algoritmo mais complexo que o SPIHT pois pode fazer várias varreduras em coeficientes *wavelet* em um mesmo passo, até encontrar a melhor escolha, e por isso obtém melhores resultados.

Na modificação proposta em [105], chamada de W-FSQ, os coeficientes *wavelet* são comparados um a um a fim de determinar a função de custo taxa×distorção (número de *bits* necessários para codificá-lo e erro produzido ao não codificar). O coeficiente pode ser classificado como não-significativo (região suave, codifica com um coeficiente nulo na decomposição *wavelet*), textura (região oscilatória, o coeficiente é quantizado e codificado e a árvore de descendência entra na codificação) ou contorno. Nesse último caso, são codificados os parâmetros *wedgelet* (d, θ, m_1, m_2) que representam o contorno naquela região. Para a decodificação, a região é preenchida no domínio *wavelet* com os coeficientes correspondentes a geometria, ou mais especificamente, ao *wedgelet*. Isso é feito usando *wedgeprints*, projeções dos *wedgelets* no domínio *wavelet* (Figura 6.20).

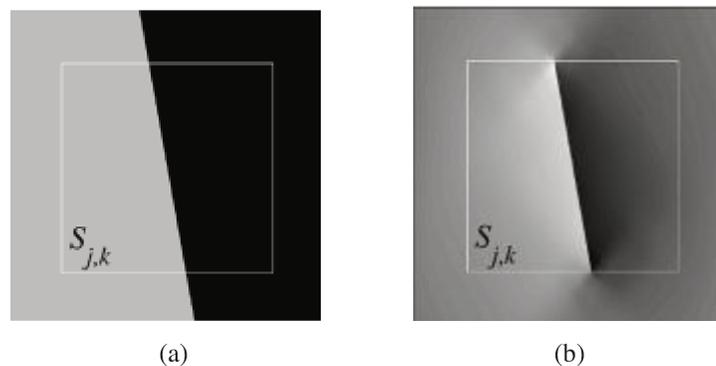


Figura 6.20 – *Wedgelets* no domínio *wavelet* (a) porção de uma imagem contendo um *wedgelet* em um bloco $S_{j,k}$ (b) *wedgeprint* obtido pela projeção do *wedgelet* na subbanda vertical do domínio *wavelet*.

Para gerar os *wedgeprints*, é utilizada uma amostra do *wedgelet* com tamanho maior que o suporte da base *wavelet* escolhida. Caso as bordas não sejam tratadas de forma adequada para a decomposição, com algum tipo de extensão periódica ou espelhamento, a reconstrução do *wedgelet* a partir do *wedgeprint* pode gerar efeitos de blocagem intensos.

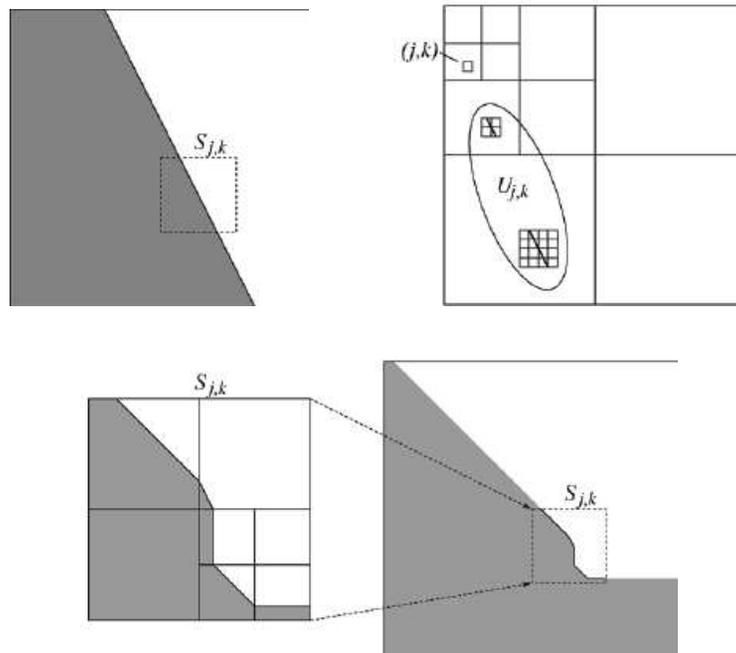


Figura 6.21 – Obtenção de um símbolo *wedgeprint*

A grande vantagem do algoritmo W-SFQ sobre as abordagens anteriores é que os coeficientes da componente *cartoon* estão embutidos na mesma árvore de decomposição *wavelet* da componente textura. Cada coeficiente dessa árvore, bem como as sub-árvores por eles geradas, passa a indicar uma região da imagem que pode ser *cartoon* ou textura, dependendo da relação taxa \times distorção pretendida. A Figura 6.22 mostra uma comparação da árvore de decomposição do SFQ e do W-SFQ. Como as *wavelets* têm suporte compacto, a região *cartoon* ou textura é bem definida, na imagem irão existir imagens dos dois tipos (Figura 6.23). Apesar disso, deve-se levar em conta a influência dos *wedgeprints* sobre os coeficientes e árvores vizinhas, de acordo com o comprimento da base *wavelet* usada. Essa influência pode ser difícil de ajustar e invalidar o uso do W-FSQ.

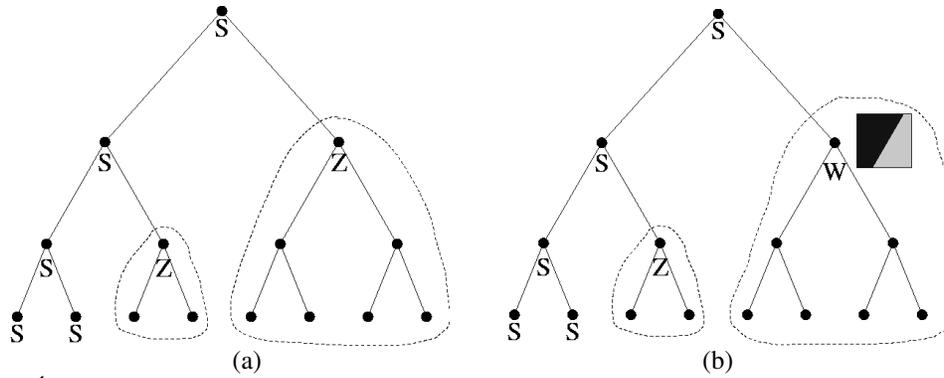


Figura 6.22 – Árvore de decomposição: S – textura significativa (quantização escalar com coeficientes *wavelet*), Z: não significativo (árvore de zeros), W: contorno significativo (árvore *wedgelet*) (a) SFQ [106] (b) W-SFQ [107].

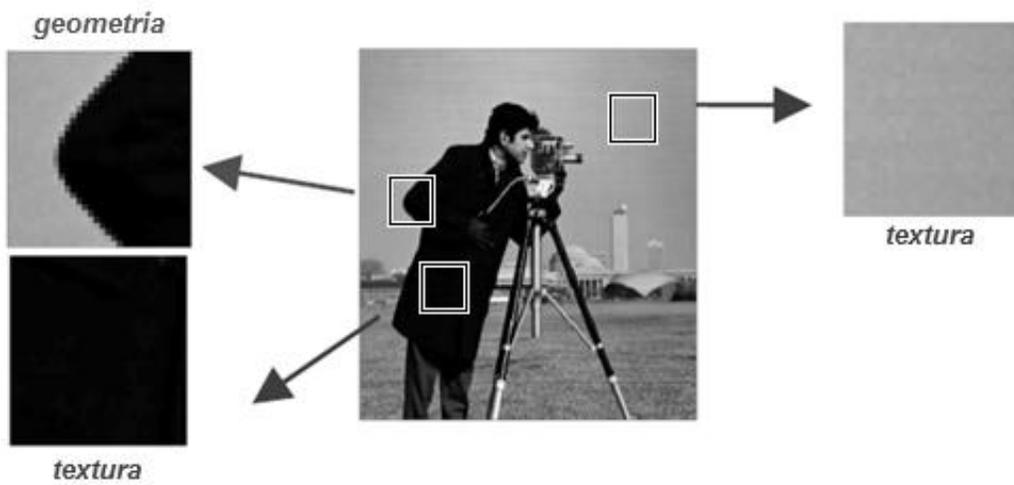


Figura 6.23 – Imagem ‘cameraman’ e diferentes tipos de regiões que ela contém.

Da mesma forma que as *wedgelets* foram estendidas para as *surflets*, as *wedgeprints* são estendidas para as *surfprints*, projeções das *surflets* no domínio *wavelet* (Figura 6.24).

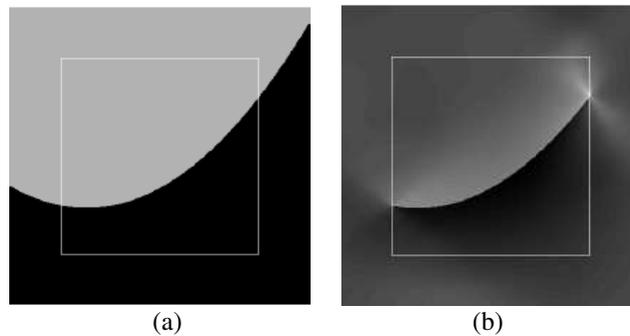


Figura 6.24 – (a) Porção de uma imagem contendo um *surflet* em um bloco $S_{j,k}$ (b) *surfprint* obtido pela projeção do *surflet* na subbanda vertical do domínio *wavelet*.

O paradigma de combinar transformadas para representar de forma mais eficiente, diferentes conteúdos de um sinal foi preconizada por Coifman no início dos anos 90 [108]. Como apontado por Baraniuk [94], o modelo de compressão *cartoon-textura* multiresolução pode ser ainda estendido para dicionários maiores (e mais redundantes). Para blocos de textura com estatística

regular podem ser adotadas bases estacionárias, como DCT. Para contornos, diferentes dicionários derivados da estrutura *wedgelet* podem ser usados (Figura 6.25).

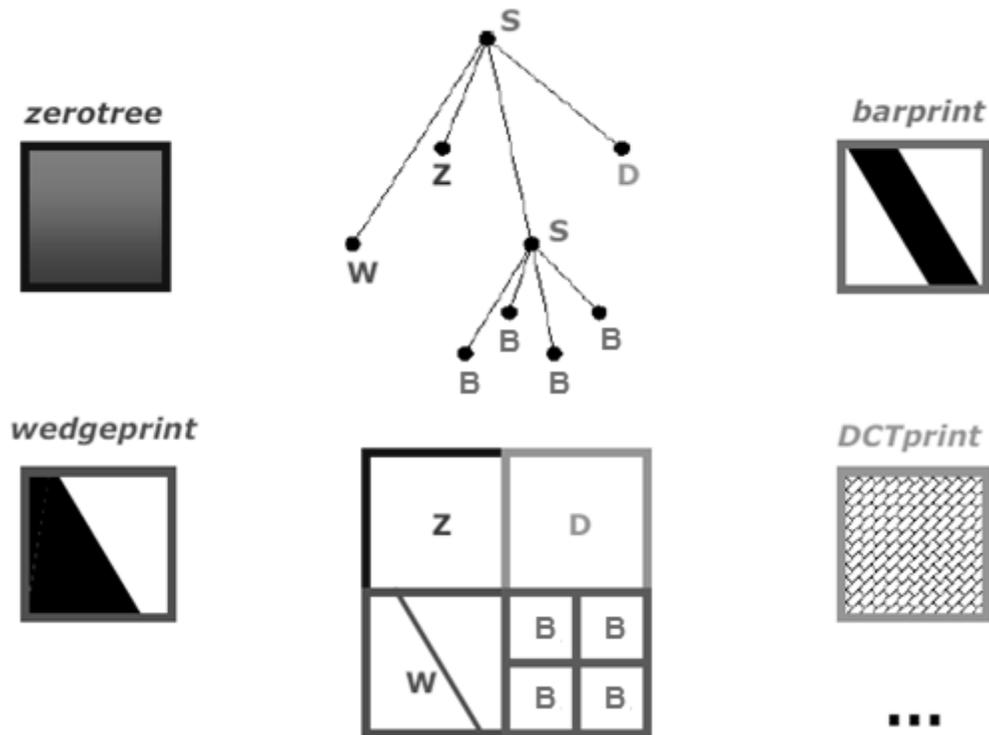


Figura 6.25 – Extensões ao modelo de multi-dicionários *cartoon-textura* [94].

O uso de dicionários redundantes para a codificação é defendida por Mallat [55], pois dicionários pequenos como bases ortogonais têm uma capacidade limitada de representação esparsa, que é necessária quando se trabalha com altas taxas de compressão. Dicionários maiores, que incorporem mais padrões, podem aumentar a esparsidade e assim inovar em aplicações de compressão, remoção de ruído, problemas inversos e reconhecimento de objetos. Encontrar um dicionário de tamanho determinado que aproxime um sinal com erro mínimo é um problema NP-complexo (de complexidade não-polinomial) em dicionários redundantes. Assim, é necessário recair sobre aproximações “boas”, mas não matematicamente ótimas, obtidas com algoritmos específicos.

Neste contexto as *wedgelets* se apresentam como um paradigma inovador para a representação esparsa da geometria das imagens, ou mais especificamente, para a estrutura *cartoon*. Com o uso híbrido combinado com outros dicionários e uma estrutura em árvore multi-resolução, melhores métodos de codificação ainda podem ser implementados no futuro.

Capítulo 7

Proposta e Resultados

Neste capítulo serão apresentados os resultados experimentais obtidos com o método proposto na presente dissertação combinando a decomposição *cartoon*-textura com um dicionário *wedgelet* e a transformada *wavelet* ortogonal. São encontrados os melhores parâmetros de cada método de codificação (seção 7.3) e é proposto um novo método de codificação híbrido (seção 7.4). Na seção 7.5 são feitas comparações adicionais com métodos *wavelet* e *wedgelet* isolados e também métodos propostos por outros autores.

Será apresentada a verificação de que o método proposto consegue combinar as vantagens dos dois métodos e atingir resultados satisfatórios para altas e baixas taxas, se aproximando de padrões com sistemas completos de codificação como JPEG 2000 e superando o JPEG. Análises subjetivas mostram uma redução desejada no efeito de *ringing*.

Os resultados são mostrados para imagens naturais e imagens do tipo desenho, nas quais se consegue um ganho ainda superior frente a outros algoritmos de codificação.

7.1 Método Proposto (WWB – *Wedgelet-Wavelet* por Blocos)

O método proposto para a codificação *cartoon*-textura (que será referenciado na presente dissertação por WWB – *Wedgelet-Wavelet* por Blocos) é mostrado na Figura 7.1. As etapas coloridas indicam as modificações inéditas propostas. Primeiramente, a imagem é dividida em blocos de tamanho 64×64 e cada um desses blocos é classificado como *cartoon* (a ser codificado com *wedgelets*) ou *textura* (a ser codificado com *wavelets*).

Embora a decomposição *cartoon*-textura já tenha sido usada por outros (seção 6.7), neste trabalho propõe-se uma nova abordagem, fazendo uma classificação em cada bloco. A escolha é feita com base na função de custo taxa-distorção (eq. 6.26). O bloco é pré-codificado com símbolos *wavelet* e *wedgelet* e o dicionário que determinar a menor função de custo é escolhido. Essa estratégia de procura exaustiva (*bottom-up*) é o que torna o algoritmo mais complexo, mas também bastante adaptável para as diferentes regiões da imagem. Assim, blocos em que a geometria é mais importante (*cartoon*) tendem a ser escolhidos por *wedgelets*. Após todos os blocos serem lidos, os coeficientes *wavelet* dos blocos *textura* são codificados por uma versão modificada do SPIHT, que

otimiza a função taxa-distorção ao invés de manter uma taxa de *bits* pré-fixada. Os *bits* gerados pelo SPIHT modificado são então adicionados ao *bitstream* correspondentes aos blocos.

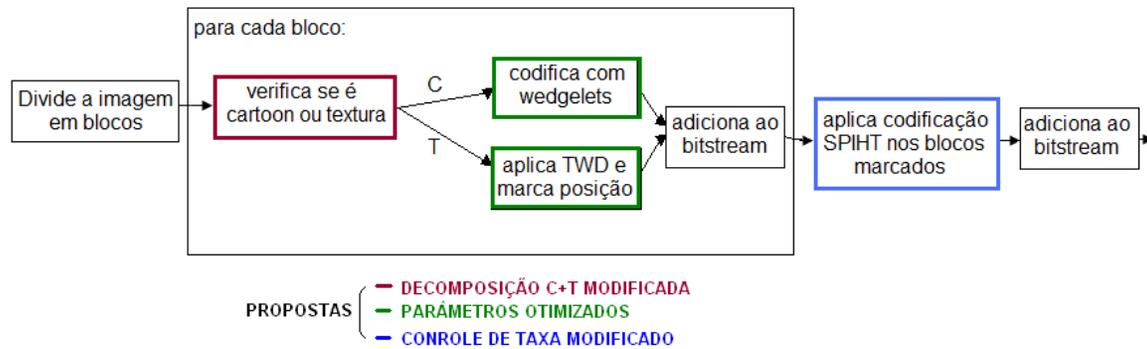


Figura 7.1 – Método proposto (WWB).

Para o dicionário *wedgelet*, o processo é dividido em 3 etapas (Figura 7.2): geração da árvore *wedgelet* (uma vez para cada imagem e para cada conjunto de parâmetros de dicionário), codificação e decodificação (ambos uma vez para cada taxa escolhida).

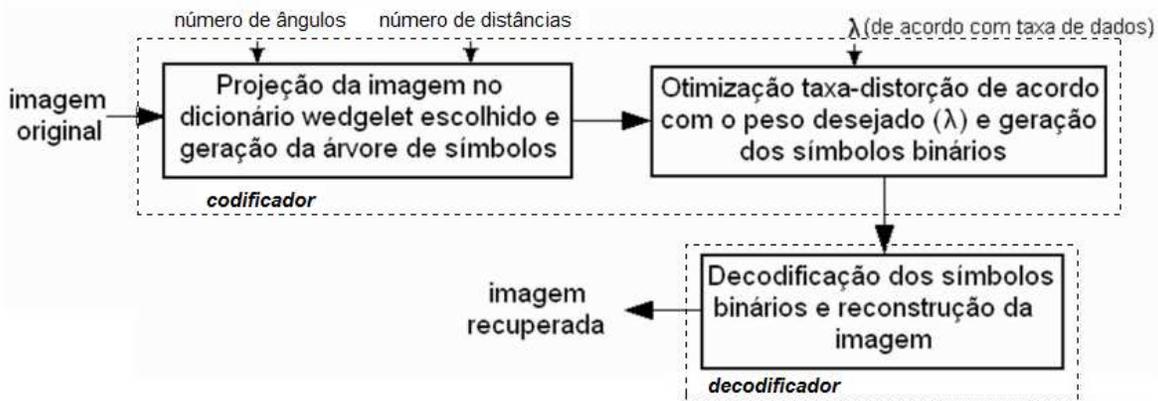


Figura 7.2 – Processo de codificação-decodificação *wedgelet*.

Busca-se a configuração de parâmetros (especificamente, número de possíveis *wedges* no dicionário) que otimiza a codificação em termos de minimizar o erro objetivo, trabalhada como uma das modificações propostas neste trabalho. O número de *wedges* possíveis depende do número de ângulos θ (são consideradas partições uniformes de 0° a 180°) e do número de distâncias d (consideradas partições uniformes de um domínio quadrado). Embora o aumento no número de possibilidades melhore a representação de uma imagem natural, tende a aumentar também o número de bits para cada elemento do dicionário. Para a codificação da imagem, os símbolos do dicionário são transformados em símbolos binários pelo método de Huffman.

Além da otimização do dicionário *wedgelet*, também é necessário determinar os melhores parâmetros para a codificação *wavelet*, como será mostrado na seção 7.3.3. Na seção 7.4 são

apresentados os resultados para o codificador proposto WWB.

7.2 Simulações

Os algoritmos foram implementados em ambiente de programação Matlab 7.5.0 (R2007b) em um computador PC Intel Core 2-Quad 2.50 GHz, 3.24 GB de RAM Windows XP, isolado de outros processos externos. Os resultados obtidos são analisados com base no tempo de processamento, na qualidade visual das imagens recuperadas (análise subjetiva), na medida de PSNR (*Peak Signal to Noise Ratio*), no número de componentes utilizadas para a aproximação e no número de *bits per pixel* (bitrate) utilizado na codificação. Para a medida de tempo foi usada a função *cputime* do Matlab.

No total dos experimentos foram utilizados 2 conjuntos de 8 imagens cada, com 256x256 *pixels* em escala de cinza: imagens naturais da base de dados Waterloo Bragzone [111] e desenhos extraídos de outros artigos de processamento de imagens devidamente referenciados no texto. Foram escolhidas imagens com características diversas em cada conjunto. Nas Figuras 7.3 e 7.4 são mostradas as imagens originais utilizadas.

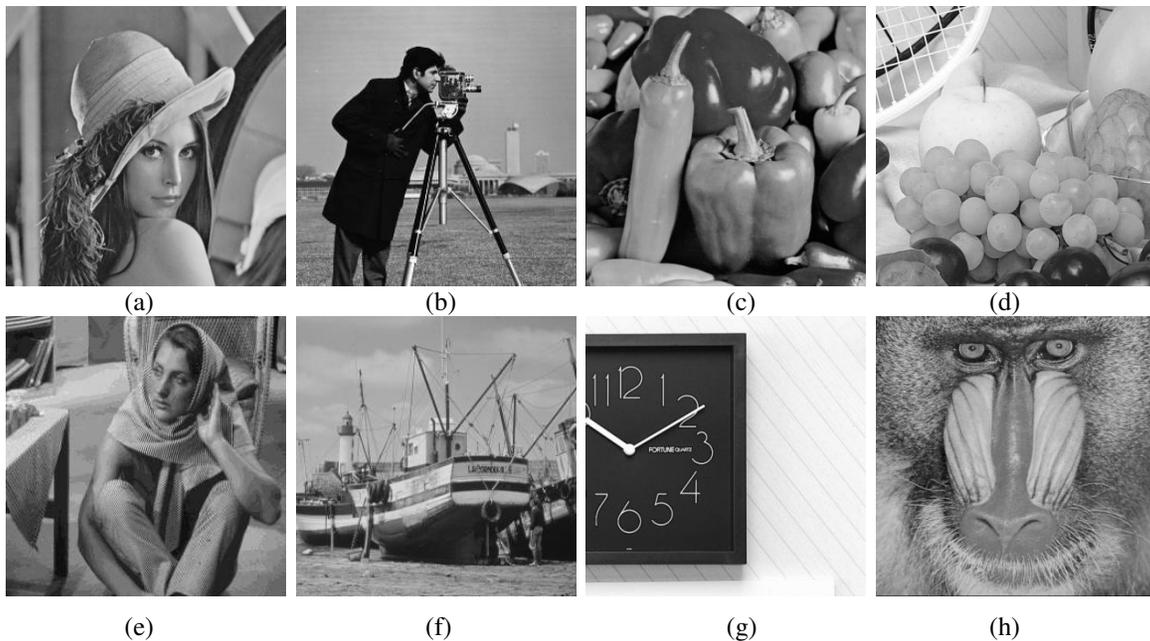


Figura 7.3 – Imagens naturais da base de dados Waterloo [111] (a) Lena (b) cameraman (c) peppers (d) fruits (e) Barbara (f) boat (g) clock (h) baboon.

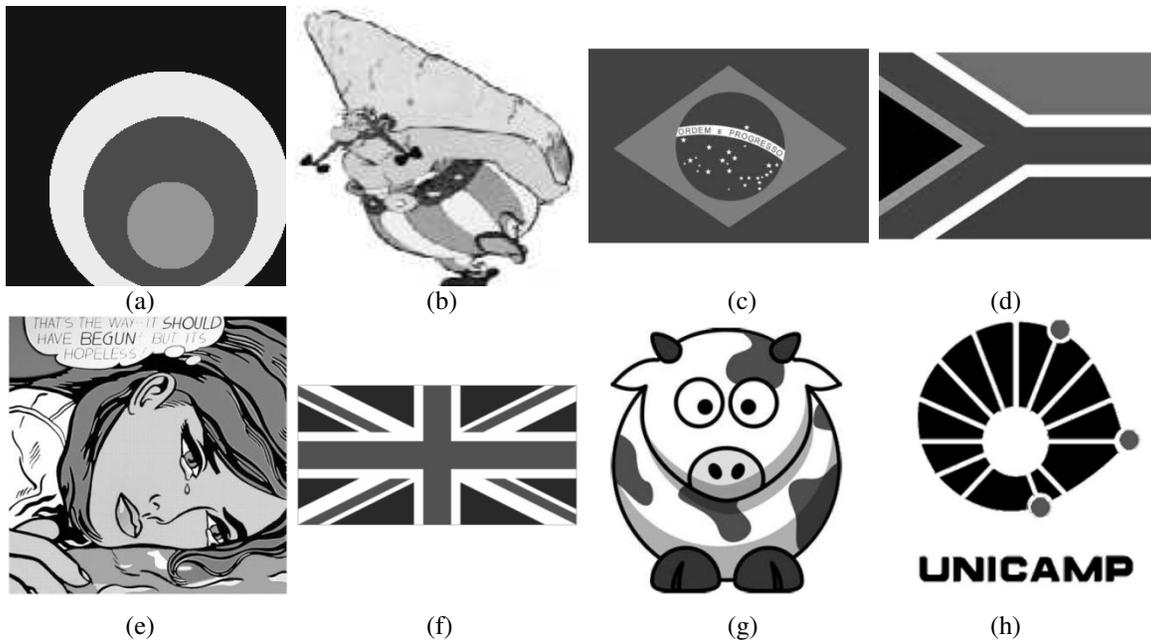


Figura 7.4 – Desenhos usados nos testes (a) *circles* (b) *obelix* (c) Brasil (d) África (e) *roy* (f) uk (g) *cow* (h) Unicamp.

O método WWB é comparado com a codificação isolada por *wavelets* e por *wedgelets*, destacando a análise subjetiva, dado o objetivo de redução de artefatos visuais. Comparações adicionais com outros métodos baseados em *wedgelets* foram feitas diretamente do material apresentado nos artigos de referência. Para a comparação com o método constante em [109] foi usado um *software* desenvolvido pelos autores e disponível gratuitamente em [110], mas sem comparação de tempo de processamento, pois foi implementado em ambiente distinto ao Matlab.

Para a comparação com o padrão JPEG foi utilizado o codificador presente no Matlab com a função *imwrite*, variando o parâmetro de qualidade de acordo com a taxa de *bits* desejada. Para a comparação com o padrão JPEG 2000 foi usada a versão aberta OpenJPEG [114].

7.3 Escolha dos Melhores Parâmetros

Nesta seção serão determinados os melhores parâmetros para a codificação com os dicionários *wedgelet* e *wavelet* isolados aplicados na imagem toda, dividida em blocos.

7.3.1 Determinação do Número de Ângulos

O parâmetro angular na determinação do *wedge* é discretizado e o número de ângulos possíveis deve ser escolhido a priori no sistema de codificação. A Figura 7.5 mostra valores de resolução angular utilizados.

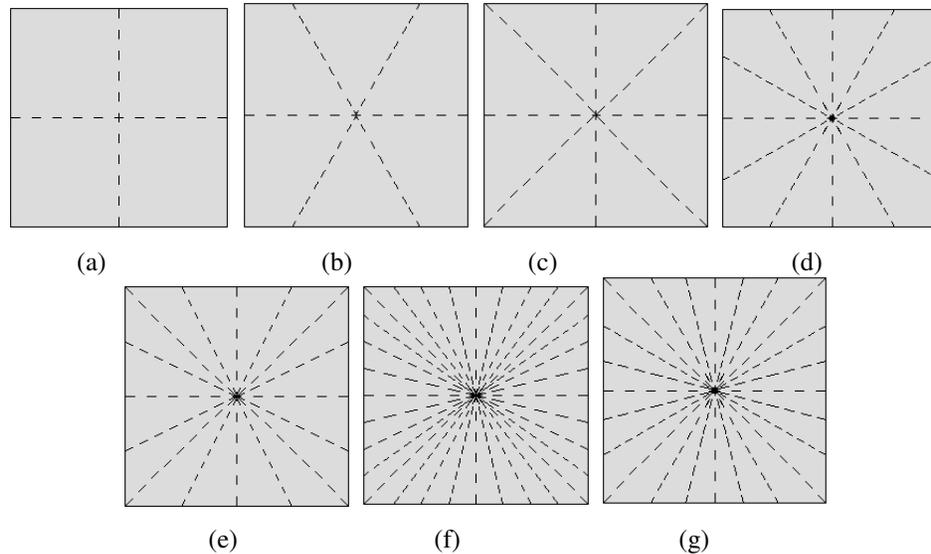


Figura 7.5 – Variação do número de ângulos testados na determinação do *wedge* (a) 2, (b) 3, (c) 4, (d) 6, (e) 8, (f) 16, (g) 18.

Na Figura 7.6 são mostradas curvas de desempenho da codificação para diferentes resoluções angulares possíveis. Resoluções angulares intermediárias a esses valores não teriam um ganho significativo devido ao domínio discretizado da imagem, que faz com que variações angulares pequenas se confundam. Para estabelecer essas curvas, o número de distâncias d foi fixado em 4. Observa-se que, embora o aumento do número de ângulos melhore a codificação, para valores maiores do que 8 ângulos a curva de PSNR tende a baixar, fazendo desse, um valor ótimo dentre os analisados, tanto para imagens naturais quanto para imagens tipo “desenho”.

É possível observar também, por meio das Tabelas 7.1 a 7.2 que o maior peso computacional, em ambos os casos, encontra-se na etapa de geração da árvore, que ocorre somente uma vez no lado do codificador. Isso torna o método mais indicado para aplicações de *broadcast*, em que o codificador pode ser muito mais complexo do que o decodificador. É possível verificar também que o tempo de processamento varia diretamente com o número de ângulos e distâncias usado.

Os valores médios de PSNR obtidos para imagens tipo “desenho” são naturalmente mais altos uma vez que as técnicas envolvidas foram originalmente desenvolvidas dedicadas para tratar imagens desse tipo. Essa vantagem também se encontra refletida nas Tabelas 7.1 e 7.2 nos tempos de processamento.

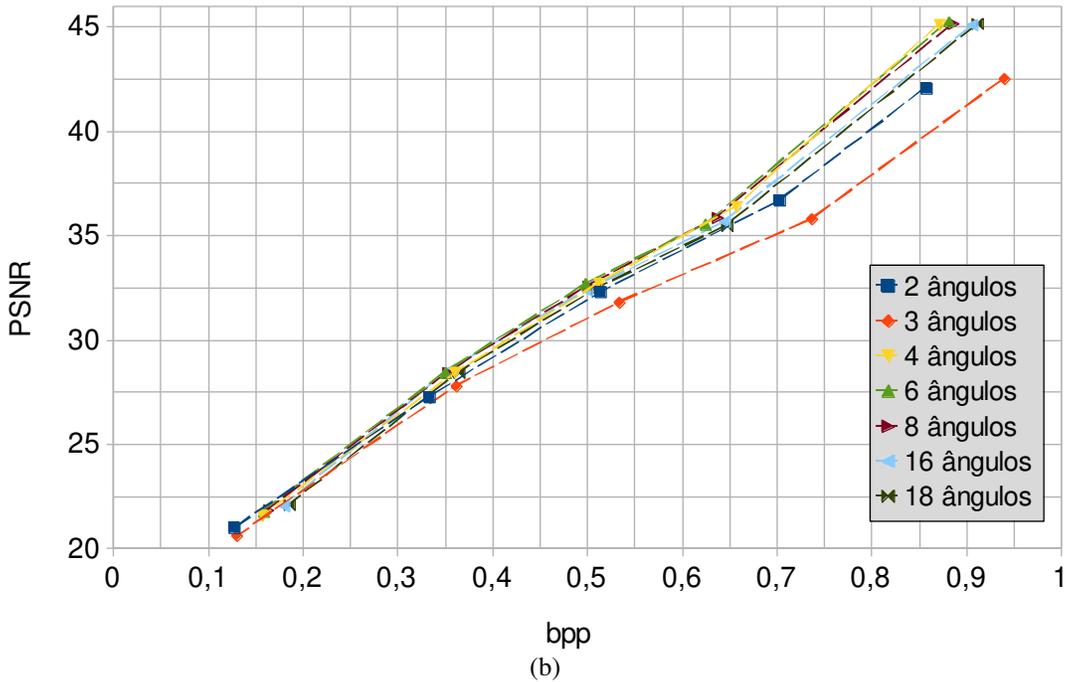
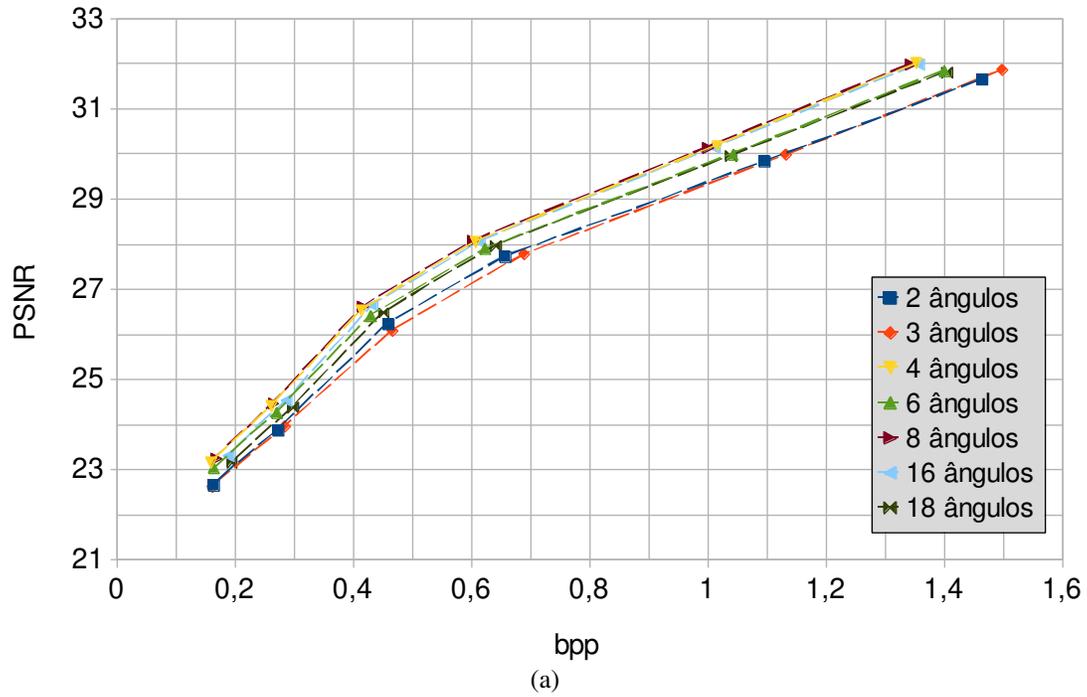


Figura 7.6 – Resultados da variação do número de ângulos (a) imagens naturais (b) desenhos.

Tabela 7.1 – Tempo de processamento para imagens naturais, variando número de ângulos.

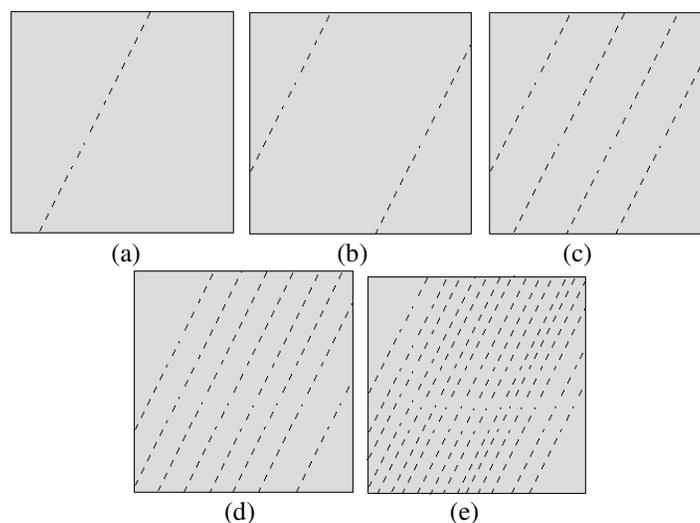
Nº de ângulos	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
2	213,46	155,53	11,07
3	283,60	149,40	12,51
4	156,49	55,83	4,03
6	526,46	151,83	12,64
8	296,41	61,09	4,88
16	587,03	59,52	6,10
18	1533,15	150,24	18,61

Tabela 7.2 – Tempo de processamento para desenhos, variando número de ângulos.

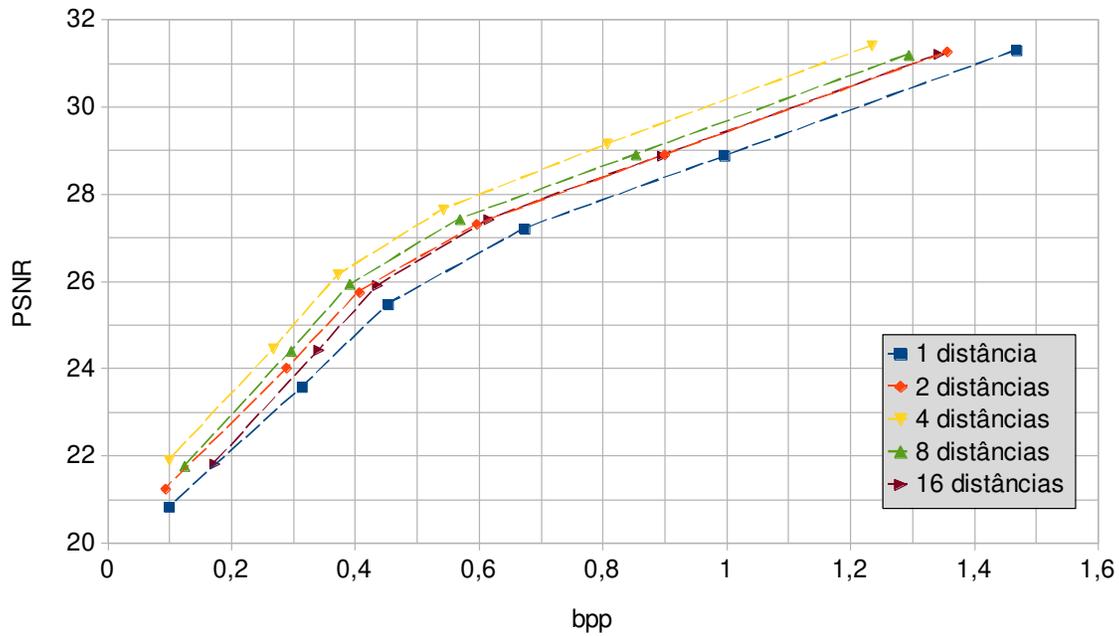
Nº de ângulos	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
2	66,97	7,26	5,66
3	96,50	7,20	6,35
4	77,25	4,79	3,81
6	183,82	7,14	6,19
8	150,94	4,73	4,27
16	304,90	4,79	5,29
18	471,36	7,31	8,78

7.3.2 Determinação do Número de Distâncias

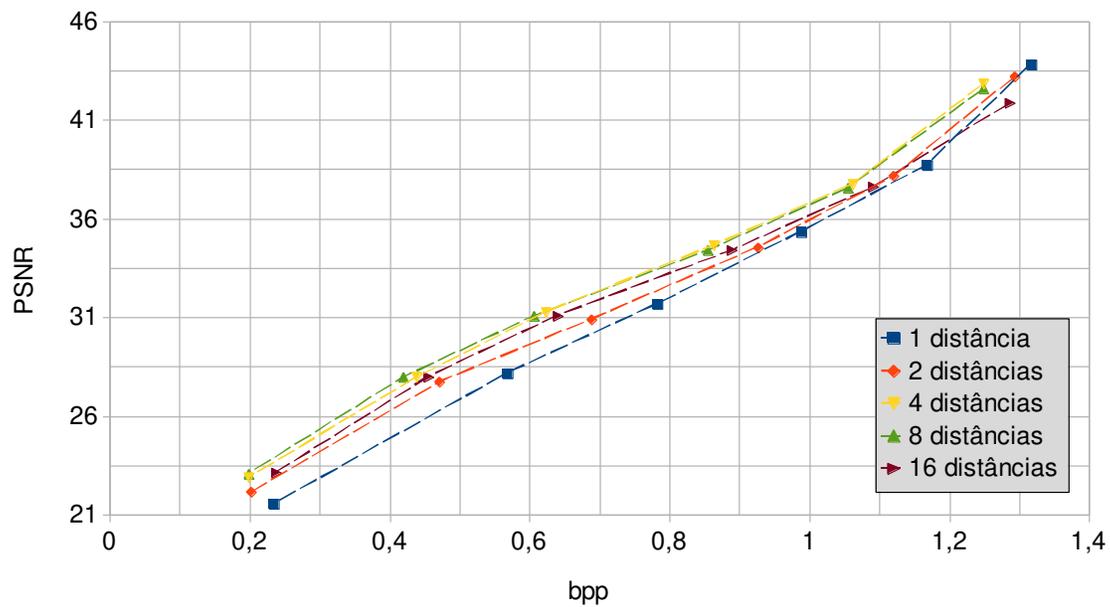
A distância d do contorno na determinação do *wedge* também é um parâmetro discretizado e deve ser escolhido a priori o número máximo de distâncias possíveis. A Figura 7.7 mostra diferentes divisões de distância utilizadas. Na prática, o espaçamento entre cada linha tende a variar com o ângulo θ utilizado, devido à discretização do domínio da imagem.

Figura 7.7 – Variação do número de distâncias na determinação do *wedge* (a) 1, (b) 2, (c) 4, (d) 8, (e) 16.

Na Figura 7.8 são mostradas curvas de desempenho da codificação variando o número de distâncias d possíveis para os *wedges* do dicionário. Assim como no caso da resolução angular, valores intermediários aos mostrados não apresentam ganho significativo devido à discretização dos elementos da imagem. Os resultados mostram que o desempenho ótimo é encontrado para 4 distâncias. Nessas curvas foram utilizados 8 ângulos possíveis, por ser o valor ótimo encontrado na seção 7.3.1. Considerações análogas acerca dos valores médios de PSNR e tempos de processamento entre imagens naturais e imagens tipo “desenho” foram mantidas.



(a)



(b)

Figura 7.7 – Resultados da variação do número de distâncias (a) imagens naturais (b) desenhos.

Pelas Tabelas 7.3 e 7.4, o tempo de geração da árvore *wedgelet* é diretamente proporcional ao número de distâncias, sendo que os tempos de codificação e decodificação não têm variação significativa, como na variação do número de ângulos. Novamente, para as imagens de desenho, o processo é bem mais rápido (tempo caindo quase à metade em relação às imagens naturais).

Na comparação subjetiva da Figura 7.8 observa-se uma melhoria da representação de acordo com o aumento do dicionário, principalmente no número de distâncias. Além disso, os contornos do ombro são melhor reconstruídos com dicionários maiores. A Figura 7.8.b corresponde a 8 ângulos e

4 distâncias, os melhores valores encontrados.

Tabela 7.3 – Tempo de processamento para fotografias, variando número de distâncias.

Nº de distâncias	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
1	87,03	59,23	4,19
2	156,85	60,83	4,06
4	291,19	59,23	4,18
8	578,28	60,08	5,60
16	1163,21	61,61	8,66

Tabela 7.4 – Tempo de processamento para desenhos, variando número de distâncias.

Nº de distâncias	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
1	50,14	12,56	8,41
2	89,83	12,77	8,09
4	181,10	13,02	8,14
8	347,18	12,74	8,89
16	680,00	13,01	11,16



Figura 7.8 – Comparação subjetiva para a imagem ‘cameraman’ a 1 bpp (a) original (a) *wedgelet* de 2 ângulos e 4 distâncias (b) *wedgelet* de 8 ângulos e 4 distâncias (c) *wedgelet* de 8 ângulos e 1 distância.

7.3.3 Determinação da Função *Wavelet*

Para codificar regiões de textura, isto é, sem contornos pronunciados, propõe-se utilizar as transformadas *wavelet* apresentadas no capítulo 5, em blocos de imagem, seguido da codificação SPIHT (*Set Partitioning In Hierarchical Trees* [112]) modificada para otimizar a função taxa-distorção (eq. 24) ao invés de obter uma taxa fixa.

Assim como o dicionário *wedgelet* foi parametrizado buscando otimizar a relação taxa-distorção, é necessário definir a melhor função *wavelet* para a codificação híbrida. Para isso são testadas 2 famílias de funções muito usadas em aplicações de processamento de imagens: as *wavelets* ortogonais de Daubechies e as *wavelets* biortogonais, também chamadas de CDF (Cohen-Daubechies-Fauveau). O comprimento do filtro usado tem grande influência também. O método proposto aplica a TWD em blocos, que pode gerar artefatos quando usadas baixas taxas, os quais crescem com o comprimento do filtro. Filtros maiores correspondem às funções com maior número de momentos nulos, aproximando melhor seções irregulares do sinal. Como a TWD será aplicada em regiões suaves e de textura, não há necessidade de filtros muito grandes.

Foram utilizadas as funções de db1(Haar), db2, db4, CDF 5/3 e CD9/7. Os resultados encontram-se na Figura 7.9.

Observa-se que a função de Haar obteve os melhores resultados para ambos os tipos de imagens, conseguindo uma vantagem muito superior para as imagens de desenho. Isso confirma que o tamanho do filtro deve ser mínimo para o processamento por blocos. A variação no tempo de processamento entre as funções *wavelet* é desprezível frente aos outros tempos envolvidos no sistema, como é possível ver nas Tabelas 7.5 e 7.6.

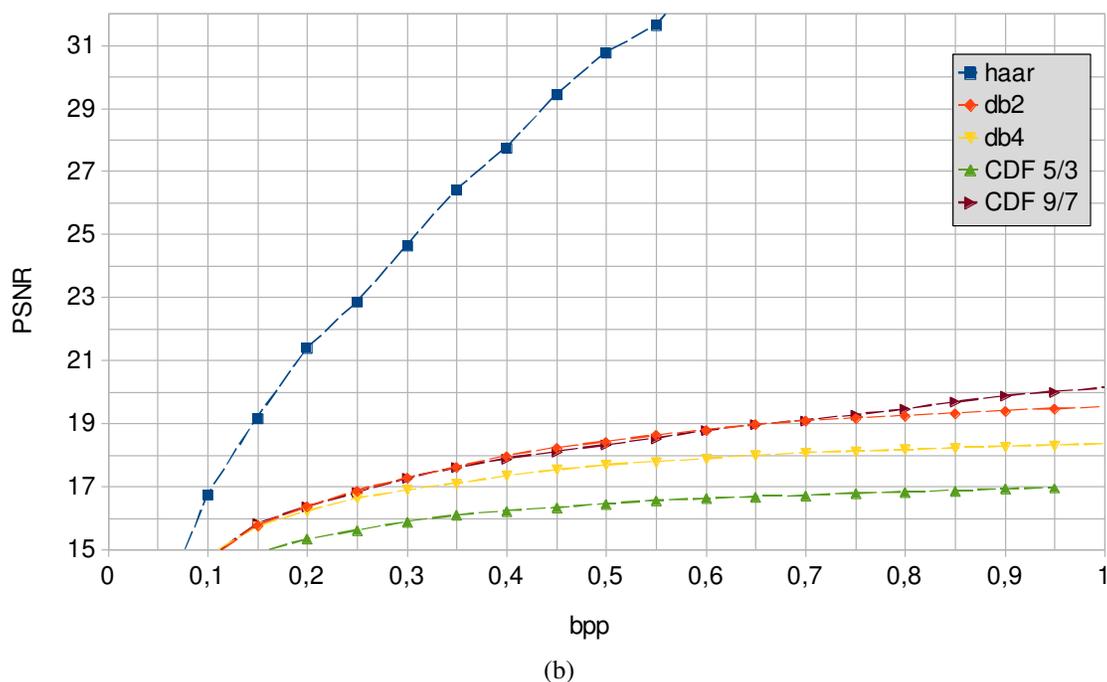
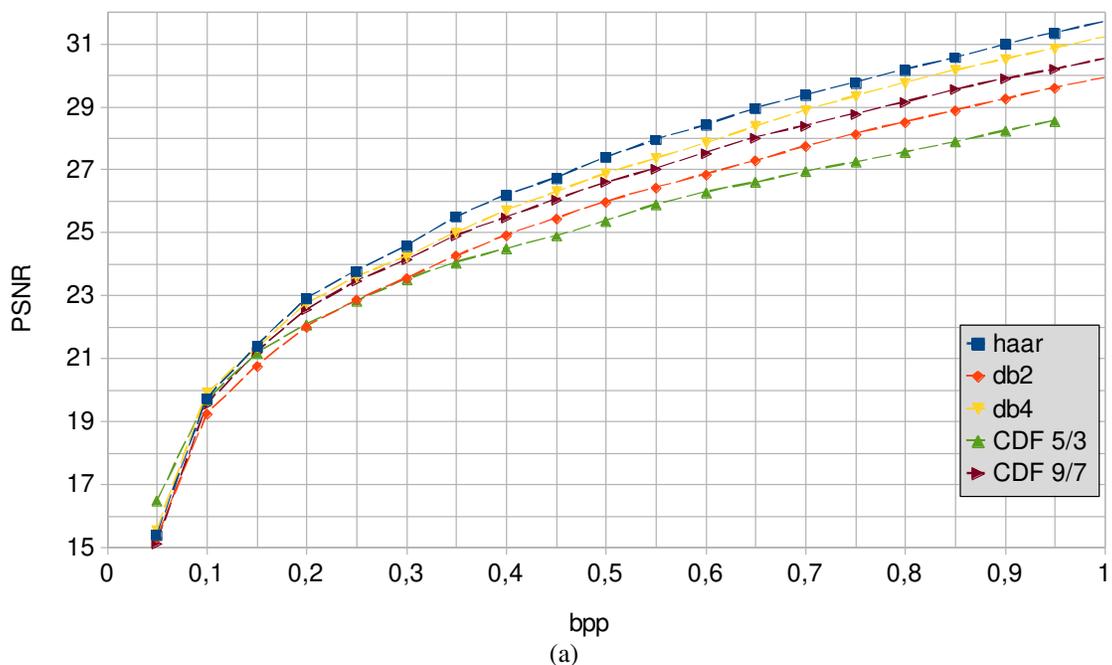


Figura 7.9 – Resultados da variação da função *wavelet* (a) imagens naturais (b) desenhos.

Tabela 7.5 – Tempo de processamento para imagens naturais, variando a função *wavelet*.

Função <i>wavelet</i>	Tempo de codificação (s)	Tempo de decodificação (s)
Haar	3,90	3,60
db2	3,90	3,63
db4	3,96	3,67
CDF 5/3	3,96	3,66
CDF 9/7	3,93	3,65

Tabela 7.6 – Tempo de processamento para desenhos, variando a função *wavelet*.

Função <i>wavelet</i>	Tempo de codificação (s)	Tempo de decodificação (s)
Haar	3,77	3,43
db2	3,81	3,50
db4	3,88	3,59
CDF 5/3	3,87	3,52
CDF 9/7	3,84	3,51

7.4 Codificação *Wedgelet-Wavelet*

Definidos os melhores parâmetros de codificação *wedgelet* e *wavelet*, propõe-se um método de codificação híbrido (WWB, Figura 7.1), que utilize para cada bloco de imagem o melhor dicionário de codificação, *wedgelet* ou *wavelet*, de acordo com uma função de custo taxa-distorção (equação 6.26).

O algoritmo SPIHT foi modificado de forma a não utilizar taxa de dados fixa, mas variar a taxa de dados de acordo com essa função de custo. Espera-se com esse método que regiões *cartoon* da imagem tenham o contorno bem preservado pelo uso do dicionário e regiões de textura utilizem a TWD. Para baixas taxas de dados, a tendência é que o codificador dê preferência à utilização de *wedgelet* e TWD para as altas taxas de dados.

Os resultados da codificação híbrida encontram-se na Figura 7.10, em comparação com as codificações por blocos *wedgelet* e *wavelet* puras utilizando os mesmos parâmetros (8 ângulos, 4 distâncias, *wavelet* Haar). Do ponto de vista de taxa-distorção, para as imagens naturais observa-se que o método WWB obtém desempenho superior à codificação *wavelet* para baixas taxas (conseguindo ganhos de até 1,5 dB) e se aproxima da codificação *wedgelet* (menos de 0,5 dB de diferença). Para taxas maiores do que 0,5 bpp o método WWB se aproxima da curva *wavelet* (diferença de 0,3 dB) e supera o método de *wedgelets* sozinho (ganho maior que 0,5 dB). No geral, o WWB tem uma qualidade uniforme em termos de PSNR para todas as taxas.

Para as imagens de desenho, o desempenho da *wavelet* sozinho é muito ruim e o método WWB consegue ser superior aos dois métodos comparados (ganhos em torno de 0,5 dB em relação às *wedgelets* e maiores que 5 dB em relação às *wavelets*). Esse fato, combinado com a análise subjetiva posteriormente apresentada, torna o WWB um bom método de codificação para aplicações que envolvam desenhos, mapas, imagens com textos, dentre outras.

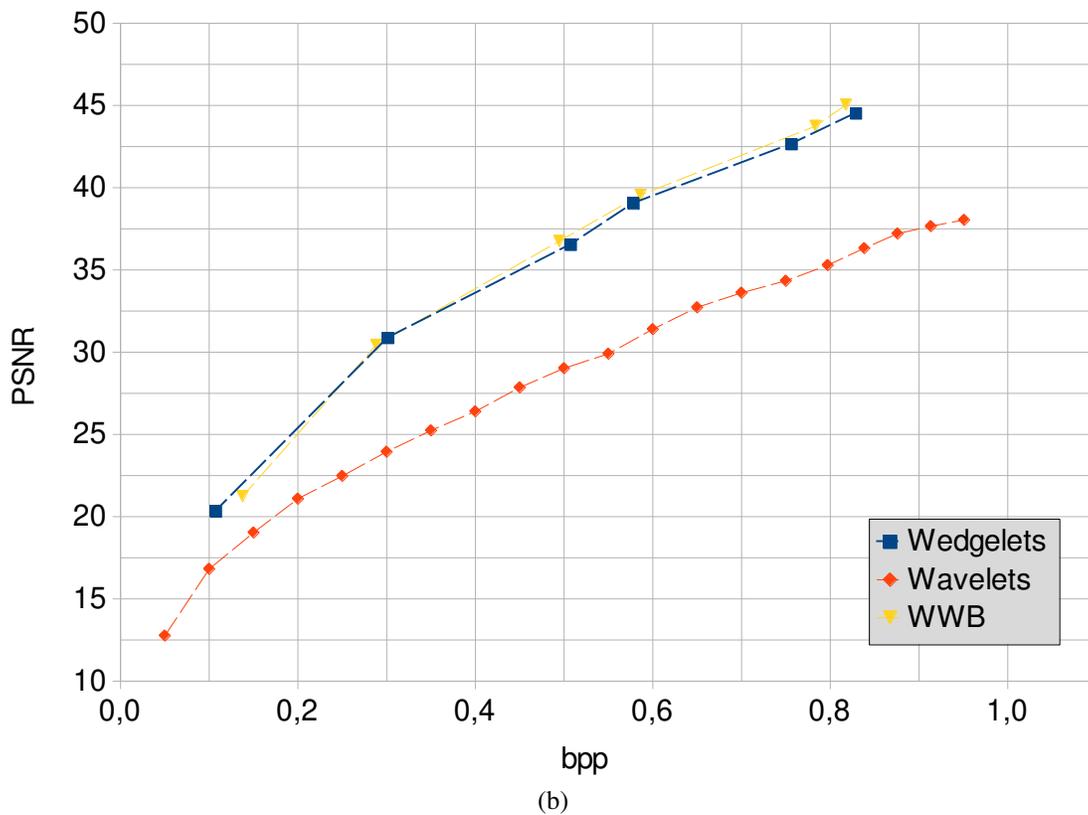
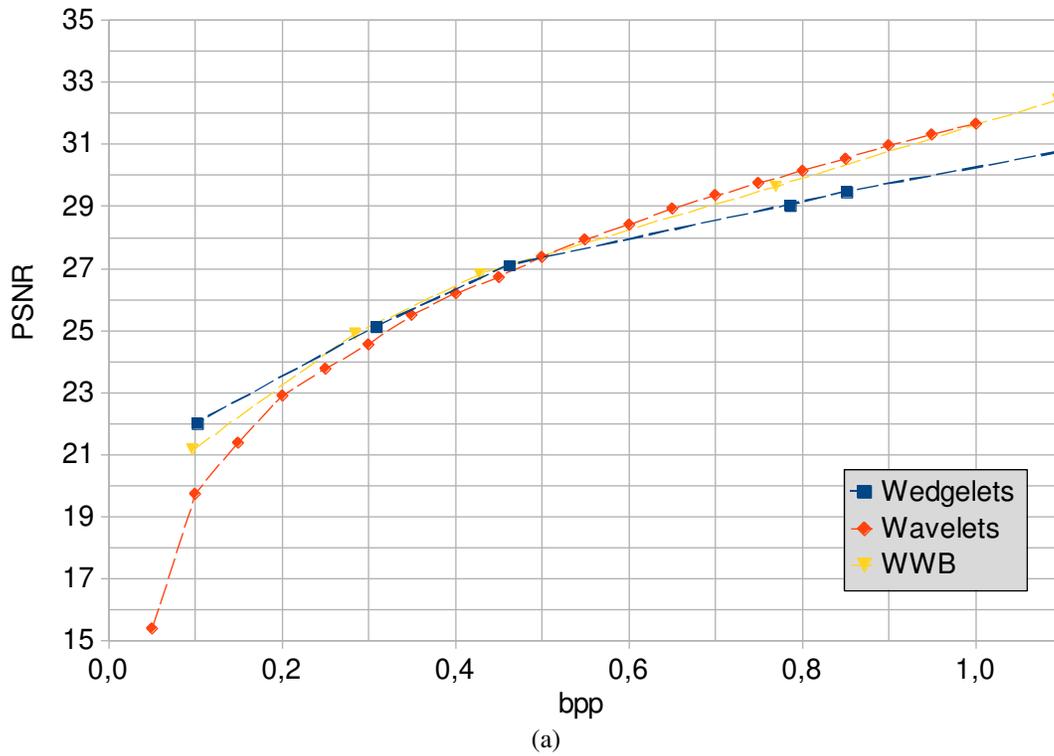


Figura 7.10 – Resultados do método WWB (a) imagens naturais (b) desenhos.

Conforme previsto, para imagens naturais, a tendência é que o sistema utilize mais componentes *wedgelet* quando são usadas baixas taxas de *bits* e passe a usar *wavelets* nas altas

taxas. Esse comportamento pode ser visto na Figura 7.11. Para imagens de desenho, ocorre uma inflexão a partir de 0,5 bpp, isto é, quando em altas taxas o sistema continua usando componentes *wedgelets*, pois essas são mais adequadas já que não há detalhes na imagem.

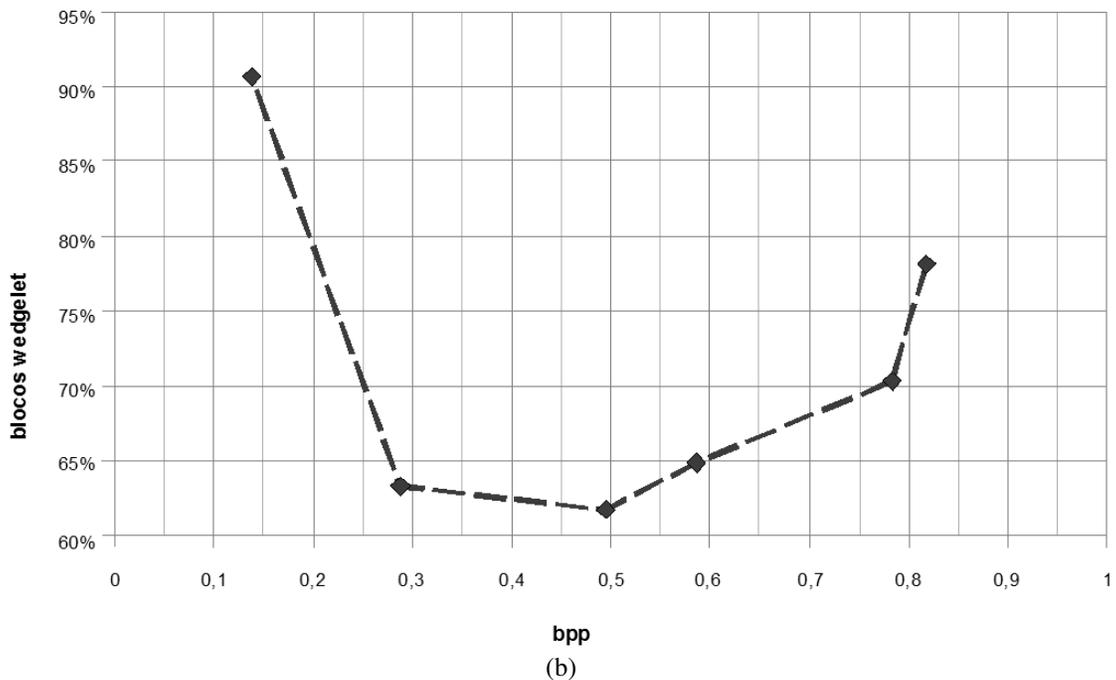
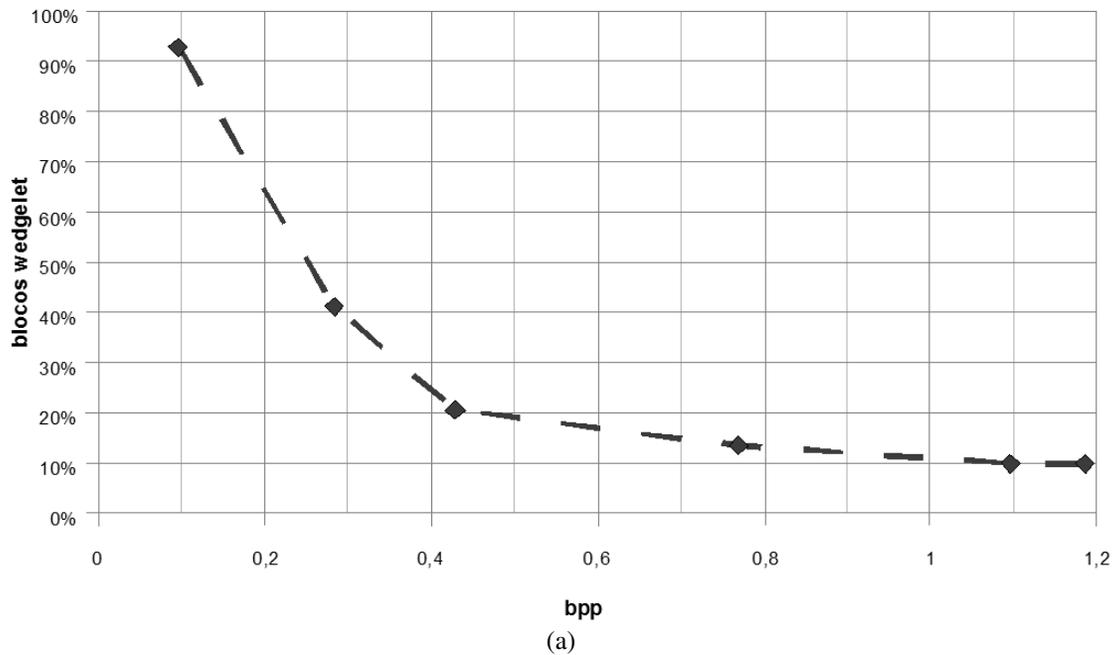


Figura 7.11 – Uso de componentes *wedgelet* de acordo com a taxa (a) imagens naturais (b) desenhos.

Nas Tabelas 7.7 e 7.8 são mostrados os tempos de processamento para cada um dos processos. Nota-se que a etapa de *wedgelets* aumenta significativamente o tempo do processo de codificação, tendo menor impacto na decodificação. Assim como no caso da codificação *wedgelet* isolada, a complexidade do codificador é muito maior do que do decodificador, deixando esse método mais

adequado para aplicações de *broadcast*. No entanto, mesmo em aplicações onde a codificação das imagens é feita por um dispositivo móvel a tendência é que o poder de processamento dos dispositivos reduza essa desvantagem. No mercado a próxima linha de *smartphones* anunciada já apresenta capacidade computacional igual ou superior a de computadores de mesa [113].

A geração da árvore, que é o processo mais custoso, pode ser desenvolvida de maneira mais rápida futuramente modificando a forma de se realizar a decomposição ou diminuindo o dicionário. Além disso, o processo de escolha entre símbolos *wedgelet* e *wavelet* faz com que a codificação seja mais lenta que os dois processos isolados. Para acelerar essa etapa deve-se pensar futuramente em comparações parciais dos blocos.

Tabela 7.7 – Tempo de processamento dos diferentes métodos para imagens naturais.

Método	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
<i>Wedgelet</i>	291,19	59,23	4,18
<i>Wavelet</i>	-	3,90	3,60
WWB	306,95	80,88	4,37

Tabela 7.8 – Tempo de processamento dos diferentes métodos para desenhos.

Método	Tempo geração da árvore (s)	Tempo de codificação (s)	Tempo de decodificação (s)
<i>Wedgelet</i>	181,10	13,02	8,14
<i>Wavelet</i>	-	3,77	3,43
WWB	174,41	24,42	3,62

As Figuras 7.12 a 7.17 mostram a codificação das imagens '*cameraman*', '*peppers*' e 'Brasil' e o mapeamento dos blocos da imagem que foram codificados por *wedgelets*. Observa-se que a grande vantagem do método proposto é a melhoria local da qualidade da imagem de acordo com a classificação em *cartoon* e textura. Como esperado, regiões suaves e de contorno bem definido (como o ombro na imagem '*cameraman*' e a região de contorno das frutas em contato com o fundo na imagem '*peppers*') foram escolhidas pela codificação *wedgelet*. Para as imagens de desenho mais blocos usam de componentes *wedgelet*.

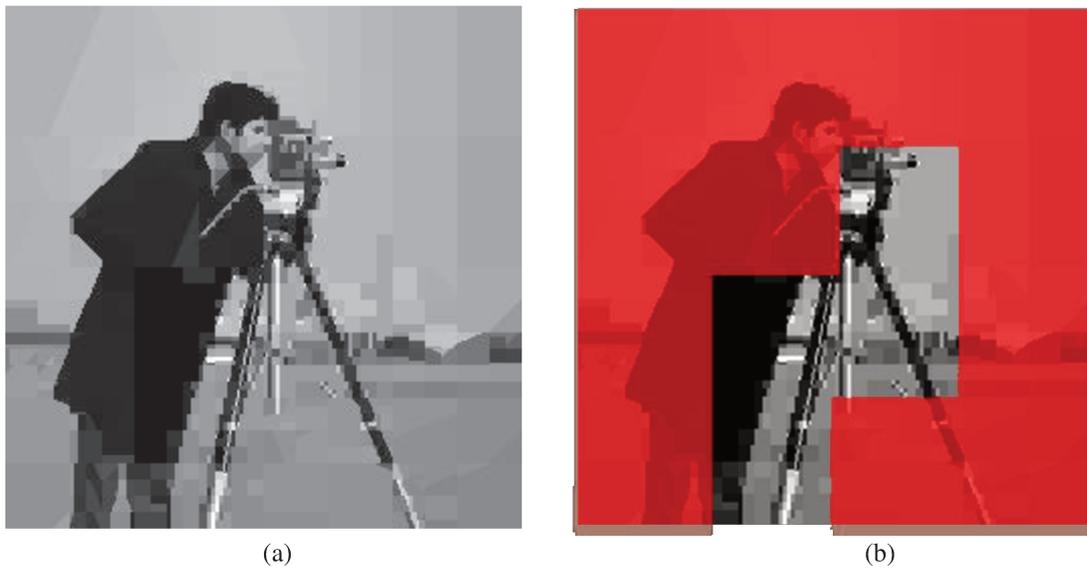


Figura 7.12 – Codificação da imagem ‘cameraman’ a 0,1736 bpp (a) imagem recuperada PSNR = 23,42 dB (b) mapeamento dos blocos codificados por *wedgelets* destacados em vermelho.

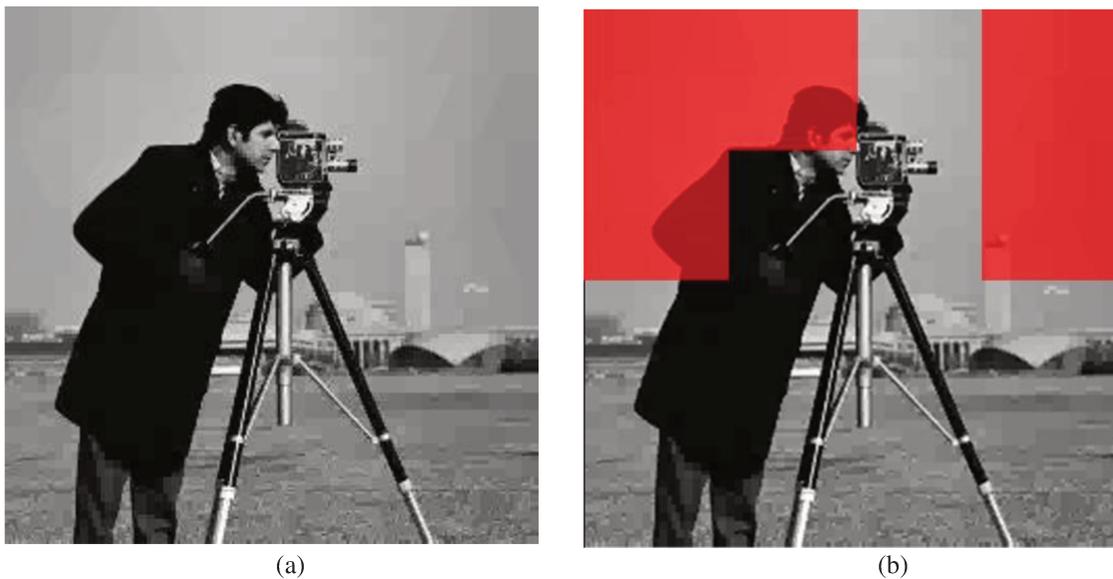


Figura 7.13 – Codificação da imagem ‘cameraman’ a 0,6507 bpp (a) imagem recuperada PSNR = 30,49 dB (b) mapeamento dos blocos codificados por *wedgelets* destacados em vermelho.

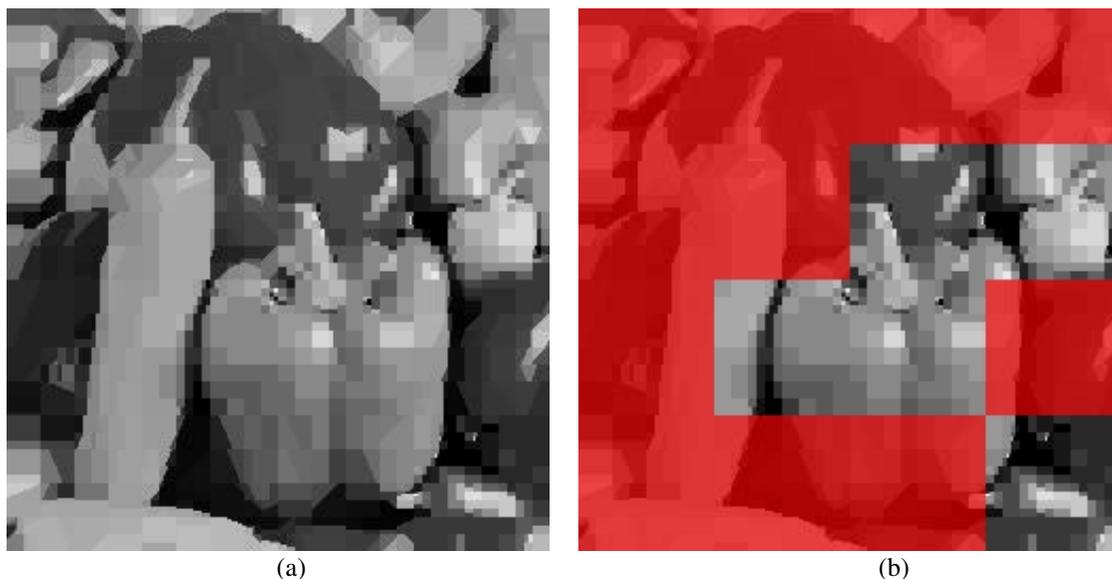


Figura 7.14 – Codificação da imagem ‘peppers’ a 0,2336 bpp (a) imagem recuperada PSNR = 23,7389 dB (b) blocos codificados por *wedgelets* destacados em vermelho.

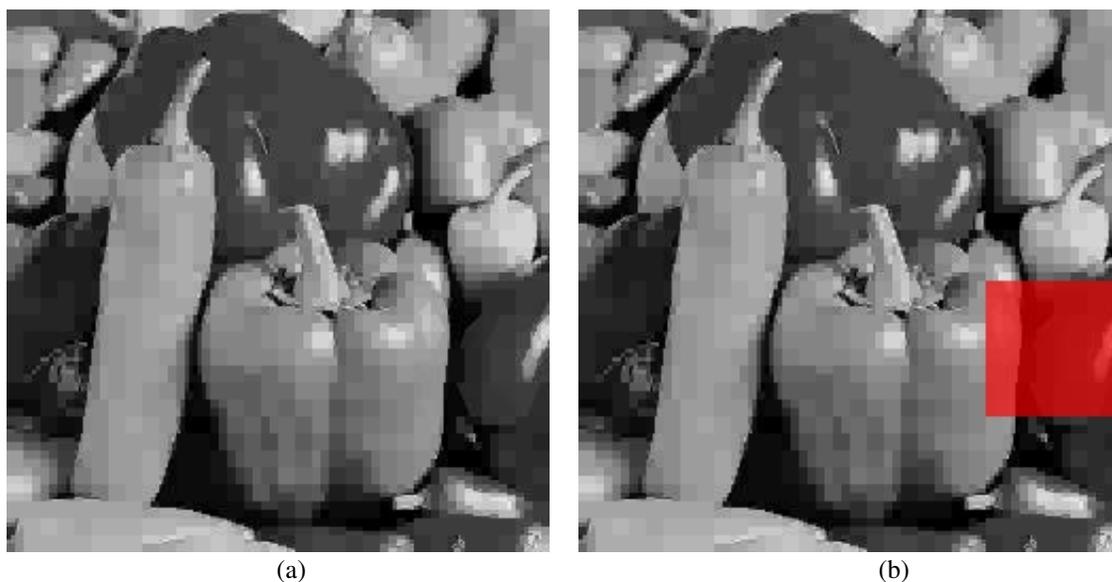
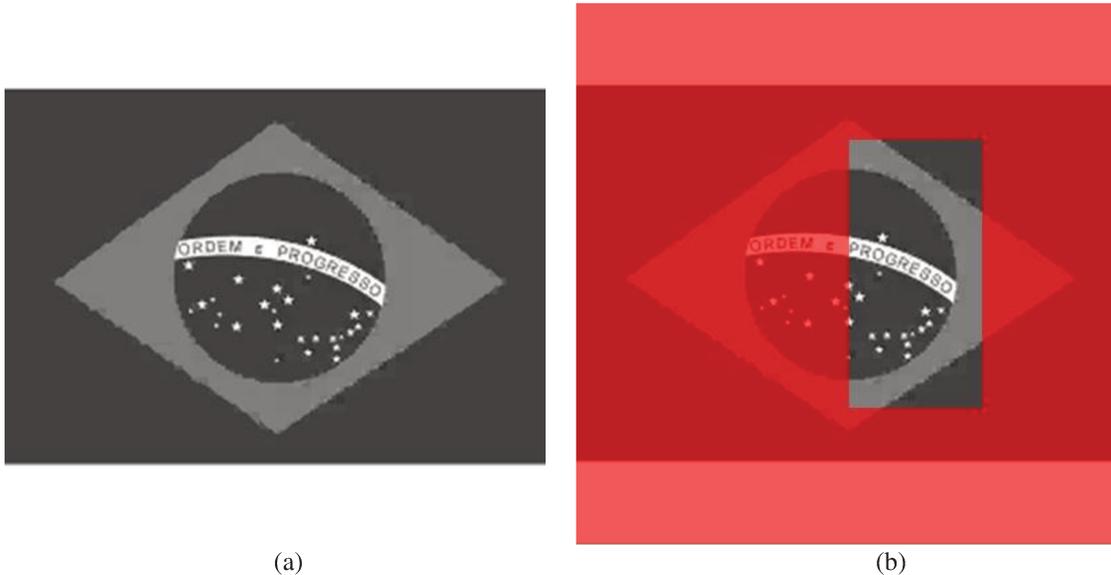
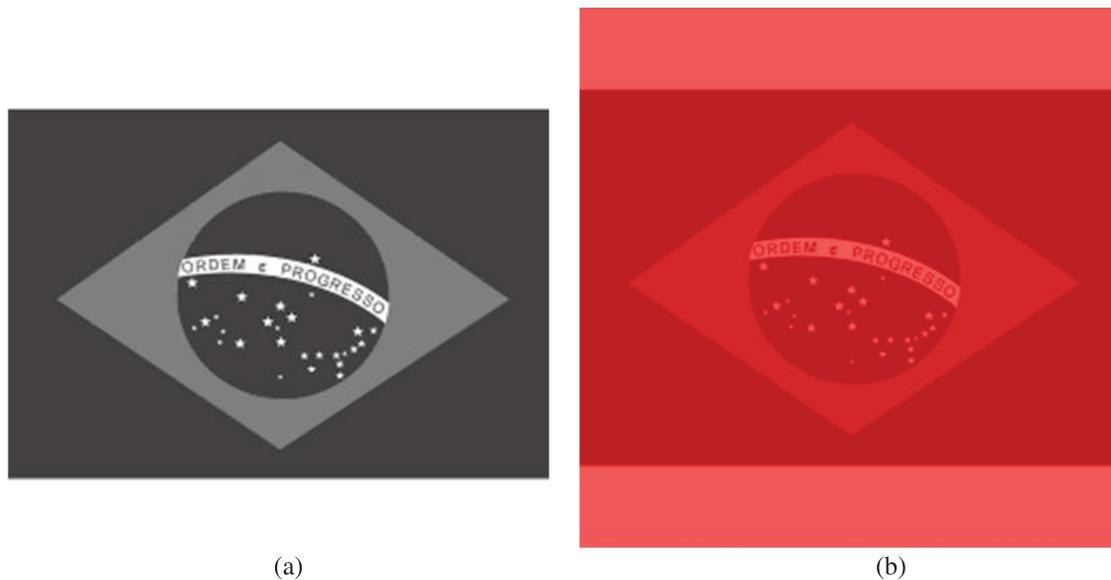


Figura 7.15 – Codificação da imagem ‘peppers’ a 0,3794 bpp (a) imagem recuperada PSNR = 26,0650 dB (b) blocos codificados por *wedgelets* destacados em vermelho.



(a) (b)
 Figura 7.16 – Codificação da imagem ‘Brasil’ a 0,3431 bpp (a) imagem recuperada PSNR = 37,77 dB (b) mapeamento dos blocos codificados por *wedgelets* destacados em vermelho.



(a) (b)
 Figura 7.17 – Codificação da imagem ‘Brasil’ a 0,5828 bpp (a) imagem recuperada PSNR = 51,79 dB (b) mapeamento dos blocos codificados por *wedgelets* destacados em vermelho.

Nas Figuras 7.18 a 7.20 são mostradas comparações visuais das imagens codificadas por *wedgelets*, *wavelets* e o método WWB. Observa-se que o WWB apresenta menos artefatos que o método de *wavelets* e tem melhor qualidade que as *wedgelets* sozinhas para

regiões com mais detalhe, como na região da mão.

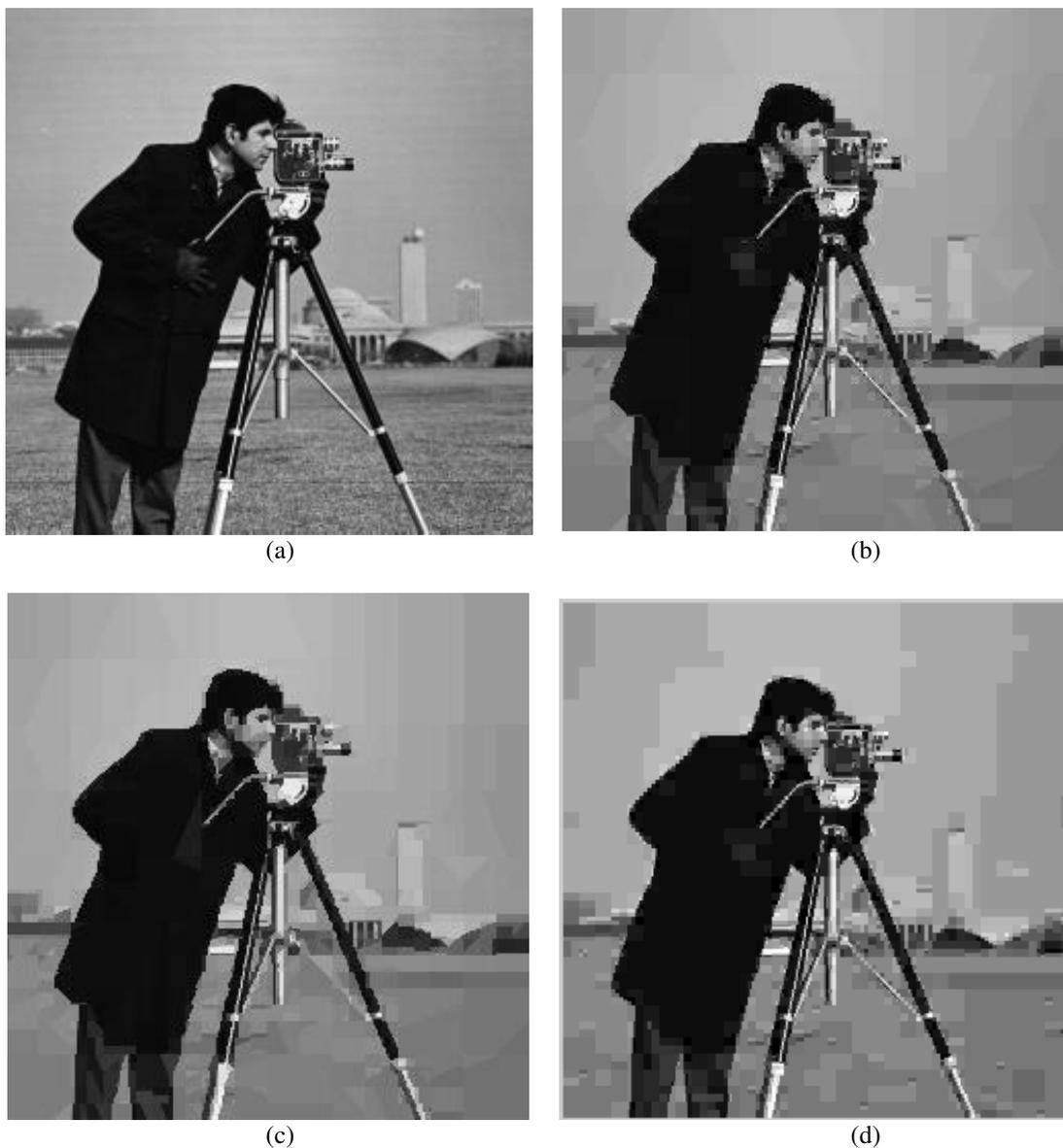


Figura 7.18 – Comparação subjetiva da imagem ‘cameraman’ com métodos *wavelet* (a) original (b) WWB 0,3251 bpp PSNR = 26,5223 dB (c) *wedgelets* 0,3194 bpp 25,8670 dB (d) *wavelets* 0,3194 bpp PSNR = 25,6961 dB.

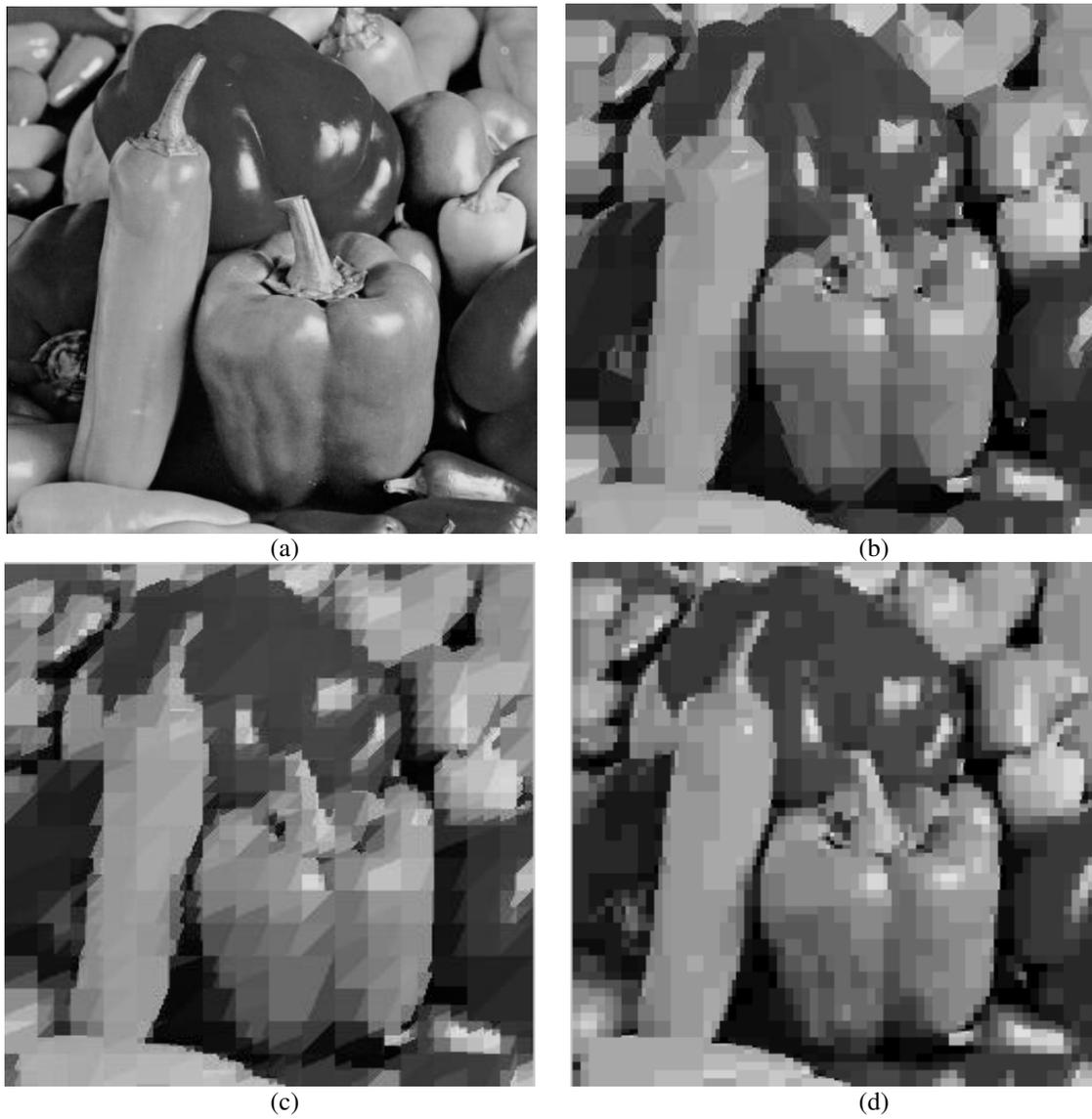


Figura 7.19 – Comparação subjetiva da imagem ‘peppers’ com métodos *wavelet* (a) original (b) WWB 0,2336 bpp PSNR = 23,7389 dB (c) *wedgelets* 0,2404 bpp 22,6040 dB (d) *wavelets* 0,2337 bpp PSNR = 23,9179 dB

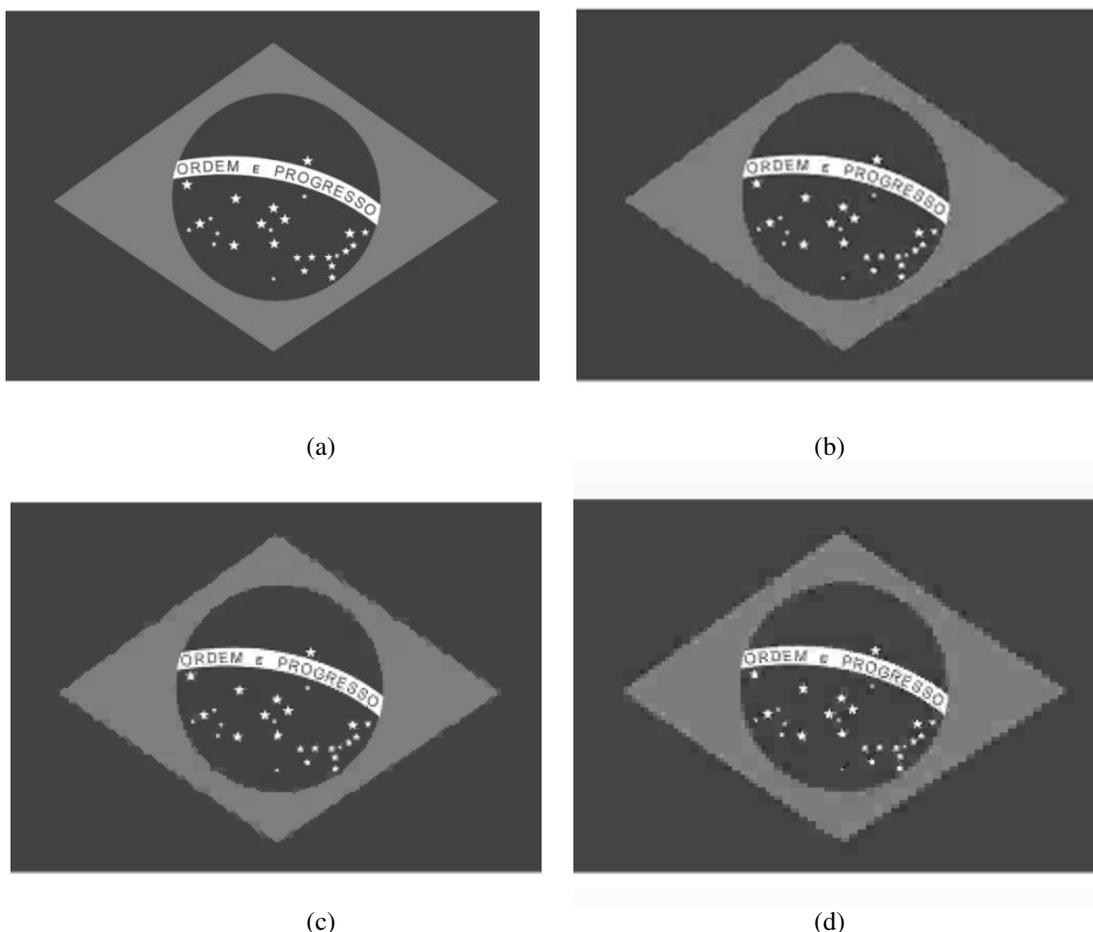


Figura 7.20 – Comparação subjetiva da imagem ‘Brasil’ com métodos *wavelet* (a) original (b) WWB 0,3431 bpp PSNR = 37,77 dB (c) *wedgelets* 0,3560 bpp 37,6629 dB (d) *wavelets* 0,3500 bpp PSNR = 32,6461 dB.

7.4.1 Comparações Adicionais com Métodos *Wedgelet*

Métodos *wedgelet* de codificação de imagens são bastante recentes e é difícil encontrar um padrão para comparação. Os artigos consultados para comparação apresentam poucos resultados e em geral sem menção a tempos de processamento. A maioria dos trabalhos na área trata apenas de imagens essencialmente *cartoons*, não conseguindo uma codificação adequada de imagens naturais. Além disso, na maior parte das vezes, os métodos se baseiam apenas em gerar símbolos *wedgelets* não-binários, que invalidam a comparação para a tarefa de compressão de dados.

Em [109] é feito um estudo aprofundado sobre a decomposição *wedgelet*, conseguindo resultados para diferentes números de ângulos e usando dicionários não-binários. Em [110] os autores disponibilizam um software, que permite reproduzir as curvas apresentadas no trabalho para qualquer imagem, número de ângulos ou parâmetro taxa-distorção. Usando esse *software* e convertendo os símbolos *wedgelets* para uma taxa binária fixa, a Figura 7.21 mostra uma comparação do método WWB com o método de [109]. Nota-se que o WWB consegue valores de

PSNR consideravelmente superiores, principalmente para imagens de desenho.

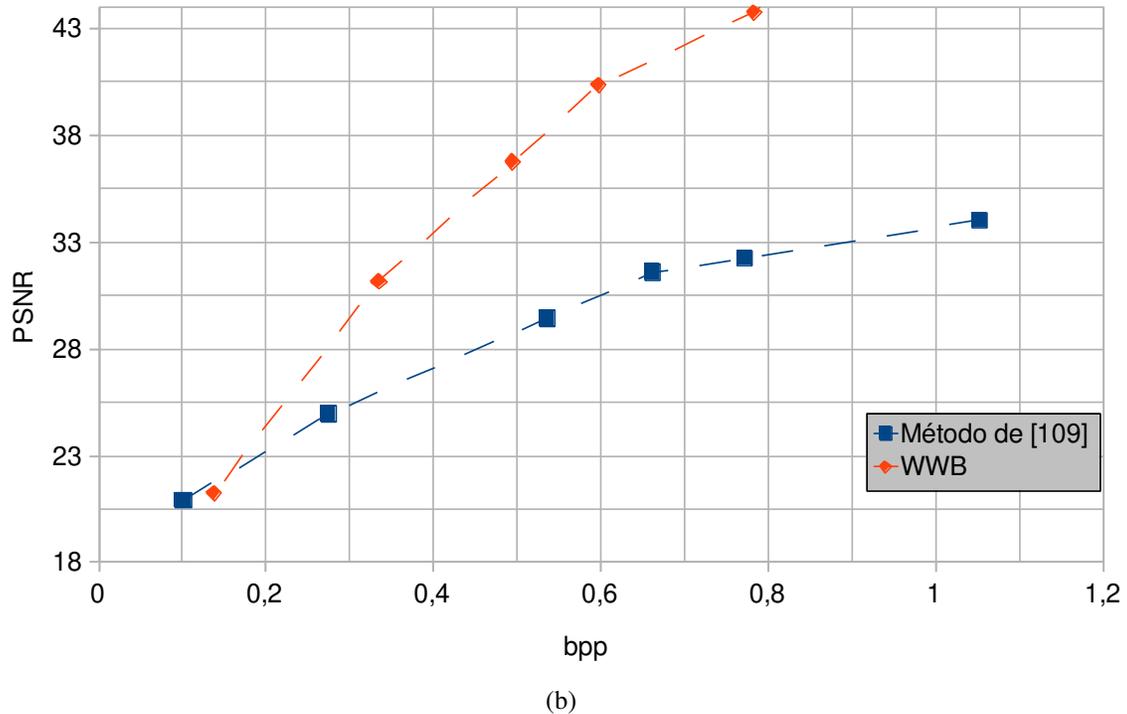
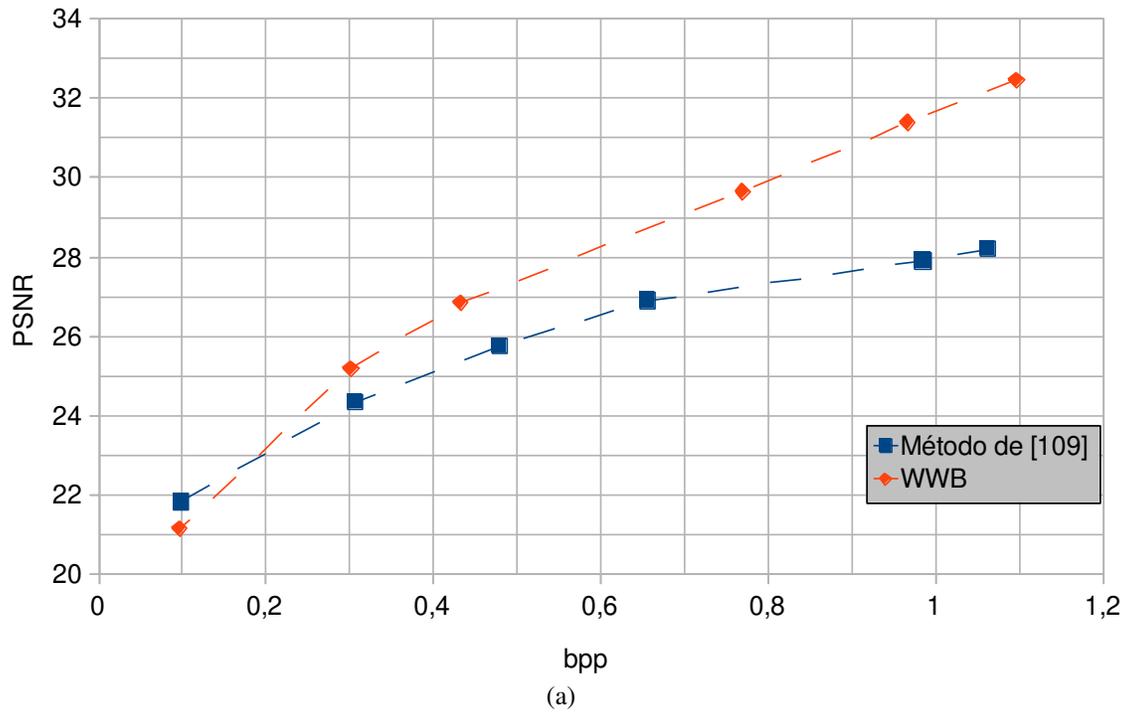
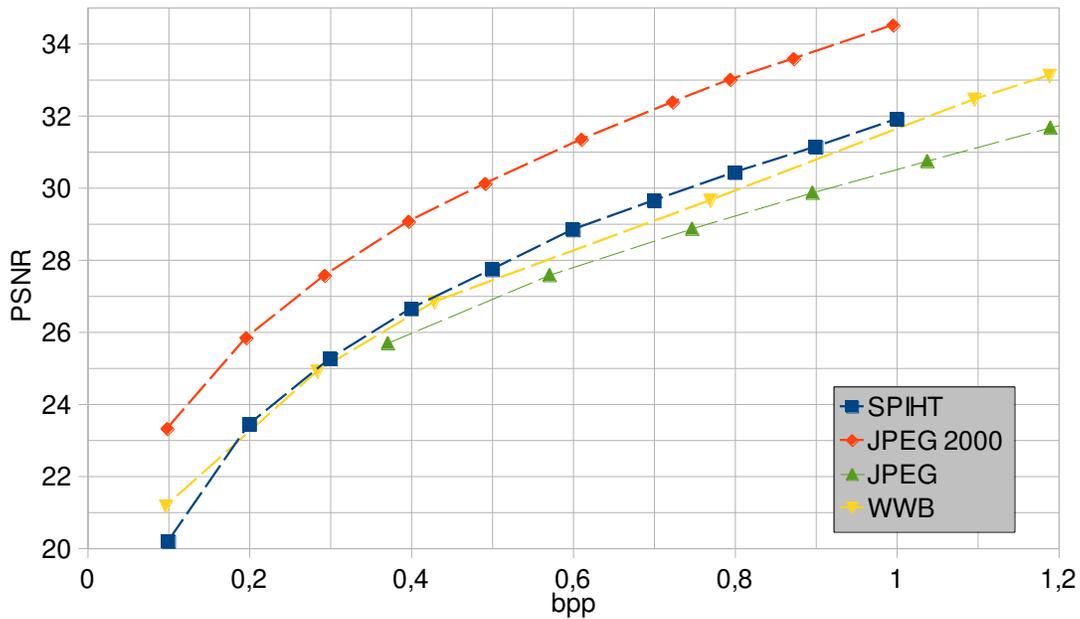


Figura 7.21 – Comparação da codificação *wedgelet* de [109] com símbolos binários e o método *wedgelet-wavelet* proposto (a) imagens naturais (b) desenhos.

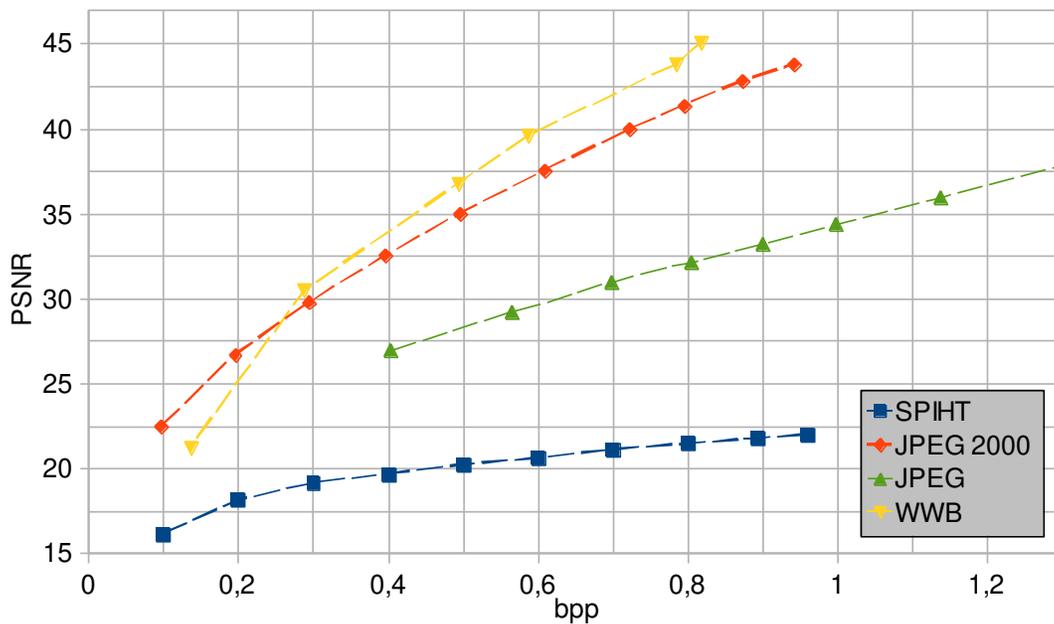
7.4.2 Comparações Adicionais com Métodos *Wavelet*

Na Figura 7.22, é apresentada uma comparação entre o método WWB e sistemas de codificação com *wavelet* biortogonal CDF 9/7, considerada estado-da-arte: codificação SPIHT e o padrão JPEG 2000. Diferente do WWB, esses sistemas aplicam a TWD na imagem toda, não é

dividido em blocos, e a taxa de dados é pré-fixada.



(a)



(b)

Figura 7.22 – Comparação com métodos *wavelet* (a) imagens naturais (b) desenhos.

Apesar de esses sistemas serem muito mais amplos, com etapas de codificação entrópica, predição e parametrização de cabeçalho e de serem padrões consolidados em anos de otimizações, o método WWB se aproximou bastante, com uma diferença de 2,5 dB em relação ao JPEG 2000, menos de 0,5 dB em relação ao SPIHT completo e mais de 0,5 dB superior ao JPEG, para as imagens naturais. Para as imagens de desenho, o método WWB é superior aos 3, com ganhos maiores que 1,5 dB.

Além disso, embora o WWB tenha um resultado em PSNR mais baixo quando a taxa de dados aumenta, tem a vantagem de otimizar localmente a relação taxa-distorção, como pode ser visto pela comparação subjetiva das imagens nas Figuras 7.23 e 7.24. Caso o método proposto fosse baseado na codificação preditiva do JPEG 2000 [1], o desempenho poderia ser melhor do que o atingido, mas a complexidade do algoritmo seria ainda maior. Foi incluída nessa figura também a curva de PSNR para o padrão JPEG, baseado em DCT, popularmente usado, apenas para fins de comparação.

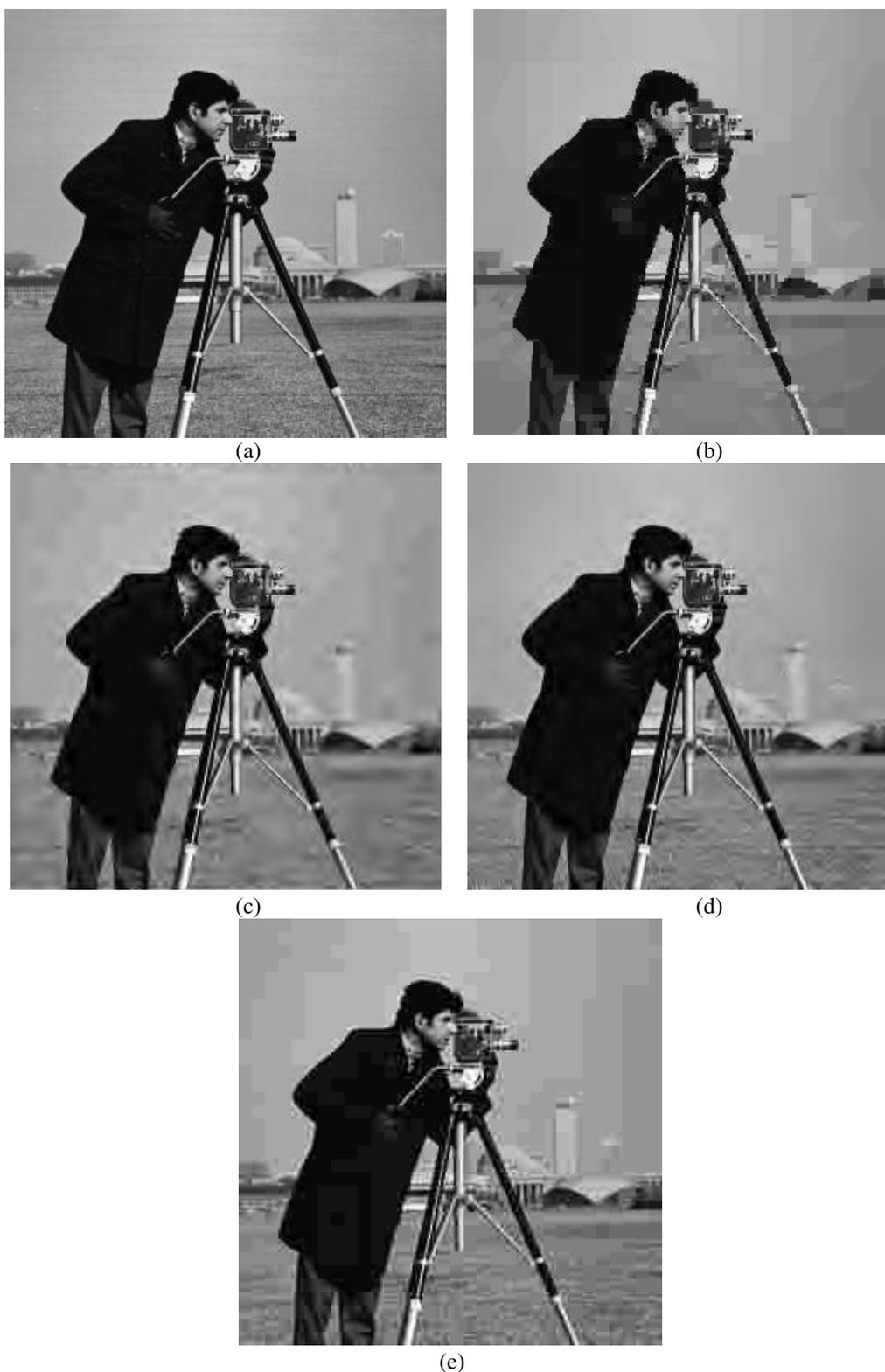


Figura 7.23 – Comparação subjetiva da imagem ‘cameraman’ com métodos *wavelet* (a) original (b) WWB 0,3251 bpp PSNR = 26,5223 dB (c) SPIHT 0,3431 bpp PSNR = 27,0453 dB (d) JPEG 2000 0,3479 bpp PSNR = 29,1556 dB (e) JPEG 0,3347 bpp PSNR = 26,4713 dB.

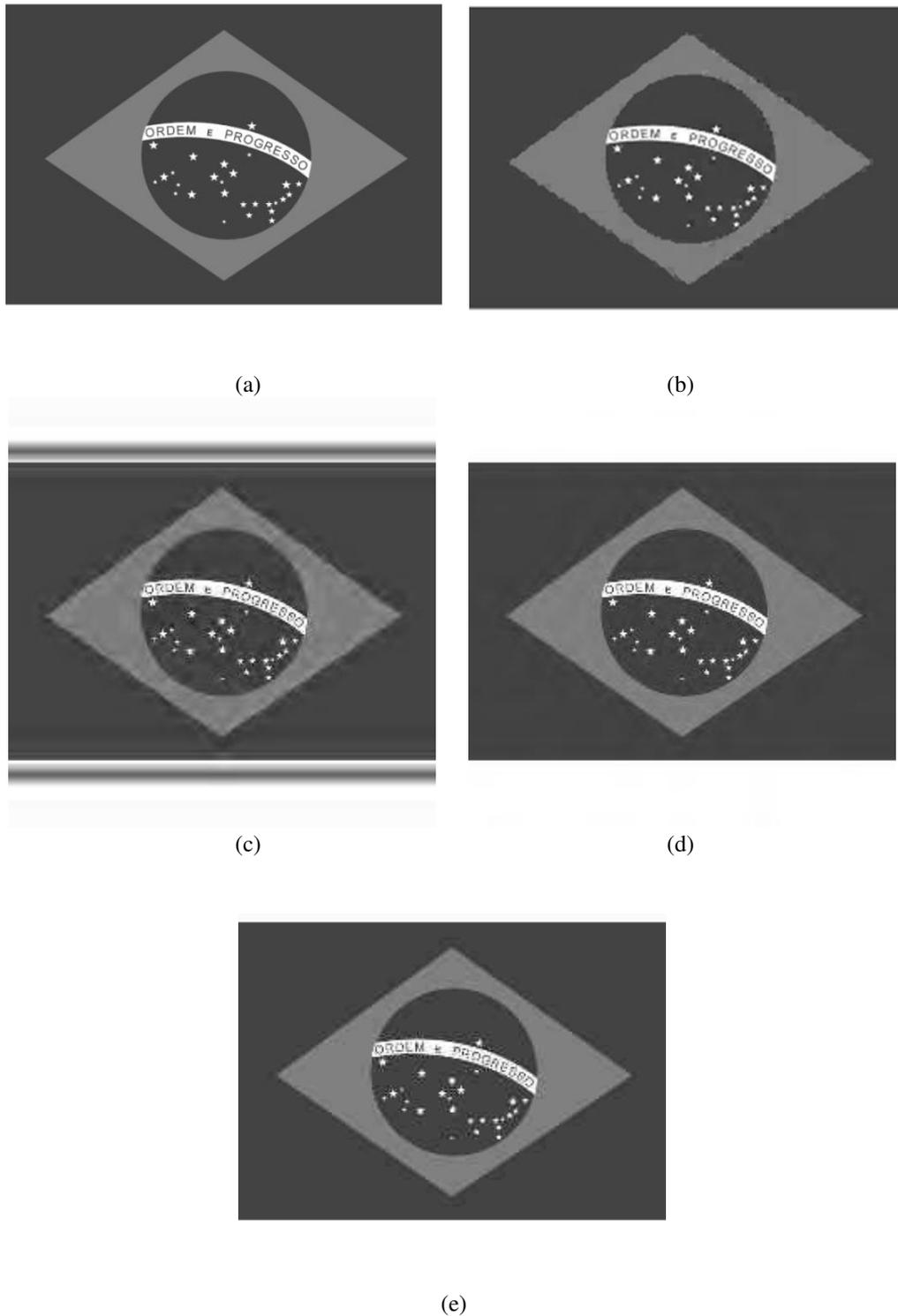


Figura 7.24 – Comparação subjetiva da imagem ‘Brasil’ com métodos *wavelet* (a) original (b) WWB 0,3431 bpp PSNR = 37,77 dB (c) SPIHT 0,3432 bpp PSNR = 17,8789 dB (d) JPEG 2000 0,3477 bpp PSNR = 38,8455 dB (e) JPEG 0,3544 bpp PSNR = 33,2960 dB.

7.4.3 Comparação com outros Métodos *Wedgelet-Wavelet*

Wakin et al. propuseram métodos *wedgelet-wavelet* distintos do proposto neste trabalho [46][105], como mostrado na seção 6.7. Apesar dos resultados muito bons apresentados, foram apresentadas poucas curvas de desempenho para que se possa fazer uma comparação completa.

Considerando apenas a imagem '*cameraman*' mostrada nos artigos, as Figuras 7.25 e 7.26 mostram comparações subjetivas do método proposto neste trabalho e dos 2 métodos propostos por Wakin et al. Não é possível pelo método proposto obter a taxa exata para a comparação devido ao processo de escolha baseado em otimização taxa-distorção, por isso são mostrados resultados em taxas próximas.

Para os métodos da Figura 7.25, nota-se que o desempenho em PSNR foi bastante próximo e a imagem reconstruída pelo método proposto apresenta menos efeito de *ringing* nos contornos, embora algumas regiões da imagem tenham ficado distorcidas, principalmente na região da mão.

Para a Figura 7.26.b, o *ringing* é bastante notável embora o desempenho em PSNR desse codificador seja muito superior. A imagem reconstruída na Figura 7.26.a, embora não tenha *ringing*, tem a aparência muito *cartoonizada*. Como o método de [105] é baseado na SFQ (*Signal Frequency Quantization*) [106], que é um método de aproximação não-linear *wavelet* muito superior ao SPIHT, já era esperado que seu desempenho fosse superior ao método proposto. No entanto, esse ganho tem como contrapartida um grande peso computacional para a quantização adaptativa do SFQ e o método de escolha dos *wedgeprints*, embora não se disponha de dados numéricos no artigo de Wakin et al [105].

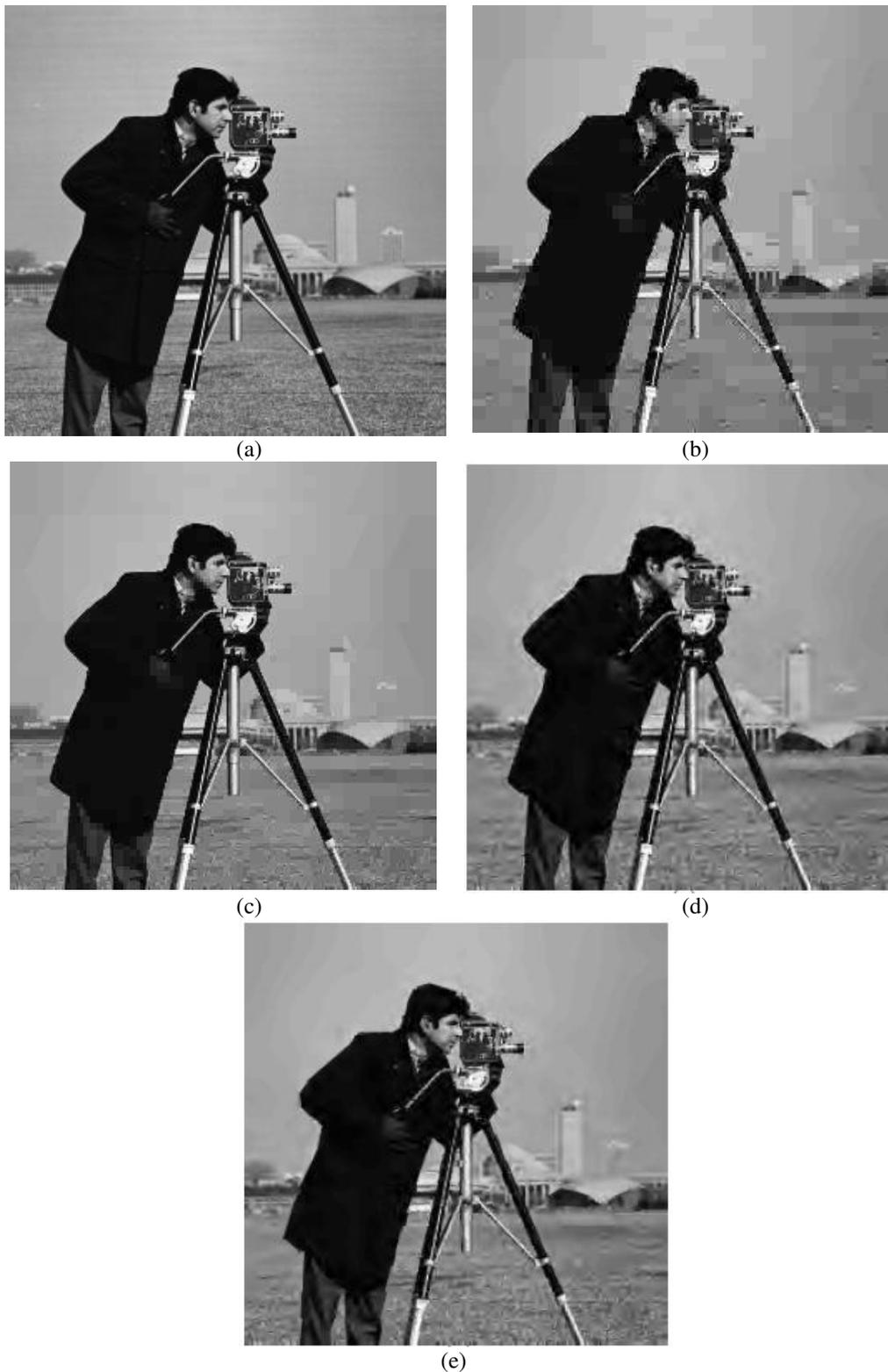


Figura 7.25 - Comparação subjetiva com outros métodos *wedgelet-wavelet* (a)original (b) WWB 0,3522 bpp PSNR = 26,6524 dB (c) WWB 0,6278 bpp PSNR = 30,3371 dB (d) método de [46] a 0,4 bpp PSNR = 28,48 dB (e) método de [46] com filtro posterior a 0,4 bpp PSNR = 27,35 dB.

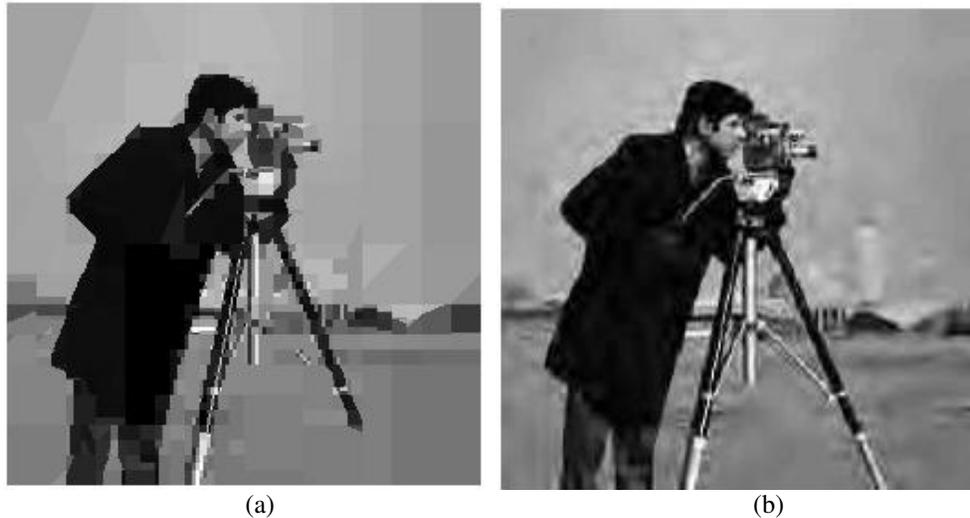


Figura 7.26 - Comparação subjetiva com outros métodos *wedgelet-wavelet* (a) WWB 0,1519 bpp PSNR = 23,0640 dB (b) método apresentado em [105] (W-SFQ) 0,146 bpp PSNR = 25,94 dB.

7.5 Comentários

Neste capítulo, foram apresentados resultados experimentais da codificação *wedgelet-wavelet* proposta (WWB), comparativos a outros métodos. Foram obtidos os melhores parâmetros para a formação do dicionário de cada um dos componentes, de forma a desenvolver um algoritmo híbrido que aproveitasse as vantagens de cada método.

Os resultados mostraram que a integração da TWD na codificação melhora o desempenho obtido apenas com *wedgelets* na maioria dos casos. Em comparação com os métodos baseados somente em TWD feita sem divisão por blocos, o WWB teve o desempenho esperado, isto é, um ganho em PSNR nas baixas taxas de dados e uma perda para as altas taxas. A maior vantagem do método proposto reside no entanto na melhoria local de acordo com a classificação dos blocos em região *cartoon* ou textura, como pode ser constatado nas comparações subjetivas. Para imagens de desenho o ganho de desempenho é ainda maior do que no caso de imagens naturais, um fato que já era esperado considerando que nessas imagens a geometria é mais importante.

O tempo de processamento de cada etapa do processo foi medido e foi constatado que o processo mais custoso computacionalmente é a projeção da árvore *wedgelet* inicial e que esse processo depende diretamente do tamanho do dicionário *wedgelet* considerado. A vantagem é que essa etapa só é realizada pelo codificador, tendo o decodificador um processamento mais simples.

Embora não exista uma literatura sobre codificação *wedgelets* tão vasta e principalmente com resultados comparativos detalhados, foram mostradas comparações adicionais com o método *wedgelet* de [109] e com os métodos híbridos de Wakin et al [46][105]. Com um estudo mais aprofundado sobre como esses métodos funcionam e principalmente como codificam os componentes *wedgelet*, pode-se introduzir ainda mais melhorias na codificação *wedgelet-wavelet*.

Capítulo 8

Conclusões e Perspectivas para Trabalhos Futuros

8.1 Comentários sobre o Contexto e Contribuições

A codificação digital aliada aos avanços da microeletrônica e das telecomunicações permitiu uma gama de novas aplicações multimídia para dispositivos móveis, seja em transmissão ponto-a-ponto ou *broadcast*. No entanto, o ambiente de transmissão móvel é muito restrito e apresenta limitações na banda de transmissão para cada usuário e emissora. No caso da transmissão de imagens e vídeo, isso se traduz em baixas taxas de dados, da ordem menor do que 1 bit por pixel. Assim se tornam necessários algoritmos de compressão eficientes, que preservem a qualidade visual e não atrapalhe a experiência do usuário.

A codificação por transformadas ortogonais e biortogonais (especificamente as transformadas *wavelets*) é considerada atualmente o estado-da-arte. Com uma formulação matemática bastante complexa, as *wavelets* são aplicadas para diferentes objetivos em processamento de sinais, seja identificação de características, remoção de ruído, ou compressão de dados. No entanto, as *wavelets* separáveis falham na representação de descontinuidades como os contornos da imagem, espalhando a energia para muitos coeficientes e não provendo representação esparsa. Neste trabalho foi descrita a teoria básica das *wavelets* e onde essas falham.

A metodologia *cartoon-textura* tem sido adotada como forma de preservar características distintas da imagem (respectivamente, a geometria e os detalhes), por meio de uma representação esparsa. Essa metodologia esteve presente nos trabalhos de Izquierdo [47], Aujol [48], Meyer [49] e Wakin [46][105] que foram usados como referência para este trabalho.

Utilizando o dicionário de *wedgelets* proposto inicialmente por Donoho [4] e tomando por base os trabalhos de Izquierdo [47] e Wakin et al [46][105], este trabalho propôs um novo método de compressão híbrido *wedgelet-wavelet* dentro da metodologia *cartoon-textura*.

As principais contribuições deste trabalho podem ser resumidas como:

- Melhoria do dicionário de *wedgelets* proposto por Donoho [4] com a otimização dos parâmetros e implementação de um algoritmo prático para a compressão de imagens.

- Modificação do algoritmo SPIHT de forma a otimizar uma função de custo taxa-distorção ao invés de usar taxa pré-fixada.
- Integração do dicionário *wedgelet* com a TWD de Haar por blocos em um codificador híbrido *wedgelet-wavelet* com desempenho superior a utilização das componentes isoladas, em termos de PSNR e qualidade subjetiva.

Este trabalho apresenta uma justificativa da metodologia *cartoon-textura* para a compressão de imagens em ambiente de baixas taxas de dados. Tal aplicação é motivada pela ineficiência matemática em conseguir representação esparsa usando transformadas ortogonais isotrópicas para sinais com descontinuidades. Os artigos de referência [54][94] corroboram essa conclusão.

8.2 Comentários sobre os Capítulos e Relação com a Proposta

Os capítulos iniciais da dissertação (capítulos 2 a 4) mostraram aspectos referentes ao contexto de aplicação do trabalho, especificamente transmissão digital móvel, qualidade visual e compressão de imagens. Os capítulos seguintes (capítulos 5 e 6) apresentaram as bases matemáticas dos métodos constituintes da proposta, a TWD e o dicionário de *wedgelets*. No capítulo 7 foi proposto um algoritmo *wedgelet-wavelet* com os melhores parâmetros encontrados e foram mostrados resultados de simulações que atestam o desempenho frente a outras implementações usando *wedgelets* e *wavelets*.

Neste capítulo finaliza-se o trabalho apresentando os comentários e conclusões a respeito dos resultados obtidos. Perspectivas de melhorias e encaminhamentos para trabalhos futuros são propostos.

8.3 Comentários sobre Simulações e Resultados

O método proposto foi implementado em Matlab 7.5.0 (R2007b) e os tempos de processamento de cada etapa foram medidos. Na etapa de identificação do *wedge* não foi utilizado nenhum método otimizado, focando apenas no desempenho taxa-distorção sem preocupação com o tempo de processamento. Por isso, a etapa mais complexa foi a geração inicial da árvore *wedgelet* da imagem, que varia com o tamanho do dicionário *wedgelet* utilizado. Como essa etapa é realizada uma única vez para cada imagem (independente da taxa de dados) e somente do lado do codificador, o método é mais indicado para aplicações de *broadcast*, em que o codificador pode ser muito mais complexo do que o decodificador.

O método proposto *wedgelet-wavelet* WWB apresentou vantagens em relação ao uso das componentes isoladas, pois se adapta de acordo com a taxa e com a geometria da imagem. Observa-se que para baixas taxas o codificador usa símbolos *wedgelet* no lugar de *wavelets* e conforme as taxas aumentam passa a usar mais componentes *wavelet* (Figura 7.11) e a curva se aproxima da curva da *wavelet* isolada (Figura 7.10). A distribuição de blocos *wavelet* e *wedgelet* varia com o

conteúdo da imagem, sendo mais usados *wedgelets* no caso de imagens de desenho (Figuras 7.13 a 7.17). Para imagens de desenho ainda, as *wedgelets* podem ser usadas mesmo em taxas mais altas.

Os resultados foram divididos entre imagens naturais e desenhos, obtendo resultados diferentes. Os valores médios de PSNR obtidos para imagens tipo “desenho” são naturalmente mais altos uma vez que as técnicas envolvidas foram originalmente desenvolvidas e dedicadas para tratar imagens desse tipo.

A característica principal apresentada pelo método proposto é a melhoria local de qualidade visual de acordo com a característica da imagem, *cartoon* ou textura. Essa característica justifica a metodologia *cartoon*-textura, que poderá ser enriquecida futuramente com novos métodos de codificação adicionados ao sistema híbrido, tratando características diversas da imagem, como foi sugerido por Coifman [108].

8.4 Sugestões para Trabalhos Futuros

A maior desvantagem do método apresentado ainda reside no alto tempo de processamento para o codificador. O peso computacional recai sobre o cálculo da decomposição *wedgelet* que ainda não foi otimizado e a escolha entre símbolo *wedgelet* e *wavelet*, que é feita de maneira quase exaustiva. Para a efetiva utilização desse método por sistemas de codificação práticos ainda é necessário um estudo que leve a uma arquitetura mais eficiente em calcular essa decomposição.

O método pode ter um desempenho ainda melhor do que apresentado neste trabalho substituindo o dicionário *wedgelet* de 1ª ordem por versões mais atualizadas, como as *arcllets* [102], *platelets*[103] e *surfllets*[104]. Espera-se um ganho subjetivo no contorno das imagens, mas um ganho efetivo em termos de taxa-distorção depende da forma como o dicionário será codificado, um detalhe que ainda não foi esclarecido pelos outros autores.

A metodologia *cartoon*-textura usando *wedgelets* e *wavelets* pode encontrar outras áreas de aplicação que este trabalho não mencionou. Qualquer processamento de imagens digitais que necessite recuperação de contornos e informação geométrica com poucos coeficientes (representação esparsa) pode em teoria ser beneficiada, seja para codificação, identificação de objetos ou pós-processamento e melhoria de qualidade.

Referencias Bibliográficas

- [1] ISO/IEC 15444 Information technology - JPEG 2000 image coding system, 2004.
- [2] ISO/IEC 14496-2 Information technology -- Coding of audio-visual objects -- Part 2: Visual, Motion Pictures Experts Group, 1999.
- [3] T.F. Chan, H.M. Zhou, Adaptive ENO-Wavelet Transforms for Discontinuous Functions, *12th International Conference on Domain Decomposition Methods*, 2001
- [4] D. L. Donoho, *Wedgelets: nearly minimax estimation of edges*, Annals of Stat., vol. 27, pp. 859–897, 1999.
- [5] C. Bhattacharya, P. R. Mahapatra, A Discrete Wavelet Transform Approach to Multiresolution Complex SAR Image Generation, IEEE Geoscience and Remote Sensing Letters, vol. PP, no. 9, pp. 416-420, 2007.
- [6] Sýkora, D., Buriánek, J., Žára, J. – Video Codec for Classical *Cartoon* Animations with Hardware Accelerated Playback, *Advances In Visual Computing : First International Symposium*, ISVC 2005
- [7] Base de dados IBGE (Instituto Brasileiro de Geografia e Estatística) março de 2007;
- [8] Site Oficial da TV Digital Brasileira - www.dtv.org.br
- [9] Portaria N° 24, de 11 de fevereiro de 2009, Norma Geral para Execução dos Serviços de Televisão Pública Digital do Ministério das Comunicações
- [10] ARIB STD-B31 ISDB-T Standard For Digital Terrestrial *Broadcasting*, Transmission System For Digital Terrestrial Television *Broadcasting*, Version 1.6, 2005
- [11] Redação Convergência Digital. Celulares: mercado mostra ligeira recuperação. Brasil chega a 153,67 milhões de assinantes, 23/04/2009
<http://www.convergenciadigital.com.br/cgi/cgilua.exe/sys/start.htm?from%5Finfo%5Findex=31&infoid=18571&sid=8>
- [12] Redação do IDG Now!, *Penetração de celular por habitante cresce 20,82% no Brasil, em um ano*, Publicada em 18 de junho de 2009 às 13h06, Atualizada em 18 de junho de 2009 às 14h03,
<http://idgnow.uol.com.br/telecom/2009/06/18/uso-de-celular-por-habitante-cresce-20-82-no-brasil-diz-anatel>;
- [13] Valor Econômico, "Pesquisa mostra perfil de usuário de celular", acessado em 06 de julho de 2009
<http://empresas.globo.com/Empresasenegocios/0,19125,ERA1702114-2884,00.html>

- [14] Feltes, S., O telespectador e a TV digital móvel entrevistas preliminares para um estudo de recepção, *Intercom – Sociedade Brasileira de Estudos Interdisciplinares da Comunicação, XXXII Congresso Brasileiro de Ciências da Comunicação*, Curitiba-PR 2009
- [15] Coelho, L. - *Produção local de conteúdo digital para uma sociedade mais justa*, Telecomonline, acessado em 06 de julho de 2009, <http://www.telecomonline.com.br/artigos/producao-local-de-conteudo-digital-para-uma-sociedade-mais-justa>;
- [16] Chan-Olmsted, S.M., Lee, S., Kim, H., Mobile television deployment strategy: A comparative analysis of mobile operators and television *broadcasters*, *Media Management and Economics Division at the annual meeting of the Association for Education in Journalism & Mass Communication*, Chicago, 2008
- [17] Prado, E. - Mobile TV ganha fôlego, IDGNow, 18/10/05, acessado em 06 de julho de 2009, <http://www.adnews.com.br/tecnologia.php?id=21674>;
- [18] ETSI EN 302 304: "Digital Video *Broadcasting* (DVB); Transmission System for Handheld Terminals (DVB-H), 2005
- [19] ETSI TS 102 428 V1.2.1 : Digital Audio *Broadcasting* (DAB); DMB video service; User application specification, 2009
- [20] Interfax China, "China releases mobile TV industrial standard". Press release, 2006
- [21] Mobile Media Management & Content Distribution Technology | MobiTV Official Site, 2011
<http://www.mobitv.com/>
- [22] Jornalismo 3G, *TV MÓVEL: Qualcomm traz MediaFlo para o Brasil*, quarta-feira, 20 de agosto de 2008
<http://jornalismo3g.blogspot.com/2008/08/tv-mvel-qualcomm-traz-mediaflo-para-o.html>
- [23] Qualcomm suspends Flo TV sales - Reuters, Oct 5 2010
- [24] Girod, B., Human Perception Topics,.: *EE368b Image and Video Compression* - Fall Quarter 2000/01, Stanford University, Department of Electrical Engineering, Information Systems Laboratory
- [25] W. K. Pratt – “Digital Image Processing” – 4a ed., PIKS Scientific Inside, Califórnia, 2007.
- [26] Santiago Ramón y Cajal, *Histologie Du Système Nerveux de l'Homme et Des Vertébrés*, Maloine, Paris, 1911
- [27] H.L. Resnikoff – “The Illusion of Reality” – Springer-Verlag, 1989.
- [28] Kalpana Seshadrinathan, Thrasyvoulos N. Pappas, Robert J. Safranek, Junqing Chen, Zhou Wang, Hamid R. Sheikh and Alan C. Bovik, *Image Quality Assessment, Essential Guide of Image Processing*, Elsevier, 2009
- [29] Durand, F., Dorsey, J., Introduction to Visual Perception, *The Art and Science of Depiction*, MIT- Lab

for Computer Science

- [30] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004
- [31] Jumisko-Pyykkö, S.H., Häkkinen, J., Effect of TV Content in Subjective Assessment of Video Quality on Mobile Devices, *Proceedings of the 13th annual ACM international conference on Multimedia*, 2005
- [32] Ries, M., Nemethova, O., Rupp, M., "Motion Based Reference-Free Quality Estimation for H.264/AVC Video Streaming," *Proc. of the ISWPC'07*, San Juan, PR, USA, Feb. 2007
- [33] Ries, M., Crespi, C., Nemethova, O., Rupp, M., "Content Based Video Quality Estimation for H.264/AVC Video Streaming," *Proc. of the IEEE WCNC'07*, Hong Kong, Mar. 2007
- [34] Ries, M., Nemethova, O., Performance Evaluation Of Mobile Video Quality Estimators, *Seminar on Signal and Image Processing*, Los Angeles, USA, 2008
- [35] Song, S., Won, Y., & Song, I. (2004). Empirical Study of User Perception Behavior for Mobile Streaming. In *Proceedings of the tenth ACM international conference on Multimedia* (pp. 327-330). New York, NY, USA: ACM Press
- [36] Knoche, H., McCarthy, J., & Sasse, M. A. (2005b). Can Small Be Beautiful? Assessing Image Resolution Requirements for Mobile TV. In *ACM Multimedia ACM*
- [37] Jumisko-Pyykkö, S., Häkkinen, J., "I would like see the face and at least hear the voice": Effects of Screen Size and Audio-video Bitrate Ratio on Perception of Quality in Mobile Television. In *G. Doukidis, K. Chorianopoulos, & G. Lekakos (Eds.), Proceedings of EuroITV '06* (pp. 339-348). Athens University of Economics and Business, 2006
- [38] C. E. Shannon: A mathematical theory of communication. *Bell System Technical Journal*, vol. 27, pp. 379-423 and 623-656, July and October, 1948
- [39] *Handbook of Image and Video Processing* – Editor Al Bovik, Ed. Academic Press, 2000
- [40] ISO/IEC 14496-10, Recommendation ITU-T H.264 Information technology - Coding of audio-visual objects - Part 10: Advanced video coding, 2006;
- [41] Gray, K. – *The JPEG 2000 Standard*, Technische Universität München Lehrstuhl für Kommunikationsnetze, 2002;
- [42] Arthur, R., Avaliação Objetiva de Codecs de Video, Dissertação de Mestrado, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, 2002
- [43] Silva, F. S., Procedimentos Para Medição E Minimização Do Efeito De Bloco Decorrente Do Processamento Digital De Imagens (PDI), Dissertação de Mestrado, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, 2001

- [44] Khryashchev, V., Ilya Apalkov, I., Shmaglit, L., A Novel Smart Bilateral Filter for *Ringling* Artifacts Removal in JPEG2000, *20th International Conference on Computer Graphics and Vision*, St.Petersburg, Russia, 2010
- [45] Führ, H., Demaret, L., Friedrich, F., Beyond *wavelets*: New image representation paradigms, *Document and Image Compression*, CRC Press, 2006
- [46] Wakin, M., Romberg, J., Choi, H., Baraniuk, R., Image Compression using an Efficient Edge *Cartoon* + Texture Model, *Proceedings Of The Data Compression Conference (DCC'02)*, 2002 IEEE Computer Science
- [47] Sprljan, N., Izquierdo, E., New Perspectives on Image Processing Using a *Cartoon* – Texture Decomposition Model, *4th EURASIP Conference focused on Video / Image Processing and Multimedia Communications*, 2-5 July 2003, Zagreb, Croatia
- [48] Aujol, J.F., Aubert, G., Feraud, L.B., Chambolle, A., Image Decomposition into a Bounded Variation Component and an Oscillating Component, *Journal of Mathematical Imaging and Vision* 22: 71-88, 2005, Springer Science + Business Media, Inc.
- [49] Meyer, Y., Oscillating patterns in image processing and in some nonlinear evolution equations, The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures, March 2001
- [50] Yan, J., Sakrison, D., Encoding of Images Based on a Two-Component Source Model, *Communications*, IEEE Transactions on, Nov 1977;
- [51] Carlsson, S., Sketch based coding of grey level images, *Signal Processing*, Elsevier North-Holland, Inc, Volume 15, Issue 1 (July 1988);
- [52] Perona, P., Malik J., Scale-space and edge detection using anisotropic diffusion, *Proceedings of IEEE Computer Society Workshop on Computer Vision*,. pp. 16–22., 1987
- [53] Perona, P., Malik J., Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12 (7): 629–639, 1990
- [54] Cohen, A., Daubechies, I., Guleryuz, O., Orchard, M., On the importance of combining non-linear approximation with coding strategies, *IEEE Trans. Inf. Theory*, 48 (7), pp. 1895--1921, 2002
- [55] Mallat, S., *A Wavelet Tour of Signal Processing*, Academic Press, 1999;
- [56] Peotta, L., Granai, L., Vandergheynst, P., Image compression using an edge adapted redundant dictionary and *wavelets*, *Signal Processing*, Elsevier North-Holland, Inc., Volume 86 , Issue 3 (March 2006);
- [57] Starck, J.L., Elad, M., Donoho, D.L., Image Decomposition Via the Combination of Sparse Representations and a Variational Approach, EDICS: 2-WAVP (*Wavelets and Multiresolution Processing*), *IEEE Transactions on Image Processing*, June 24, 2004
- [58] ISO/IEC 14496-20:2008 - Information technology -- Coding of audio-visual objects -- Part 20:

- Lightweight Application Scene Representation (LAsER) and Simple Aggregation Format (SAF).
- [59] Bhattacharya, P. R. Mahapatra, A Discrete *Wavelet* Transform Approach to Multiresolution Complex SAR Image Generation, *IEEE Geoscience and Remote Sensing Letters*, vol. PP, no. 9, pp. 416-420, 2007.
- [60] Using LizardTech *Wavelet* Compression with Oracle Spatial GeoRaster,
<http://www.oracle.com/technology/products/spatial>
- [61] Behroozmand, R., Almasganj F., Optimal selection of waveletpacket- based features using genetic algorithm in pathological assessment of patients speech signal with unilateral vocal fold paralysis, *Comput. Biol. Med.*, vol. 37, no. 4, pp. 474-485, Apr, 2007
- [62] Dettori, L., Semler L., A comparison of *wavelet*, ridgelet, and *curvelet*-based texture classification algorithms in computed tomography, *Comput. Biol. Med.*, vol. 37, no. 4, pp. 486-498, Apr, 2007
- [63] Akhbardeh, S. Junnila, M. Koivuluoma, T. Koivistoinen, V. Turjanmaa, T. Kööbi and A. Varri, Towards a heart disease diagnosing system based on force sensitive chair's measurement, biorthogonal *wavelets* and neural networks, *Eng. Appl. Artif. Intell.*, vol. 20, no. 4, pp. 493-502, Jun, 2007
- [64] J. Ng, R. Prager, N. Kingsbury, G. Treece, A. Gee, *Wavelet* restoration of medical pulse-echo ultrasound images in an EM framework, *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol.54, no. 3, pp. 550-568, Mar, 2007
- [65] R. S. S. Kumari, V. Sadasivam, A novel algorithm for *wavelet* based ECG signal coding, *ACM Computers and Electrical Engineering*, vol.33, pp. 186-194, May, 2007
- [66] X. Li, J. Li, and X. Yao, A *wavelet*-based data pre-processing analysis approach in mass spectrometry, *Comput. Biol. Med.*, vol. 37, n. 4, pp. 509-516, Apr, 2007
- [67] C. Aggarwal, and P. S. Bradley, On the Use of *Wavelet* Decomposition for String Classification, *Data Min. Knowl. Discov.*, vol. 10, no. 2, pp. 117-139, Mar, 2005
- [68] Castleman, K. R. Digital Image Processing, Ed. Prentice Hall, New Jersey, USA, 1996
- [69] 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
- [70] Chui C.F., An Introduction to *Wavelets*, Academic Press Inc., San Diego, USA, 1992
- [71] Daubechies, I., Ten Lectures on *Wavelets*, SIAM, Philadelphia, USA, 1992.
- [72] Farias, M. C. Q. Aplicação da Transformada *Wavelet* na Compressão de Imagens, 'Dissertação de Mestrado, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, 1998
- [73] Mallat, S., A Theory for Multiresolution Signal Decomposition: the *Wavelet* Representation, *IEEE*

- Transactions on PAMI*, vol. 11, pp. 674-693, 1989
- [74] Hanzo, L.L., Cherriman, P., Streit, J., *Wireless Video Communications: Second to Third Generation and Beyond*, *IEEE Series on Mobile & Digital Communications*, Wiley-IEEE Press; 1 Edition, 2001
- [75] Woods, J. W., O'Neill, S. D., Subband Coding of Images, *IEEE Transactions on ASSP*, vol. 34, pp. 1278-1288, 1986
- [76] Silva, A.L.M.C.S., Procedimentos Para Método Híbrido De Compressão De Imagens Digitais Utilizando Transformadas *Wavelet* E Codificação Fractal, Tese de Doutorado, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, 2005
- [77] Burt, P. J., Adelson, E. H., The Laplacian Pyramid as a Compact Image Code, *Proc. IEEE Int. Conf. Commun.*, 31(4):532-540, 1983
- [78] Mallat, S., Multifrequency Channel Decompositions of Images and *Wavelet* Models, *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no.12, Dez., 1989
- [79] Adelson, E. H., Simoncelli, E., Hingorani, R., Orthogonal Pyramid Transforms for Image Coding, *Proceedings of SPIE*, vol. 845, pp. 50-58, Out., 1987
- [80] Porat, B., *Digital Processing of Random Signals*, Ed. Prentice Hall, Pennsylvania, USA, 1992
- [81] Ansari, R., Guillemont, C., Exact reconstruction Filter Banks Using Diamond FIR Filters, *Proc. Bilkent Intl. Conf.*, pp. 1412-1424, July 1990
- [82] Kovacevic, J., Vetterli, M., Nonseparable Multidimensional Perfect reconstruction Filter Banks and *Wavelet* Bases for \mathbb{R}^n , *IEEE Trans. Info. Theory*, 38(2):533-555, 1992
- [83] Mallat, S., Multiresolution Approximation and *Wavelets*, *IEEE Transactions on PAMI*, vol. 11, no.07, Jul, 1989
- [84] Kazinnik, R., Image Compression using Piecewise Polynomials, *Doctor of Philosophy Thesis, School of Mathematical Sciences Tel Aviv University*, 2008
- [85] Wakin, M., Hyeokho Choi, Baraniuk, R., Romberg, J., *Wedgelets: A Multiscale Geometric Representation for Images*, *UCLA IPAM Tutorials on Multiscale Geometry and Analysis in High Dimensions*, Los Angeles, California, 2004
- [86] Mallat, S., Peyré, G., A review of Bandlet methods for geometrical image representation, *Numerical Algorithms*, Vol. 44(3), p. 205-234, 2007
- [87] Velisavljevic, V., Beferull-Lozano, B., Vetterli, M., Space-Frequency Quantization for Image Compression With *Directionlets*, *IEEE Transactions on Image Processing*, 2007
- [88] Nikol'ski, S., *Approximation of functions of several variables and imbedding theorems*, New York: Springer-Verlag, 1975
- [89] Neumann, M., von Sachs, R., *Wavelet* thresholding in anisotropic function classes and application to

- adaptive estimation of evolutionary spectra, *Ann. Statist.*, vol. 25, no. 1, pp. 38-76, 1997
- [90] Candès, E. J., Donoho, D. L., *Curvelets: A surprisingly effective nonadaptive representation for objects with edges*, *Curve and Surface fitting*, A. Cohen, C. Rabut and L. L. Schumaker (Eds.), 1999
- [91] Le Pennec, E., Mallat, S., *Bandelet Image Approximation and Compression*, *SIAM Multiscale Model. Simul.*, 4(3):992-1039, 2005
- [92] Mallat, S., Peyré, G., Orthogonal *bandelet*s for geometric image approximation, *Comm. Pure Appl. Math.*, 61(9):1173-1212, 2008
- [93] Le Pennec, E., Mallat, S., Sparse geometric image representations with *bandelet*s, *IEEE Trans. Image Process.*, 14(4):423-438, 2005
- [94] Baraniuk, R., H. Choi, Romberg, J. Wakin, M., *Multiscale Geometric Image Compression using Wavelets and Wedgelets*, Rice University, 2010
- [95] Radon, J.; Parks, P.C., On the determination of functions from their integral values along certain manifolds, *IEEE Transactions on Medical Imaging* 5 (4): 170–176, 1986
- [96] Hower, G. A., Kuo, W., Hanson, G., Sickman, F., Double Density Complex *Wavelet* Based Image *Cartoon-Texture* Decomposition, *Signal Processing for Image Enhancement and Multimedia Processing*, Springer US, 2008
- [97] Lisowska, A., Moments-Based Fast *Wedgelet* Transform, *Journal of Mathematical Imaging and Vision*, pp. 1-13, 2010
- [98] Friedrich, F., Demaret, L., Führ, H., Wicker, K., Efficient Moment Computation over Polygonal Domains with an Application to Rapid *Wedgelet* Approximation, *SIAM J. Sci. Comput.* Volume 29, Issue 2, pp. 842-863, 2007
- [99] Demaret, L., Führ, H., *Wedgelet* Segmentations and Approximation Classes, preprint, 2010
- [100] Lisowska, A., Extended *Wedgelets* — Geometrical *Wavelets* in Efficient Image Coding, *Machine Graphics & Vision International Journal*, 2004
- [101] Lisowska, A., Second Order *Wedgelets* in Image Coding, *EUROCON - The International Conference on Computer as a Tool*, 2007
- [102] Piyanut Pongpiyapaiboon, Development of Efficient Algorithm for Geometrical Representation based on *Arcllet* Decomposition, *Master's Thesis, Technische Universität München, Computational Science and Engineering*, 2005
- [103] Willett, R. M. , Nowak, R. D., *Platelets: a multiscale approach for recovering edges and surfaces in photon-limited medical imaging*, *IEEE Transactions on Medical Imaging*, 2003
- [104] Chandrasekaran, V., Wakin, M. B., Baron, D., Baraniuk, R. G., Representation and Compression of multidimensional piecewise functions using *surflets*, *IEEE Transactions on Information Theory*, 2009

- [105] Wakin, M.B., Romberg, J.K., Choi, H., Baraniuk, R.G., Rate-Distortion Optimized Image Compression Using *Wedgelets*, in *IEEE 2002 International Conference on Image Processing - ICIP*, Rochester, New York, September 2002
- [106] Xiong, Z., Ramchandran, K., Orchard, M. T., Space-frequency quantization for *wavelet* image coding, *IEEE Trans. Image Proc.*, vol 6, no 5, pp. 677-693, 1997
- [107] Romberg, J., Wakin, M., Hyeokho Choi, Baraniuk R., Multiscale Geometric Modeling of Edges in Images, RICE University, 2011
- [108] Coifman, R. R., Wickerhauser, M. V., Entropy Based Algorithms for Best Basis Selection, *IEEE Transactions on Information Theory*, 1992
- [109] Wicker, K., Das Potts Modell zur Segmentation verrauschter Daten, *Diploma Thesis*, TU. München, 2004
- [110] Wicker, K., Friedrich, F., Demaret, L., Führ, H., Poongpyiapaiboon, P., *Wedgelet* decomposition Oberon-based software,
<http://www.wedgelet.de>
- [111] The Waterloo Fractal and Coding and Analysis Group,
<http://links.uwaterloo.ca>
- [112] Said, A., Pearlman, W. A., A new fast and efficient image coding based on set partitioning in hierarchical trees, *IEEE Transactions on Circuits and Systems for Video Technology*, 1996
- [113] Lopes, A., Falar é o de menos, *Revista INFO, Editora Abril*, Março 2011
- [114] OpenJPEG, open-source JPEG 2000 codec,
<http://www.openjpeg.org>