

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



UMA COMBINAÇÃO ENTRE OS CRITÉRIOS OBJETIVO E
SUBJETIVO NA CLASSIFICAÇÃO DE IMAGENS
MAMOGRÁFICAS COMPRIMIDAS PELO MÉTODO FRACTAL

Stéphany Vergütz

UBERLÂNDIA

2013

STÉPHANY VERGÜTZ

UMA COMBINAÇÃO ENTRE OS CRITÉRIOS OBJETIVO E
SUBJETIVO NA CLASSIFICAÇÃO DE IMAGENS
MAMOGRÁFICAS COMPRIMIDAS PELO MÉTODO FRACTAL

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para a obtenção do título de mestre em Ciências.

Banca Examinadora:

Antônio Cláudio Paschoarelli Veiga, Dr. -
Orientador (UFU)

Edna Lúcia Flôres, Dra. (UFU)

Kátia Lopes Silva, Dra. (Faculdade Pitágoras
de Uberlândia)

UBERLÂNDIA

2013

- V498c Vergütz, Stéphany, 1983-
2013 Uma combinação entre os critérios objetivo e subjetivo na
classificação de imagens mamográficas comprimidas pelo método
fractal / Stéphany Vergütz. -2013.
150 p. : il.
Orientador: Antônio Cláudio Paschoarelli Veiga.
Dissertação (mestrado) – Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Elétrica.
Inclui bibliografia.
1. Engenharia elétrica - Teses. 2. Mamas – Radiografia - Teses. 3.
Compressão de imagens - Teses. 4. Fractais - Processamento de dados
- Teses. 5. Compressão de dados (Computação) – Teses. I. Veiga,
Antônio Cláudio Paschoarelli. II. Universidade Federal de Uberlândia.
Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

UMA COMBINAÇÃO ENTRE OS CRITÉRIOS OBJETIVO E
SUBJETIVO NA CLASSIFICAÇÃO DE IMAGENS
MAMOGRÁFICAS COMPRIMIDAS PELO MÉTODO FRACTAL

Stéphany Vergütz

Dissertação apresentada por Stéphany Vergütz à Universidade Federal de Uberlândia
como parte dos requisitos à obtenção do título de mestre em Engenharia Elétrica.

Prof. Dr. Antônio Cláudio Paschoarelli Veiga
Orientador

Prof. PhD. Edgard Afonso Lamounier Júnior
Coordenador do curso de Pós-Graduação

AGRADECIMENTOS

Agradeço primeiramente a Deus por estar sempre guiando meus passos e abençoando cada momento da minha vida e de toda minha família.

Agradeço aos meus pais, Ilmo e Neli, aos meus irmãos Leonardus e Thômas, e ao meu marido Murilo, que me apoiam em todos os momentos da minha vida, com palavras e gestos de amor e carinho.

Aos meus amigos pelos excelentes momentos que sempre compartilhamos. Tenho a certeza que torceram pela finalização deste trabalho.

À CAPES pelo apoio financeiro, ao professor Paschoarelli, meu orientador, pela oportunidade que tive e pelo seu apoio e incentivo.

À todos os professores que passaram por minha formação, pelo aprendizado constante.

Aos médicos especialistas, Dr. Flávio Rocha Gil, Dr. Raphael Alves Ferreira Tomé e Dra. Thaís Simone Procópio Silva Dixini, pela disponibilidade em analisar as imagens, o que enriqueceu muito este trabalho, em especial ao Dr. Edson Teixeira Fabrini que me auxiliou na procura dos médicos especialistas.

RESUMO

As imagens são fontes relevantes de informação em diversas áreas da ciência e tecnologia. O processamento dessas informações melhora e otimiza sua utilização. A compressão de imagens faz com que a representação da informação seja mais eficiente, reduzindo a quantidade de dados necessários para representar uma imagem. O objetivo deste trabalho é apresentar a avaliação do desempenho da compressão fractal aplicada a imagens mamográficas, pela combinação entre o critério objetivo, fornecido pela relação sinal ruído de pico (*Peak Signal Noise Ratio* - PSNR), e o critério subjetivo, especificado pela análise visual de médicos especialistas. A análise visual foi realizada comparando as imagens mamográficas comprimidas com diferentes taxas de compressão e a imagem original. Os especialistas classificaram as imagens comprimidas como "inaceitável", "aceitável", "boa" ou "ótima". Dessa maneira, conseguiu-se combinar a taxa de compressão e o valor de PSNR, para que as imagens comprimidas sejam consideradas aceitáveis pelos especialistas. Para avaliar o desempenho da compressão fractal foram realizados testes e análises visuais com as mesmas imagens utilizando o método de compressão JPEG2000.

Palavras chave: Compressão fractal de imagens, Mamografia, Taxa de compressão, Relação sinal ruído de pico.

ABSTRACT

Images are relevant sources of information in many areas of science and technology. The processing of such information improves and optimizes its use. The image compression causes the information representation is more efficient, reducing the amount of data required to represent an image. The objective of this study is to evaluate the performance of “Fractal Compression” technique onto mammograms through an association between the objective criteria, provided by Peak Signal Noise Ration (PSNR); and the subjective criteria, given by visual analysis of an expert physician. Visual analysis was performed comparing mammograms compressed to different extents (compression rate) with the original image, where experts classified the compressed images as “unacceptable”, “acceptable”, “good” or “great”. In doing so, the optimal compression rate and PSNR values of mammograms was achieved, where images are considered acceptable according to experts. In order to compare the performance of fractal compression technique with another compression method, visual analysis was also done on images compressed by JPEG2000 method.

Keywords: Fractal image compression, Mammography, Compression rate, Peak signal noise ratio.

LISTA DE ILUSTRAÇÕES

Figura 2.1 – Variação do número de amostras e do número de níveis de quantização (a) 256 x 256/256 níveis; (b) 128 x 128/256 níveis; (c) 32 x 32/256 níveis e (d) 256 x 256/2 níveis. ...	22
Figura 2.2 – Esquema de compressão e descompressão de uma imagem.	24
Figura 2.3 – Técnicas de compressão de imagens.	25
Figura 3.1 – Exemplos de fractais encontrados na natureza. Brócolis, caramujo, estalactites, copa de árvore, nuvem, semente de girassol, galáxia, samambaia, e flor.	31
Figura 3.2 – Exemplos de estruturas matemáticas fractais: triângulo de <i>Sierpinski</i> , conjunto de <i>Cantor</i> , curva de <i>Koch</i> e conjunto de <i>Mandelbrot</i> [6].	32
Figura 3.3 – Distância entre um ponto e um conjunto e distância entre dois conjuntos.	33
Figura 3.4 – 1ª iteração da máquina fotocopadora [3].	36
Figura 3.5 – Primeiras iterações da máquina fotocopadora especial para várias imagens de entrada [3].	37
Figura 3.6 – Conjunto de Cantor [5].	40
Figura 3.7 – Curva de Koch [5].	41
Figura 3.8 – Processo de construção do triângulo de <i>Sierpinsky</i> [5].	42
Figura 4.1 – Similaridades dentro da própria imagem [3].	45
Figura 4.2 – <i>Range block</i> e <i>domain block</i> [13].	46
Figura 4.3 – 8 transformações (4 rotações e 4 inversões) [12].	47
Figura 4.4 – Esquema de compressão fractal pelo método da força bruta.	50
Figura 4.5 – Imagem inicial e as 5 primeiras iterações do processo de descompressão de método força bruta.	57
Figura 4.6 – Imagem inicial e as 5 primeiras iterações do processo de descompressão do método força bruta.	58
Figura 4.7 – Representação em árvore da partição Quadtree [3].	60
Figura 4.8 – Esquema de compressão fractal pelo método quadtree.	62
Figura 4.9 – Numeração dos blocos para classificação [3].	66
Figura 4.10 – Superclasses 1, 2 e 3 [3].	67
Figura 4.11 – 8 transformações que são classificadas dentro de cada superclasse.	68
Figura 4.12 – Imagem inicial e as 5 primeiras iterações do processo de descompressão.	75

Figura 4.13 – Imagem inicial e as 5 primeiras iterações do processo de descompressão.....	76
Figura 5.1 – Onda seno (análise fourier) e onda <i>wavelet</i> (análise <i>wavelet</i>) [20].....	79
Figura 5.2 – Comparação da primeira parte do sinal com a <i>wavelet</i> [20].....	80
Figura 5.3 – Deslocamento da <i>wavelet</i> e comparação com a próxima parte do sinal [20].	81
Figura 5.4 – Expansão da <i>wavelet</i> [20].	81
Figura 5.5 – Diferentes escalas da <i>wavelet</i> [20].	82
Figura 5.6 – Coeficientes gerados pela passagem de <i>wavelets</i> por um sinal qualquer [20].....	82
Figura 5.7 – Conjunto de filtros, passa-baixa e passa-alta [20].....	83
Figura 5.8 – Aplicação do conjunto de filtros em um nível de transformação [22].	84
Figura 5.9 – 1°, 2° e 3° níveis de transformação [21].....	85
Figura 5.10 – Imagem inicial e 5 taxas de compressão JPEG2000.	87
Figura 6.1 – Vistas MLO e CC [27].	90
Figura 6.2 – Caso 1 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].	92
Figura 6.3 – Caso 2 (a) Mama Direita – vista MLO; (b) Mama Direita – vista; CC (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].	93
Figura 6.4 – Caso 3 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].	94
Figura 6.5 – Caso 4 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].	95
Figura 6.6 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 2,09 bpp; (c) 1,39 bpp; (d) 0,75 bpp; (e) 0,51 bpp; (f) 0,39 bpp.	98
Figura 6.7 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,09 bpp; (c) 1,19 bpp; (d) 0,7 bpp; (e) 0,48 bpp; (f) 0,31 bpp.....	99
Figura 6.8 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 2,17 bpp; (c) 1,51 bpp; (d) 0,87 bpp; (e) 0,56 bpp; (f) 0,43 bpp.	100
Figura 6.9 - Compressão of Fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 2,19 bpp; (c) 1,34 bpp; (d) 0,78 bpp; (e) 0,51 bpp; (f) 0,36 bpp.	101
Figura 6.10 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,41 bpp; (c) 0,41 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	102
Figura 6.11 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,43 bpp; (c) 0,43 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	103

Figura 6.12 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,43 bpp; (c) 0,43 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.....	104
Figura 6.13 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	105
Figura 6.14 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,9 bpp; (c) 1 bpp; (d) 0,61 bpp; (e) 0,43 bpp; (f) 0,29 bpp.	106
Figura 6.15 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,39 bpp; (c) 1,26 bpp; (d) 0,73 bpp; (e) 0,51 bpp; (f) 0,34 bpp.....	107
Figura 6.16 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,78 bpp; (c) 0,97 bpp; (d) 0,56 bpp; (e) 0,39 bpp; (f) 0,29 bpp.	108
Figura 6.17 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 1,97 bpp; (c) 1 bpp; (d) 0,63 bpp; (e) 0,41 bpp; (f) 0,29 bpp.	109
Figura 6.18 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	110
Figura 6.19 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,51 bpp; (c) 0,51 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	111
Figura 6.20 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.....	112
Figura 6.21 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	113
Figura 6.22 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,83 bpp; (c) 0,87 bpp; (d) 0,58 bpp; (e) 0,31 bpp; (f) 0,21 bpp.	114
Figura 6.23 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 1,92 bpp; (c) 0,9 bpp; (d) 0,58 bpp; (e) 0,31 bpp; (f) 0,21 bpp.....	115
Figura 6.24 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,78 bpp; (c) 0,83 bpp; (d) 0,58 bpp; (e) 0,29 bpp; (f) 0,21 bpp.	116
Figura 6.25 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 1,75 bpp; (c) 0,85 bpp; (d) 0,61 bpp; (e) 0,36 bpp; (f) 0,26 bpp.	117
Figura 6.26 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	118
Figura 6.27 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	119

Figura 6.28 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.....	120
Figura 6.29 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	121
Figura 6.30 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,8 bpp; (c) 1,48 bpp; (d) 1 bpp; (e) 0,53 bpp; (f) 0,43 bpp.	122
Figura 6.31 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,22 bpp; (c) 1,63 bpp; (d) 1,02 bpp; (e) 0,56 bpp; (f) 0,48 bpp.....	123
Figura 6.32 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,8 bpp; (c) 1,36 bpp; (d) 0,87 bpp; (e) 0,46 bpp; (f) 0,36 bpp.	124
Figura 6.33 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 2,39 bpp; (c) 1,53 bpp; (d) 1 bpp; (e) 0,56 bpp; (f) 0,41 bpp.	125
Figura 6.34 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	126
Figura 6.35 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,53 bpp; (c) 0,53 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	127
Figura 6.36 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,41 bpp; (c) 0,41 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.....	128
Figura 6.37 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,51 bpp; (c) 0,51 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.	129

LISTA DE TABELAS

Tabela 2.1 – Quantidade de bits de armazenamento para vários valores de N e k .	23
Tabela 4.1 – Valores de PSNR variando a sobreposição e o tamanho do <i>range block</i> .	54
Tabela 4.2 – Valores de PSNR variando a quantidade de bits para representar o contraste (<i>sbits</i>) e o brilho (<i>obits</i>).	55
Tabela 4.3 – Valores de PSNR variando o valor máximo para o contraste.	56
Tabela 4.4 – Valores de PSNR variando o limiar de erro.	70
Tabela 4.5 – Valores de PSNR variando a quantidade de bits para representar o contraste (<i>sbits</i>) e o brilho (<i>obits</i>).	71
Tabela 4.6 – Valores de PSNR variando o valor máximo do contraste (<i>smax</i>).	72
Tabela 4.7 – Tamanho dos <i>range blocks</i> para cada nível de partição.	73
Tabela 4.8 – Valores de PSNR variando o nível máximo de partição.	74
Tabela 6.1 – Valor em bpp das taxas de compressão.	131
Tabela 6.2 – Classificações do médico 1 (compressão fractal).	131
Tabela 6.3 – Classificações do médico 1 (compressão JPEG2000).	132
Tabela 6.4 – Classificações do médico 2 (compressão fractal).	133
Tabela 6.5 – Classificações do médico 2 (compressão JPEG2000).	134
Tabela 6.6 – Classificações do médico 3 (compressão fractal).	135
Tabela 6.7 – Classificações do médico 3 (compressão JPEG2000).	136
Tabela 6.8 – Resumo da análise dos 3 médicos quando as imagens foram comprimidas utilizando a compressão fractal.	137
Tabela 6.9 – Resumo da análise dos 3 médicos quando as imagens foram comprimidas utilizando a compressão JPEG2000.	137
Tabela 6.10 – Média da taxa de bits do método fractal.	138
Tabela 6.11 – Média do valor de PSNR do método fractal.	139
Tabela 6.12 – Média da taxa de bits do método JPEG2000.	139
Tabela 6.13 – Média do valor de PSNR do método JPEG2000.	140
Tabela 6.14 – Limite da taxa de bits e valor da PSNR para imagens mamográficas comprimidas utilizando os métodos fractal e JPEG2000.	140

LISTA DE ABREVIATURAS

bpp	Bits por pixel
CC	Cranio-caudal
CV	<i>Cleavage</i>
CWT	<i>Continuous Wavelet Transform</i>
dB	Decibel
DSPG	<i>Digital Signal Processing Group</i>
DCT	<i>Discrete Cosine Transform</i>
DFT	<i>Discrete Fourier Transform</i>
DWT	<i>Discrete Wavelet Transform</i>
IFS	<i>Iterated Function System</i>
Inatel	Instituto Nacional de Telecomunicações
ISO	<i>International Organization for Standardization</i>
ITU	<i>International Telecommunication Union</i>
JPEG	<i>Joint Photographic Experts Group</i>
LM	Lateromedial
ML	Médio-lateral
MLO	Médiolateral oblique
P	Perfil externo
PDI	Processamento Digital de Imagens

PIFS	<i>Partitioned Iterated Function System</i>
PSNR	<i>Peak Signal Noise Ratio</i>
RCC	Caudocranial
RLE	<i>Run Length Encoding</i>
RMS	<i>Root Mean Square</i>
XCC	<i>Exaggerated crânio-caudal</i>

SUMÁRIO

1	CAPÍTULO 1	16
1.1	Apresentação	16
1.2	Objetivos	17
1.3	Estrutura da Dissertação.....	18
1.4	Considerações Finais do Capítulo	18
2	CAPÍTULO 2	20
2.1	Introdução.....	20
2.2	Representação de Imagens Digitais	21
2.3	Compressão de Imagens Digitais.....	23
2.4	Técnicas de Compressão de Imagens sem Perdas	25
2.4.1	Codificação de Huffman	26
2.4.2	Codificação de Lempel-Ziv.....	26
2.4.3	Codificação por sequência repetitiva	26
2.4.4	Codificação por plano de bit	26
2.5	Técnicas de Compressão de Imagens com Perdas.....	27
2.5.1	Codificação por truncagem de blocos.....	27
2.5.2	Codificação por transformada	27
2.6	Medidas de Desempenho	28
2.7	Considerações Finais do Capítulo	29
3	CAPÍTULO 3	30
3.1	Introdução.....	30
3.2	Definição de Fractais	30
3.3	Base Matemática da Geometria Fractal	32

3.3.1	Sequência de <i>Cauchy</i> e espaços métricos completos.....	34
3.3.2	Métrica e espaço métrico de <i>Hausdorff</i>	35
3.4	Geração de Fractais.....	36
3.5	Sistema de Funções Iterativas (IFS)	38
3.5.1	Fractais Gerados por IFS.....	40
3.6	Considerações Finais do Capítulo	43
4	CAPÍTULO 4	44
4.1	Introdução.....	44
4.2	Método da Força Bruta	45
4.2.1	Testes e resultados utilizando o método da força bruta	53
4.3	Método da Partição Quadtree	59
4.3.1	Classificação de Média e Variância.....	65
4.3.2	Testes e resultados utilizando o método da partição quadtree	69
4.4	Considerações Finais do Capítulo	77
5	CAPÍTULO 5	78
5.1	Introdução.....	78
5.2	Transformada <i>Wavelet</i>	79
5.3	Padrão de Compressão JPEG2000.....	83
5.3.1	Resultados da Compressão JPEG2000	85
5.4	Considerações Finais do Capítulo	88
6	CAPÍTULO 6	89
6.1	Introdução.....	89
6.2	Imagens mamográficas	90
6.2.1	Caso 1.....	98
6.2.2	Caso 2.....	106
6.2.3	Caso 3.....	114

6.2.4	Caso 4.....	122
6.3	Análises dos Médicos Especialistas.....	130
6.4	Considerações Finais do Capítulo	141
7	CAPÍTULO 7	142
7.1	Introdução.....	142
7.2	Conclusões	142
7.3	Contribuições deste Trabalho	145
7.4	Propostas para Trabalhos Futuros.....	146
8	REFERÊNCIAS BIBLIOGRÁFICAS.....	147
9	APÊNDICE	150

CAPÍTULO 1

INTRODUÇÃO

1.1 Apresentação

Como diz o ditado popular “uma imagem vale mais do que mil palavras”. As imagens constituem fontes importantes de informações e são utilizadas em diversas aplicações. O Processamento Digital de Imagens (PDI) vem apresentando um expressivo crescimento e vem sendo cada vez mais utilizado em vários ramos da ciência e tecnologia. Suas aplicações consistem em analisar e tratar as informações retiradas de uma imagem melhorando a informação visual e auxiliando a interpretação humana. Utilizando diversos conceitos e metodologias, o PDI trata as imagens digitais conforme a necessidade de cada área.

A evolução e a popularização de novas tecnologias fazem com que o número de informações geradas seja cada vez maior. O fluxo, o armazenamento e o gerenciamento de uma quantidade enorme de informação pode ser um grande problema. Para evitar o congestionamento no processo de transmissão ou mesmo otimizar o armazenamento dessa grande quantidade de dados é preciso fazer com que a representação da informação seja da forma mais eficiente possível.

A representação eficiente dos dados é realizada pela eliminação ou pela minimização da redundância existente na informação. Essa eliminação ou minimização é realizada pela compressão. A gama de imagens utilizadas em diversas aplicações é muito grande e o espaço em disco que essa enorme quantidade de dados requer é muito grande também. É por esse motivo que a Compressão de Imagens é um tema muito analisado atualmente, pois ele

diminui a quantidade de dados para representar cada imagem. Existem diversos métodos que reduzem o espaço necessário para o armazenamento de imagens, ou seja, diminuem o tamanho das mesmas. Cada método utiliza uma técnica específica e o método abordado nesta dissertação é baseado na Teoria dos Fractais.

A principal característica dos fractais é a auto-similaridade global existente neles, ou seja, cada pedaço de um fractal é igual a ele por inteiro. Em imagens reais não existe a auto-similaridade global, mas existem pedaços dentro da própria imagem que podem ser iguais ou parecidos entre si. A Compressão Fractal de Imagens utiliza essa característica das imagens para gerar os coeficientes da Compressão Fractal.

Este capítulo apresenta os objetivos e a estrutura desta dissertação. Finalmente, são realizadas as considerações finais deste capítulo.

1.2 Objetivos

O objetivo geral desta dissertação é o estudo da Compressão Fractal de Imagens, abordando os dois métodos propostos por *Fischer*, Força Bruta e Partição Quadtree. Tem-se como objetivos específicos:

- O estudo dos efeitos que os parâmetros de entrada utilizados na etapa de compressão têm na qualidade da imagem reconstruída;
- A avaliação do desempenho da Compressão Fractal aplicada a imagens mamográficas;
- A comparação dos resultados com o método de compressão JPEG2000 também aplicado a imagens mamográficas;
- A análise das imagens mamográficas comprimidas pelos médicos especialistas e a combinação dos critérios objetivo e subjetivo com a finalidade de definição de um limite da taxa de bits para as imagens comprimidas serem consideradas aceitáveis pelos especialistas.

1.3 Estrutura da Dissertação

A dissertação está dividida em 7 capítulos. O primeiro capítulo faz uma breve introdução, apresentando os objetivos e a estrutura desta dissertação.

O Capítulo 2 faz uma abordagem dos fundamentos da compressão de imagens digitais, descrevendo alguns métodos de compressão e mostra os métodos de comparação entre a imagem original e a imagem reconstruída.

O Capítulo 3 introduz o conceito de Fractal e aborda os conceitos de Teoria Fractal na qual a Compressão Fractal de imagens é baseada.

No Capítulo 4 inicia-se o estudo sobre Compressão Fractal onde são abordados dois métodos conhecidos como Método da Força Bruta e Método da Partição Quadtree, ambos baseados nos estudos de Fischer. Para ambos os métodos serão realizados o estudo dos efeitos que os parâmetros de entrada, utilizados na etapa de compressão, têm na qualidade da imagem reconstruída.

O Capítulo 5 descreve a Compressão JPEG2000, baseada na Transformada *Wavelet*. Essa introdução sobre essa compressão é necessária, pois a mesma é comparada com a Compressão Fractal.

O Capítulo 6 mostra uma breve introdução sobre imagens mamográficas. Algumas imagens mamográficas retiradas de um banco de imagens serão utilizadas para testes de compressão utilizando a Compressão Fractal e a Compressão JPEG2000. Esses testes serão apresentados nesse capítulo, assim como os resultados obtidos. O Capítulo 7 apresenta as conclusões e contribuições gerais deste trabalho e as sugestões para trabalhos futuros.

1.4 Considerações Finais do Capítulo

Este capítulo apresentou uma breve introdução sobre o PDI e suas diversas aplicações, e também uma introdução sobre o objetivo desta dissertação que é a compressão de imagens. Foi abordado superficialmente a Teoria dos Fractais, pois a Compressão Fractal de Imagens é a base desta dissertação.

O próximo capítulo primeiramente mostra a representação de imagens digitais e posteriormente apresenta um apanhado geral sobre compressão de imagens e algumas técnicas de compressão de imagens.

CAPÍTULO 2

FUNDAMENTOS DA COMPRESSÃO DE IMAGENS DIGITAIS

2.1 Introdução

As imagens digitais são representadas por pixels. Cada um dos pixels de uma imagem representa o nível de cinza da imagem naquele determinado ponto. A qualidade da imagem depende do número de pixels que a representa e também do número de níveis de cinza existentes para representar cada um dos pixels. Quanto maior é o número de pixels de uma imagem, e quanto mais níveis de cinza forem utilizados, melhor é a qualidade da imagem e também maior é o tamanho do arquivo (taxa de bits).

Imagens com boa qualidade geram arquivos bastante grandes, o que inviabiliza a transmissão ou o armazenamento dessas imagens. A finalidade da Compressão de Imagens é reduzir o número de bits necessários para representar uma imagem digital. Existem diversas técnicas utilizadas atualmente com o intuito de comprimir imagens. Algumas técnicas conseguem reconstruir o arquivo idêntico ao original, são as chamadas técnicas de compressão sem perdas, outras técnicas reconstroem uma imagem aproximada da original, que são as técnicas de compressão com perdas.

2.2 Representação de Imagens Digitais

Uma imagem é uma função bidimensional de intensidade da luz $f(x, y)$, onde x e y são coordenadas espaciais. O valor da função em qualquer ponto da imagem representa o brilho da imagem naquele ponto [1].

Quando as coordenadas espaciais e a amplitude (brilho) são discretizadas (amostradas e quantizadas) tem-se uma imagem digital. A amostragem das coordenadas espaciais é realizada pela divisão da imagem em uma malha de tamanho $n \times m$ (n linhas e m colunas), e cada um desses pequenos blocos formados pela intersecção das linhas e colunas são chamados “*pixels*”. A quantização da amplitude de cada pixel da imagem é realizada atribuindo um valor de nível de cinza a cada par de coordenadas, ou seja, a cada *pixel* [1].

Uma imagem digital é uma matriz de pixels e o valor em cada um desses pixels representa o nível de cinza de cada pixel. A qualidade da imagem digital depende tanto do número de pixels que vão representar essa imagem, quanto da quantidade disponível de níveis de cinza que pode representar cada pixel. Quanto mais pixels são utilizados para representar a imagem e quanto maior é a quantidade de níveis de cinza que são disponibilizados para representar cada pixel, melhor é a qualidade da imagem digital.

Considerando uma imagem em níveis de cinza (sem cor, apenas com variações entre o preto e o branco), a mesma pode ser representada por diferentes quantidades de níveis de cinza. Se uma imagem digital é representada por L níveis de cinza, isso significa que o intervalo de variação é $[0, L - 1]$, onde o nível 0 representa a cor preta, o nível $L - 1$ representa a cor branca e todos os níveis intermediários são as variações de tons de cinza entre a cor preta e a cor branca.

O número de níveis de cinza é uma potência inteira de 2 da forma: $L = 2^k$, onde k é o número de bits necessários para se ter L possíveis níveis de cinza (níveis de quantização). A Figura 2.1 mostra como a variação do número de amostras (índices n e m), e a variação do número de níveis de quantização (índice L), afetam a qualidade da imagem digital. A primeira imagem é 256×256 e com 256 níveis de cinza, a segunda imagem é 128×128 e com 256 níveis de cinza, a terceira imagem é 32×32 e com 256 níveis de cinza e a quarta imagem é 256×256 e com 2 níveis de cinza. Comparando as 4 imagens, pode-se observar que quanto

maior é a quantidade de pixels e quanto maior é a quantidade de níveis de cinza utilizados, melhor a qualidade da imagem.



Figura 2.1 – Variação do número de amostras e do número de níveis de quantização (a) 256 x 256/256 níveis; (b) 128 x 128/256 níveis; (c) 32 x 32/256 níveis e (d) 256 x 256/2 níveis.

A quantidade de bits necessários para armazenar uma imagem digital depende da quantidade de pixels que foram utilizados na amostragem (índices n e m), e da quantidade de níveis de cinza utilizados na etapa de quantização (índice L , $L = 2^k$). O número de bits b necessários para armazenar uma imagem digital de tamanho $m \times n$, utilizando k bits para a quantização dos níveis de cinza é obtido pela Equação (2.1).

$$b = n * m * k \quad (2.1)$$

Por exemplo, para que cada pixel de uma imagem possa ser representado por 256 níveis de cinza, ou seja, entre o preto e o branco existem 254 níveis de cinza, onde o nível 0 representa a cor preta e o nível 255 a cor branca, são necessários 8 bits para cada pixel da imagem ($2^8 = 256$). Considerando uma imagem de tamanho 64×64 , representada por 256 níveis de cinza possíveis para cada pixel (níveis de quantização), a mesma gasta, em termos de espaço de armazenamento, aproximadamente 32.768 bits ($64 * 64 = 4.096 \text{ pixels} * 8 \text{ bpp} = 32.768 \text{ bits}$). Considerando que 1 byte (B) é equivalente a 8 bits, essa imagem gasta em torno de 4 kB para ser armazenada [1].

O aumento de n , m e k , implica em um aumento da qualidade da imagem digital, embora implique também em um aumento do número de bits necessários para a codificação da imagem, aumentando assim o volume de dados a serem armazenados, processados ou

transmitidos. A Tabela 2.1 mostra o aumento da quantidade de bits para armazenamento considerando imagens quadradas $n \times m$, em função do aumento de N e k , onde ($N = n = m$).

Tabela 2.1 – Quantidade de bits de armazenamento para vários valores de N e k .

$N \setminus k$	1 ($L = 2$)	2 ($L = 4$)	3 ($L = 8$)	4 ($L = 16$)	5 ($L = 32$)	6 ($L = 64$)	7 ($L = 128$)	8 ($L = 256$)
32	1.024	2.048	3.072	4.096	5.120	6.144	7.168	8.192
64	4.096	8.192	12.288	16.384	20.480	24.576	28.672	32.768
128	16.384	32.768	49.152	65.536	81.920	98.304	114.688	131.072
256	65.536	131.072	196.608	262.144	327.680	393.216	458.752	524.288
512	262.144	524.288	786.432	1.048.576	1.310.720	1.572.864	1.835.008	2.097.152
1024	1.048.576	2.097.152	3.145.728	4.194.304	5.242.880	6.291.456	7.340.032	8.388.608

O espaço em disco que uma imagem de boa qualidade requer é bastante alto. Levando em consideração que vários ramos da ciência utilizam as imagens como ferramenta, o estudo de técnicas de compressão de imagens tem sido um assunto constante de pesquisas.

2.3 Compressão de Imagens Digitais

A quantidade de dados utilizados para representar imagens digitais pode ser muito grande, inviabilizando assim o armazenamento ou a transmissão da mesma. Uma mesma informação pode ser representada utilizando diferentes quantidades de dados. Quando essa informação é representada usando uma quantidade maior de dados do que o necessário, pode-se considerar que existe redundância de dados na representação dessa imagem.

A compressão de imagens é um conjunto de técnicas para reduzir o número de dados (taxa de bits) necessários para representar uma imagem digital. Essa redução da taxa de bits é realizada principalmente eliminando ou reduzindo a redundância existente nas imagens. As técnicas que fazem essa redução ou eliminação de redundância podem preservar a qualidade da imagem integralmente, reconstruindo uma cópia idêntica à original, ou reconstruindo a imagem o mais próximo possível da original, mas perdendo alguns detalhes ou informações da imagem original. A Figura 2.2 mostra o esquema de compressão de uma imagem, com a finalidade que ela seja armazenada ou transmitida e então descomprimida [1].

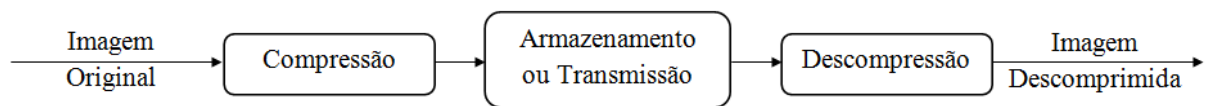


Figura 2.2 – Esquema de compressão e descompressão de uma imagem.

Normalmente, uma imagem digital é representada por uma quantidade de dados maior do que o necessário, e esses dados “extras” são chamados de dados redundantes. A redundância de dados em imagens digitais pode ser identificada e explorada com a finalidade de realizar a compressão dessa imagem. A redundância de dados dentro de uma imagem pode ser classificada em três tipos: redundância de codificação, redundância interpixels e redundância psicovisual. A compressão é alcançada quando se consegue eliminar ou reduzir uma ou mais dessas redundâncias [1].

A redundância de codificação é a utilização de mais símbolos do que o necessário para representar certos níveis de cinza, não observando a probabilidade de ocorrência de cada nível de cinza. Se a probabilidade de ocorrência de cada nível de cinza dentro da imagem é levada em conta, pode-se fazer com que os níveis de cinza com maior probabilidade de ocorrência sejam representados por códigos menores (menos bits), e níveis de cinza com menor probabilidade de ocorrência sejam representados por códigos maiores, eliminando (ou reduzindo) assim a chamada redundância de codificação [1].

A redundância interpixel são as relações geométricas ou estruturais existentes entre os objetos de uma imagem, mais especificamente às correlações interpixels dentro de uma imagem. Lembrando que o coeficiente de correlação em uma imagem avalia o grau de relacionamento entre dois pixels. A redução desse tipo de redundância está relacionada ao fato de que o nível de cinza de um determinado pixel pode ser previsto a partir dos valores dos pixels vizinhos, podendo-se utilizar, por exemplo, as diferenças entre os pixels adjacentes para representar uma imagem, ao invés da tradicional matriz bidimensional de pixels [1].

A redundância psicovisual é a percepção visual do ser humano. O sistema visual humano tem uma complexidade muito grande e a redução da redundância psicovisual é realizada pela eliminação de informações com baixa importância visual [1].

As técnicas de compressão de imagens podem ser divididas em dois tipos fundamentais que são determinados pelos níveis de perdas do arquivo original em relação ao arquivo descomprimido posteriormente. Os dois tipos fundamentais em que as técnicas de compressão são divididas são a compressão sem perdas, também conhecida como *lossless* e a compressão com perdas, conhecida como *lossy*. A Figura 2.3 mostra algumas técnicas de compressão com perdas e sem perdas.

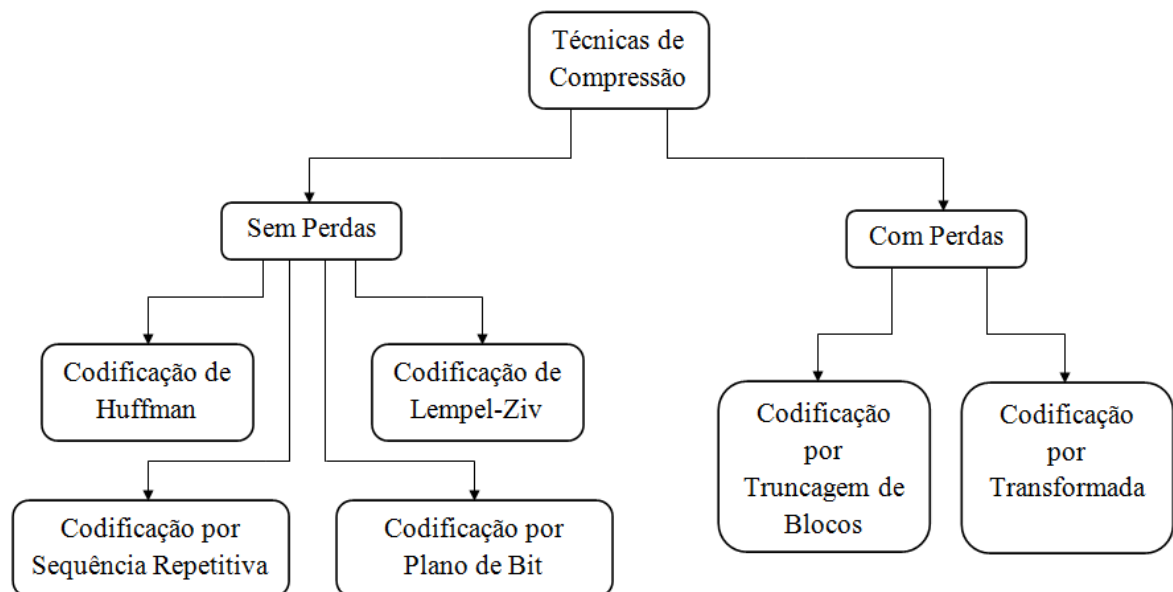


Figura 2.3 – Técnicas de compressão de imagens.

Os subitens 2.4 e 2.5 mostram uma pequena abordagem sobre cada uma dessas técnicas de compressão de imagens.

2.4 Técnicas de Compressão de Imagens sem Perdas

A compressão sem perdas, como o próprio nome diz, resulta em um arquivo idêntico ao original sem a perda de nenhuma informação. Esse tipo de compressão é realizado a partir da codificação dos dados de entrada com a finalidade de que os dados codificados resultantes ocupem menos espaço que os dados originais. Uma desvantagem dos métodos de compressão sem perdas é que eles não conseguem atingir altas taxas de compressão, pois nenhuma informação é perdida nesse tipo de compressão, os dados de entrada são apenas codificados.

2.4.1 Codificação de Huffman

A codificação de Huffman foi desenvolvida em 1951 por Davis A. Huffman. Ela é a técnica mais popular de codificação para redução de redundância. A técnica de Huffman codifica um conjunto de dados baseado na probabilidade de ocorrência, de forma que os símbolos com uma frequência de ocorrência maior são codificados com um menor número de bits. No caso de imagens utiliza-se a probabilidade de ocorrência de cada nível de cinza para então codificar aqueles com maior probabilidade de ocorrência com uma quantidade menor de bits e vice-versa [1, 2].

2.4.2 Codificação de Lempel-Ziv

A codificação de Lempel-Ziv foi desenvolvida em 1977 por Jacob Ziv e Abraham Lempel. Essa codificação ficou conhecida como LZ-1 ou LZ-77. Ela armazena em uma lista (ou dicionário) as sequências de símbolos que ocorrem com maior frequência. Cada sequência de símbolos, referente ao arquivo original, é representada na lista apenas pelo seu índice, que indica a posição de início e o tamanho da sequência. Desta maneira, longas sequências são representadas por códigos menores, realizando então a compressão do arquivo original [1, 2].

2.4.3 Codificação por sequência repetitiva

A *Run Length Encoding* - RLE, também conhecida como codificação por sequência repetitiva, substitui os símbolos idênticos por símbolos de tamanho menor. Quanto menor é a proporção de símbolos iguais, menor é a taxa de compressão. A principal desvantagem dessa técnica é que se o arquivo contém poucos símbolos idênticos, a taxa de compressão é muito baixa [1, 2].

2.4.4 Codificação por plano de bit

Na codificação por planos de bits a imagem original é decomposta em várias imagens binárias e então cada imagem binária pode ser comprimida por alguma técnica de compressão sem perdas [1].

2.5 Técnicas de Compressão de Imagens com Perdas

A compressão com perdas suprime informações do arquivo original de tal forma que essas informações não façam falta posteriormente. Essa perda da integridade dos dados originais faz com que os dados descomprimidos não sejam uma cópia idêntica dos dados originais, apenas aproximações que na maioria das vezes são bem satisfatórias. Devido à perda de parte da informação, os métodos de compressão com perdas conseguem atingir taxas de compressão bem maiores do que os métodos de compressão sem perdas [1].

2.5.1 Codificação por truncagem de blocos

A codificação por truncagem de blocos divide a imagem original em blocos não-sobrepostos de tamanho 4×4 . Calcula-se o valor médio ($\bar{x}$) e o desvio padrão (σ) para cada bloco, e o valor médio é utilizado como limiar. Os pixels de cada bloco que tem valor menor do que a média é atribuído o valor zero e os pixels que tem valor maior do que a média é atribuído o valor 1. Essa quantização individual de cada bloco resulta em um mapa de bits para cada bloco. Conhecendo-se o valor médio dos pixels, o desvio padrão e o mapa de bits resultante da quantização de cada bloco, consegue-se reconstruir, com perdas, bloco por bloco [1].

2.5.2 Codificação por transformada

A codificação por transformada é uma técnica que utiliza o domínio da frequência. A transformada é uma ferramenta matemática que permite que os dados sejam analisados no domínio da frequência e não mais no domínio do tempo. No caso de imagens, a transformada transporta a matriz de pixels, que é a própria imagem, no domínio do tempo para o domínio da frequência. A imagem no domínio da frequência concentra a maior parte da energia do sinal em poucos coeficientes da matriz transformada. Os coeficientes da matriz transformada que tem uma magnitude pequena podem ser desconsiderados sem causar grandes perdas na qualidade da imagem reconstruída e os coeficientes que apresentam a maior parte da energia são então quantizados e codificados por alguma técnica de compressão sem perdas. Isso é realizado bloco por bloco até atingir a imagem toda.

O processo de codificação por transformada começa com a divisão da imagem em blocos não-sobrepostos. A transformada é aplicada em cada bloco separadamente. Os coeficientes que possuem uma baixa concentração de energia são eliminados e os coeficientes com alta concentração de energia são quantizados e codificados por alguma técnica de compressão sem perdas.

A codificação por transformada pode ser resumida em quatro passos: partição da imagem, transformada direta, quantização e codificação de símbolos. A decodificação é o processo contrário e é resumida em quatro passos: decodificação de símbolos, transformação inversa, desquantização e junção da imagem [1].

Existem várias transformadas de imagem que podem ser utilizadas no processo de codificação por transformadas. Pode-se citar a Transformada de Fourier Discreta (*Discrete Fourier Transform* - DFT), a Transformada Cosseno Discreta (*Discrete Cosine Transform* - DCT), a Transformada de Hadamar, a Transformada *Wavelet* (*Discrete Wavelet Transform* - DWT), entre outras.

O padrão de compressão *Joint Photographic Experts Group* (JPEG) utiliza a DCT e é atualmente o padrão mais popular de compressão de imagens. O padrão de compressão JPEG2000, que será abordado no Capítulo 5 deste trabalho, usa a DWT.

2.6 Medidas de Desempenho

As técnicas de compressão com perdas podem ser analisadas em relação ao desempenho pelas seguintes considerações: taxa de bits (bits por pixel - bpp) e relação sinal ruído de pico (*Peak Signal Noise Ratio* - PSNR) [2, 3].

Para imagens digitais, a taxa de bits é obtida pela relação entre o tamanho do arquivo comprimido (em bits) e número de pixels da imagem original, como mostra a Equação (2.2):

$$Taxa\ de\ bits = \frac{tamanho\ arquivo\ comprimido\ em\ bits}{número\ de\ pixels\ da\ imagem\ original} \quad (2.2)$$

Quando um sinal é comparado antes e depois de um processo de degradação, a PSNR é a relação entre o máximo possível de potência de um sinal pela potência do ruído. No caso da compressão de imagens a imagem original é comparada com a imagem reconstruída (comprimida e descomprimida). A unidade usada para representar esse valor é o decibel (dB). Um valor alto de PSNR significa uma alta relação entre a potência do sinal pela potência do ruído, ou seja, uma melhor qualidade da imagem reconstruída. Em termos matemáticos, o valor do PSNR entre uma imagem original $(F(x, y))$ e uma imagem reconstruída $(F'(x, y))$ é obtido pela Equação (2.3) [2, 3, 4]:

$$PSNR = 10 \log_{10} \left(\frac{m*m*255^2}{\sum_{x=0}^{m-1} \sum_{y=0}^{m-1} ((F(x,y)) - (F'(x,y)))^2} \right) \quad (2.3)$$

2.7 Considerações Finais do Capítulo

Este capítulo mostrou a representação de imagens digitais e apresentou a fundamentação teórica sobre compressão de imagens digitais e algumas técnicas utilizadas para a compressão, tanto técnicas com perda quanto técnicas sem perdas. Foram mostrados também alguns métodos de comparação entre a imagem original e a imagem descomprimida.

O próximo capítulo apresenta a Teoria Fractal, que é a base para a Compressão Fractal de Imagens.

CAPÍTULO 3

TEORIA FRACTAL

3.1 Introdução

A principal característica dos fractais é auto-similaridade global existente neles, ou seja, qualquer parte de um fractal é similar a ele como um todo. Diversos exemplos de fractais podem ser encontrados na natureza, como por exemplo, a copa de uma árvore, ou uma nuvem. Outros exemplos de fractais podem ser encontrados na literatura, que são as estruturas matemáticas, tais como o triângulo de *Sierpinski* e a curva de *Koch*.

Os fractais podem ser gerados por transformações geométricas simples do próprio espaço métrico em si mesmo ou por equações recursivas.

Este capítulo apresenta a Teoria Fractal, mostrando a base matemática da geometria fractal e os métodos de geração de fractais.

3.2 Definição de Fractais

Fractais são definidos pela propriedade de auto-similaridade apresentando a mesma complexidade de detalhamento independente da escala em que são observados. Partes do fractal são similares ao fractal como um todo [5].

O termo fractal foi mencionado pela primeira vez por *Benoit Mandelbrot*, que o definiu como sendo uma forma que descreve padrões irregulares contendo uma auto-

similaridade global. São ditos irregulares, pois possuem comprimento infinito, detalhes infinitos, entre outras características [6].

Os fractais possuem uma riqueza de detalhes muito grande e podem facilmente ser identificados na natureza como, por exemplo, em árvores, em uma couve flor, em montanhas, em crateras, nuvens, entre outros, como mostra a Figura 3.1.

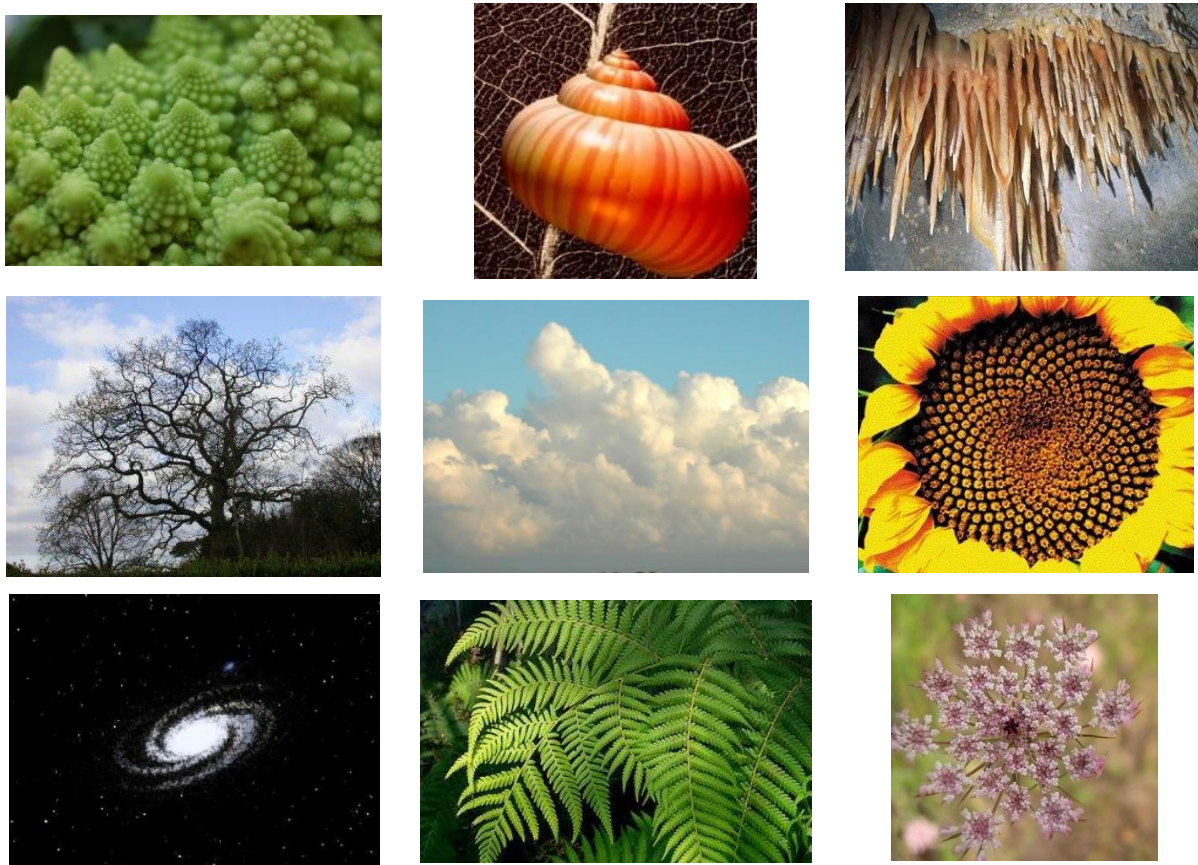


Figura 3.1 – Exemplos de fractais encontrados na natureza. Brócolis, caramujo, estalactites, copa de árvore, nuvem, semente de girassol, galáxia, samambaia, e flor.

Outros exemplos de fractais que se pode encontrar na literatura são as estruturas matemáticas. Essas estruturas são exatamente e globalmente auto-similares, pois elas são geradas por meio de transformações geométricas ou equações recursivas muito bem definidas. Alguns exemplos dessas estruturas matemáticas fractais mostradas na Figura 3.2 são: o triângulo de *Sierpinski*, o conjunto de *Cantor*, a curva de *Koch* e o conjunto de *Mandelbrot* [5, 6]. A seguir é descrito o processo de construção de algumas dessas estruturas matemáticas

fractais, tanto pelo processo de transformações geométricas quanto pela aplicação das equações recursivas.

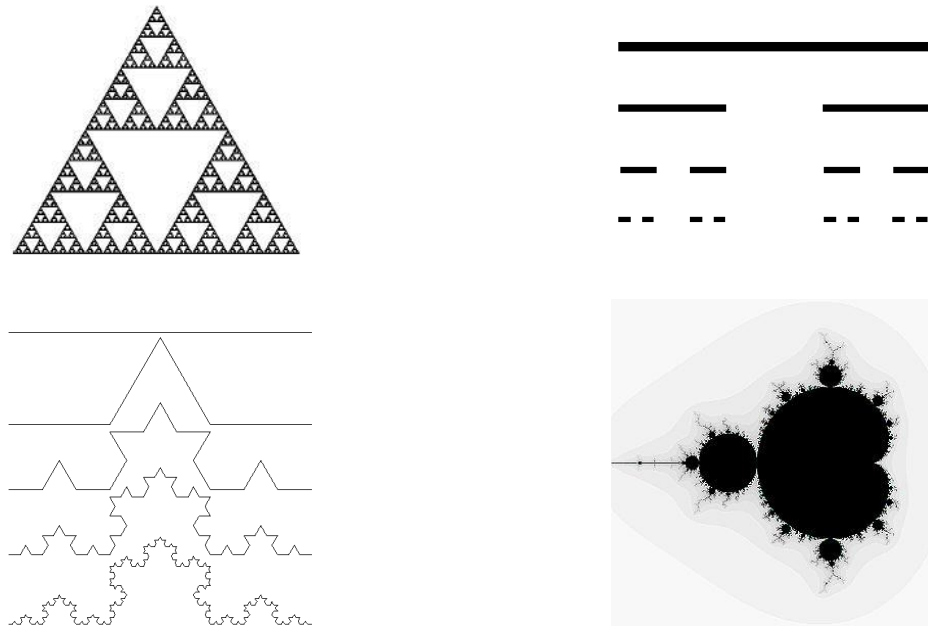


Figura 3.2 – Exemplos de estruturas matemáticas fractais: triângulo de *Sierpinski*, conjunto de *Cantor*, curva de *Koch* e conjunto de *Mandelbrot* [6].

3.3 Base Matemática da Geometria Fractal

Antes de exemplificar e descrever os processos de geração de fractais são definidos alguns conceitos da base matemática da geometria fractal que são referenciados nos itens subsequentes.

Os primeiros conceitos que são definidos são os de distâncias entre um ponto e um conjunto e entre dois conjuntos. Esses conceitos de distância serão utilizados para posterior definição de uma métrica utilizada na Teoria dos Fractais chamada Métrica de *Hausdorff*.

A distância entre um ponto $y \mid y \notin A$ e um conjunto compacto A é definida como a mínima distância entre o ponto y e o ponto $x \mid x \in A$, conforme a Equação (3.1) [5, 7]:

$$d(y, A) = \min_{x \in A} \{d(y, x)\} \quad (3.1)$$

A distância entre dois conjuntos A e B é definida pela máxima distância entre um ponto $x \mid x \in A$ e o conjunto B . Porém, como foi visto no parágrafo anterior, a distância entre um ponto $x \mid x \notin B$ e um conjunto compacto B , é definida como a menor distância entre o ponto x e o ponto $y \mid y \in B$. A definição de distância entre dois conjuntos pode ser melhor visualizada pela Equação (3.2) [5, 7]:

$$d(A, B) = \max_{x \in A} \{d(x, B)\} = \max_{x \in A} \{\min_{y \in B} \{d(x, y)\}\} \quad (3.2)$$

Pode-se observar na Equação (3.2) que a distância entre A e B é diferente da distância entre B e A , ou seja, $d(A, B) \neq d(B, A)$. A Equação (3.3) define a distância entre os conjuntos B e A . A Figura 3.3 exemplifica essa diferença [5, 7].

$$d(B, A) = \max_{y \in B} \{d(y, A)\} = \max_{y \in B} \{\min_{x \in A} \{d(y, x)\}\} \quad (3.3)$$

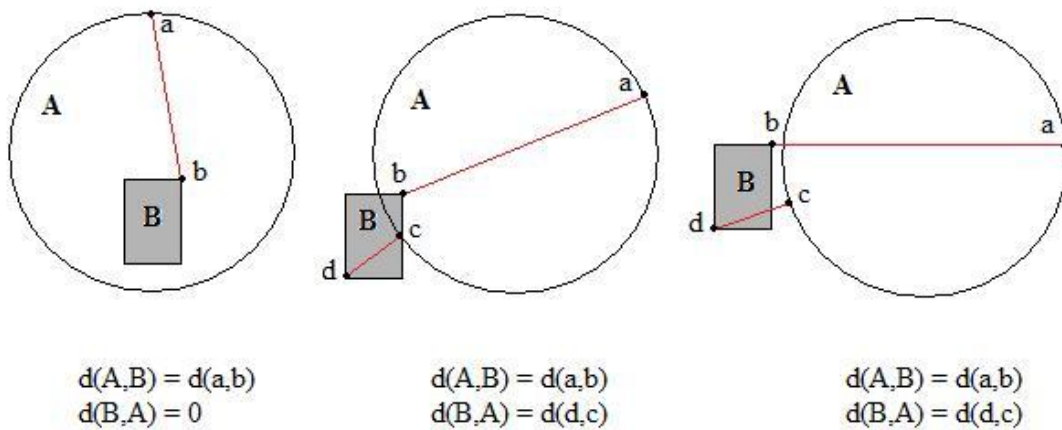


Figura 3.3 – Distância entre um ponto e um conjunto e distância entre dois conjuntos.

Outra definição importante para entender a teoria fractal é a *Métrica* e o *Espaço Métrico*. Considere o espaço X como sendo um conjunto, e a coleção de pontos do próprio espaço, que são as imagens propriamente ditas, como elementos desse conjunto. Os fractais podem ser definidos em um espaço métrico qualquer, mas aqui o objetivo é o plano ($\mathbb{R}^2$), pois

as imagens são elementos bidimensionais. Um espaço métrico é um par (X, d) definido como um espaço X juntamente com uma função real de distância d ($d: X \times X \rightarrow \mathbb{R}$), que associa dois elementos de um conjunto a um número real, ou seja, mede a distância entre dois pontos, x e y quaisquer, dentro do espaço X . Essa função real de distância d é chamada de métrica e ela é a formalização matemática do conceito de distância. Uma métrica associa a cada par de pontos $(x, y \in X)$ um número real $d(x, y)$ que é a distância entre x e y . Uma métrica deve satisfazer as propriedades definidas pelas Equações (3.4) a (3.7) [5, 7].

$$d(x, x) = 0, \quad \forall x \in X \quad (3.4)$$

$$d(x, y) = d(y, x), \quad \forall x, y \in X \quad (3.5)$$

$$0 < d(x, y) < \infty, \quad \forall x, y \in X \text{ e } x \neq y \quad (3.6)$$

$$d(x, y) \leq d(x, z) + d(z, y), \quad \forall x, y, z \in X \quad (3.7)$$

A métrica mais natural para calcular a distância entre dois pontos é a Métrica Euclidiana. Considerando o plano ($\mathbb{R}^2$), a distância euclidiana entre dois pontos $x, y \in \mathbb{R}^2$ é encontrada pela Equação (3.8).

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \quad (3.8)$$

Generalizando para $\mathbb{R}^n$ tem-se a Equação (3.9).

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (3.9)$$

3.3.1 Sequência de *Cauchy* e espaços métricos completos

Para definir um espaço métrico completo primeiro é necessário definir o que é uma sequência de *Cauchy*.

Em um espaço métrico (X, d) , uma sequência $(x_n: 0 < n < \infty)$ é considerada uma sequência de Cauchy se para $x_n \in X$ e para qualquer $\varepsilon > 0 \mid \varepsilon \in \mathbb{R}$, existe um número inteiro $N > 0 \mid N \in \mathbb{N}$, tal que $d(x_n, x_m) < \varepsilon, \forall m, n > N$. Isto quer dizer que quanto maior é o número de elementos da sequência (n) , mais eles se aproximam um do outro. Nem toda sequência de *Cauchy* é convergente, pois uma sequência de *Cauchy* pode convergir para um ponto que não está dentro do espaço X . Então, pode-se afirmar que toda sequência convergente é de *Cauchy*, mas nem toda sequência de *Cauchy* é convergente.

Um espaço métrico é dito completo se toda a sequência de *Cauchy* no espaço X converge para um ponto dentro do intervalo do espaço X [5, 7].

3.3.2 Métrica e espaço métrico de Hausdorff

Dado um espaço métrico completo (X, d) , o espaço Métrico de Hausdorff $(\mathcal{H}(X), h)$ é o espaço cujos pontos são subconjuntos compactos não vazios de X , onde h é chamada de métrica de Hausdorff.

Primeiro, suponha um espaço métrico completo (X, d) , tendo $x, y \in X$ e $A, B \in \mathcal{H}(X)$ como subconjuntos compactos do espaço métrico (X, d) . A métrica de Hausdorff é definida conforme a Equação (3.10) [5, 7]:

$$h = \max_{A, B \in \mathcal{H}(X)} \{ \max_{x \in A} \{ \min_{y \in B} \{ d(x, y) \} \}, \max_{y \in B} \{ \min_{x \in A} \{ d(x, y) \} \} \} \quad (3.10)$$

Considerado A e B subconjuntos de $\mathcal{H}(X)$ e, x e y pontos do conjunto X , para entender a Equação (3.10), são definidas as distâncias entre um dos pontos do conjunto e um dos subconjuntos, por exemplo, $d(x, B)$, e entre os dois subconjuntos, $d(A, B)$ [8].

1. Distância entre um ponto $x \in X$ e um subconjunto $B \in \mathcal{H}(X)$ é um subconjunto de X , conforme a Equação (3.11):

$$d(x, B) = \min_{y \in B} \{ d(x, y) \} \quad (3.11)$$

2. Distância entre dois conjuntos $A, B \in \mathcal{H}(X)$ que são subconjuntos de X , conforme a Equação (3.12):

$$d(A, B) = \max_{x \in A} \{d(x, B)\} \quad (3.12)$$

A distância entre conjuntos não pode ser considerada uma métrica, pois pode-se observar que, nem sempre $d(A, B) = d(B, A)$. Então, a métrica de *Hausdorff* é definida como a máxima distância entre essas duas. As Equações (3.13) a (3.15) mostram a métrica de *Hausdorff* [5, 7, 8]:

$$h(A, B) = \max_{A, B \in \mathcal{H}(X)} \{d(A, B), d(B, A)\} \quad (3.13)$$

$$h(A, B) = \max_{A, B \in \mathcal{H}(X)} \{\max_{x \in A} \{d(x, B)\}, \max_{y \in B} \{d(y, A)\}\} \quad (3.14)$$

$$h(A, B) = \max_{A, B \in \mathcal{H}(X)} \{\max_{x \in A} \{\min_{y \in B} \{d(x, y)\}\}, \max_{y \in B} \{\min_{x \in A} \{d(x, y)\}\}\} \quad (3.15)$$

3.4 Geração de Fractais

Existem várias maneiras de gerar fractais, tanto por transformações geométricas simples do próprio espaço métrico em si mesmo, quanto por equações recursivas.

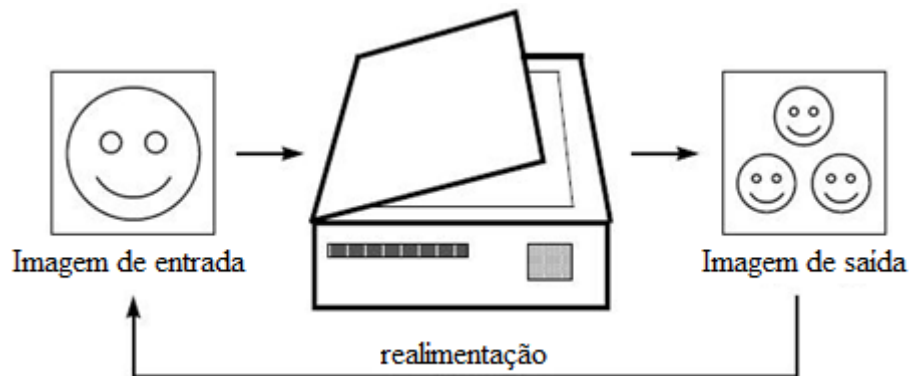


Figura 3.4 – 1ª iteração da máquina fotocopadora [3].

Uma analogia feita por *Fischer* consegue mostrar a geração de um fractal por transformações geométricas. Tendo uma imagem qualquer como entrada, uma fotocopadora especial reduz essa imagem inicial pela metade e como saída tem-se essa imagem reduzida reproduzida três vezes como mostra a Figura 3.4 [3].

Se a imagem de saída obtida por essa máquina fotocopadora especial é colocada como imagem de entrada nessa mesma máquina fotocopadora e repetir-se esse processo iterativamente, é obtido então um fractal, onde partes dele são similares a ele como um todo.

Após algumas iterações, para diferentes imagens de entrada colocadas nessa máquina copiadora, pode-se observar que as imagens de saída, independentemente da imagem de entrada colocada, tendem para uma mesma imagem final, para essa máquina copiadora. Essa imagem é chamada de “atrator” [3].

Algumas iterações desse processo de construção do fractal são mostradas na Figura 3.5.

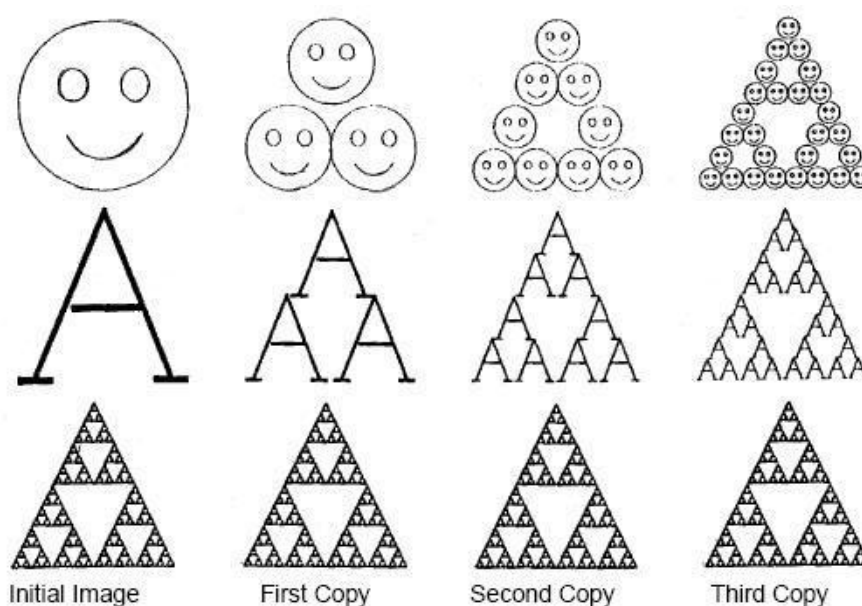


Figura 3.5 – Primeiras iterações da máquina fotocopadora especial para várias imagens de entrada [3].

As transformações geométricas aplicadas na imagem de entrada da máquina copiadora são chamadas de “transformações afins”. Essas transformações determinam o “atrator”.

3.5 Sistema de Funções Iterativas (IFS)

Um dos métodos conhecidos para a geração de fractais é o chamado “Sistema de funções Iterativas”, do inglês *Iterated Function System* (IFS). Esse método foi desenvolvido pelo matemático inglês *Michael Barnsley* por volta de 1986. O IFS é baseado na aplicação de transformações afins sobre alguma imagem inicial arbitrária [8].

Para a geração de um fractal qualquer, as transformações afins são equações recursivas aplicadas, por exemplo, a um sistema de coordenadas (x,y) do plano $(\mathbb{R}^2)$. É escolhido um ponto qualquer A , então, uma das equações (transformações afins) é selecionada aleatoriamente e obtém-se as coordenadas do próximo ponto B , e assim sucessivamente. A união desses pontos gerados pelas transformações no plano é o atrator para aquele determinado conjunto de transformações afins. O conjunto de transformações afins (w_i 's) responsáveis pela geração de diversos fractais pode ser conforme a Equação (3.16) [3].

$$w_i \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} \quad (3.16)$$

Para ser gerado, cada atrator necessita de um número de transformações. Para gerar o do triângulo de *Sierpinski* são necessárias quatro transformações.

Dependendo dos valores dos coeficientes de transformações, eles podem transladar, escalonar, refletir ou rotacionar uma imagem de entrada.

Independente da imagem de entrada, para um mesmo conjunto de transformações, tem-se sempre a imagem final tendendo a um mesmo atrator. Para diferentes conjuntos de transformações, tem-se diferentes atratores. Uma limitação técnica das transformações afins que definem um fractal é que elas devem ser contrativas, ou seja, aproximar dois pontos quaisquer da imagem de entrada [5].

O conjunto de transformações contrativas, aplicadas iterativamente a uma imagem inicial qualquer, necessária para definir um fractal é chamado de Sistema de Funções Iterativas (IFS). Esse conjunto mapeia o plano $\mathbb{R}^2$ para si mesmo $\{w_i: \mathbb{R}^2 \rightarrow \mathbb{R}^2 | i = 1, 2, \dots, n\}$ e define o mapa mostrado pela Equação (3.17) [5, 8].

$$W(.) = \cup_{i=1}^n w_i (.) \quad (3.17)$$

onde:

n – número de transformações contrativas necessárias para definir um atrator.

Após aplicação de um conjunto de transformações, o mapeamento contrativo garante que a distância euclidiana entre dois pontos quaisquer em uma imagem é menor do que a distância euclidiana original entre esses mesmos dois pontos. A definição matemática para contratividade é [5]:

“Uma transformação $f: X \rightarrow X$ em um espaço métrico (X,d) é chamada contrativa ou mapeamento contrativo se existe uma constante real $0 \leq s < 1$ tal que:

$$d(f(x), f(y)) \leq s \cdot d(x, y), \forall x, y \in X, \quad (3.18)$$

onde:

s – fator de contratividade de f . ”

O mapa W não é aplicado ao plano (x,y) e sim a um subconjunto do plano, que é uma coleção de pontos do próprio plano (imagem inicial). Se a imagem inicial é tratada como um subconjunto f_0 do plano (x,y) , aplicando uma vez o mapa contrativo W nessa imagem de entrada, é obtida $f_1 = W(f_0)$. Aplicando a segunda vez é obtida $f_2 = W(f_1) = W(W(f_0)) = W^2(f_0)$, e assim por diante. O atrator resultante para o mapa W é mostrado na Equação (3.19).

$$f_\infty = \lim_{n \rightarrow \infty} W^{on}(f_0) \quad (3.19)$$

Um Sistema de Funções Iterativas é descrito unicamente por um conjunto de transformações contrativas definidas em um espaço métrico completo.

3.5.1 Fractais Gerados por IFS

A construção de fractais mostrada anteriormente é pelas transformações geométricas. Os fractais podem ser gerados também pelo método IFS. A seguir são mostrados alguns exemplos de fractais gerados por esse método [5].

Conjunto de Cantor

O fractal denominado conjunto de *Cantor* é gerado por 2 transformações afins, mostradas nas Equações (3.20) e (3.21):

$$w_1(x) = \frac{1}{3}x \quad (3.20)$$

$$w_2(x) = \frac{1}{3}x + \frac{2}{3} \quad (3.21)$$

O processo é iniciado com um segmento de reta denominado A, que é uma coleção de pontos do espaço $\mathbb{R}$. O conjunto de transformações afins é definido como $W(A) = w_1(A) \cup w_2(A)$ e o atrator/fractal (conjunto de cantor) é definido como $CC = \lim_{n \rightarrow \infty} W^n(A)$. A Figura 3.6 mostra o fractal denominado conjunto de Cantor.



Figura 3.6 – Conjunto de Cantor [5].

Curva de Koch

A curva de *Koch* é definida por 4 transformações afins do plano $\mathbb{R}^2$, mostradas pelas Equações (3.22) a (3.25).

$$w_1 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{3} \begin{pmatrix} \cos 0 & -\sin 0 \\ \sin 0 & \cos 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (3.22)$$

$$w_2 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{3} \begin{pmatrix} \cos \pi/3 & -\sin \pi/3 \\ \sin \pi/3 & \cos \pi/3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (3.23)$$

$$w_3 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{3} \begin{pmatrix} \cos(-\pi/3) & -\sin(-\pi/3) \\ \sin(-\pi/3) & \cos(-\pi/3) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 3/2 \\ \sqrt{3}/2 \end{pmatrix} \quad (3.24)$$

$$w_4 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{3} \begin{pmatrix} \cos 0 & -\sin 0 \\ \sin 0 & \cos 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 2 \\ 0 \end{pmatrix} \quad (3.25)$$

Para gerar a curva de *Koch*, inicia-se com um segmento de reta A, que é uma coleção de pontos do espaço $\mathbb{R}^2$.

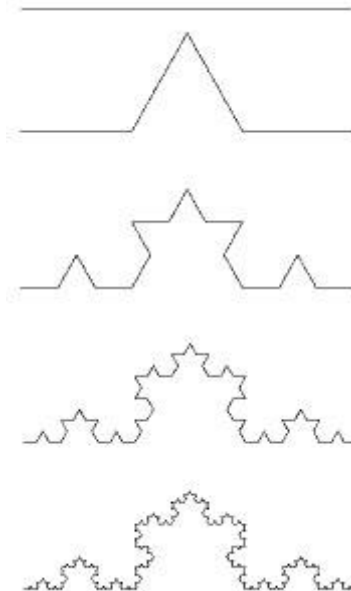


Figura 3.7 – Curva de Koch [5].

O conjunto de quatro transformações afins $W(A) = w_1(A) \cup w_2(A) \cup w_3(A) \cup w_4(A)$ aplicado iterativamente ao segmento de reta que é tido como imagem inicial, define o fractal curva de *Koch* ($CK = \lim_{n \rightarrow \infty} W^n(A)$), como mostra a Figura 3.7.

Triângulo de Sierpinski

O triângulo de *Sierpinski* é gerado pelo método IFS por 3 transformações afins do plano $\mathbb{R}^2$, $w_i = \mathbb{R}^2 \rightarrow \mathbb{R}^2, i = 1, 2, 3$, mostradas nas Equações (3.26) a (3.28).

$$w_1 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (3.26)$$

$$w_2 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1/2 \\ 0 \end{pmatrix} \quad (3.27)$$

$$w_3 \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1/4 \\ 1/2 \end{pmatrix} \quad (3.28)$$

O processo de geração do triângulo de *Sierpinski* pelo método IFS é iniciado com um triângulo que é uma coleção de pontos do espaço $\mathbb{R}^2$. O conjunto de transformações $W(A) = w_1(A) \cup w_2(A) \cup w_3(A)$ aplicado iterativamente na imagem inicial gera o triângulo $TS = \lim_{n \rightarrow \infty} W^n(A)$ como mostra a Figura 3.8.

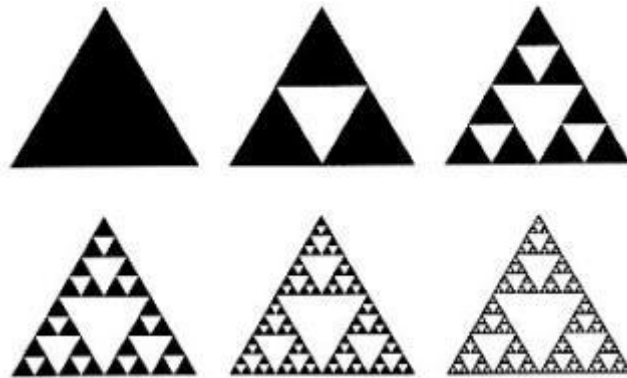


Figura 3.8 – Processo de construção do triângulo de *Sierpinsky* [5].

Como citado anteriormente, um atrator resultante para um dado conjunto de transformações afins independe da imagem inicial que essas transformações são aplicadas. Portanto, se é colocado como imagem inicial para as transformações que geram o triângulo de *Sierpinski*, por exemplo, um “*smile*” ou uma letra qualquer, após várias iterações o atrator final continua o mesmo, como mostra a Figura 3.5.

3.6 Considerações Finais do Capítulo

Este capítulo apresentou a teoria dos fractais, base da compressão fractal de imagens, mostrando desde a base matemática da teoria até como é a geração dos fractais. O Sistema de Funções Iterativas (IFS) é um método de geração de fractais baseado na aplicação de transformações contrativas (equações recursivas) em uma imagem inicial arbitrária. Esta imagem após um determinado número de iterações tende a uma imagem conhecida como atrator, que é o próprio fractal. Os métodos de compressão fractal de imagens utilizam como base um sistema inspirado no IFS, que usa a similaridade existente nas imagens digitais reais, diferente da auto-similaridade global existente nos fractais.

O próximo capítulo descreve a compressão fractal de imagens, que utiliza a similaridade existente em imagens reais e os métodos de compressão existentes baseados na teoria dos fractais.

CAPÍTULO 4

COMPRESSÃO FRACTAL DE IMAGENS

4.1 Introdução

A teoria dos fractais pode ser explorada em análise e processamento de dados reais, como por exemplo, no processamento digital de imagens, mais especificamente na compressão de imagens. A compressão fractal de imagens é baseada nos princípios da Teoria Fractal, utilizando o conceito de auto-similaridade existente nas imagens digitais reais que podem ser modeladas como conjuntos fractais [9].

Ao contrário dos fractais, as imagens digitais reais não possuem auto-similaridade global, portanto, não se pode associar o método de geração de fractais (IFS) às imagens reais, descrevendo-as como um conjunto de transformações contrativas. No entanto, as imagens reais possuem algumas áreas que apresentam similaridades, mesmo que em diferentes escalas e diferentes posições [3]. A Figura 4.1 mostra as áreas da “Lena” onde existem similaridades, como por exemplo, nas regiões do chapéu, espelho e ombro.

São essas similaridades encontradas dentro das imagens reais que são utilizadas para codificar a imagem utilizando a técnica de Sistema de Funções Iterativas Particionadas, do inglês *Partitioned Iterated Function System* (PIFS). Esta técnica é inspirada na técnica IFS.



Figura 4.1 – Similaridades dentro da própria imagem [3].

A diferença entre as técnicas PIFS e IFS, é que na técnica PIFS a imagem é dividida em blocos e é possível encontrar similaridades entre esses blocos. O que se deseja é encontrar dentro da própria imagem um bloco que depois de aplicadas determinadas transformações, melhor substitua outro bloco [10]. Obviamente, um bloco dentro da imagem, mesmo depois de aplicadas as transformações, não substitui outro bloco de maneira perfeita, pois não existe auto-similaridade global em imagens reais. Portanto, a imagem codificada pela técnica PIFS, após ser decodificada, é uma aproximação da imagem original e não é uma cópia idêntica da mesma. É por isso que a compressão fractal de imagens que utiliza a técnica PIFS é dita compressão com perdas, pois admite perda de qualidade da imagem [11].

4.2 Método da Força Bruta

A compressão fractal de imagens baseia-se na teoria dos fractais no sentido de que a auto-similaridade existente em imagens reais é utilizada para codificar a representação das imagens ao invés de representá-las pixel a pixel. O que é armazenado são os coeficientes das transformações realizadas para identificar as auto-similaridades.

Para a compressão fractal de imagens reais é utilizada a técnica PIFS onde a imagem original é particionada em blocos não sobrepostos chamados *range blocks* e em blocos chamados *domain blocks* que podem ser ou não sobrepostos [3, 12]. Esse método de partição

é chamado de método da força bruta. A Figura 4.2 mostra as partições em *range blocks* e em *domain blocks*. Pode-se verificar na partição em *domain blocks* a possibilidade de sobreposição e também pode-se observar que o tamanho dos *domain blocks* é maior do que o tamanho dos *range blocks*. Normalmente, o tamanho dos *domain blocks* é o dobro dos *range blocks* [3].

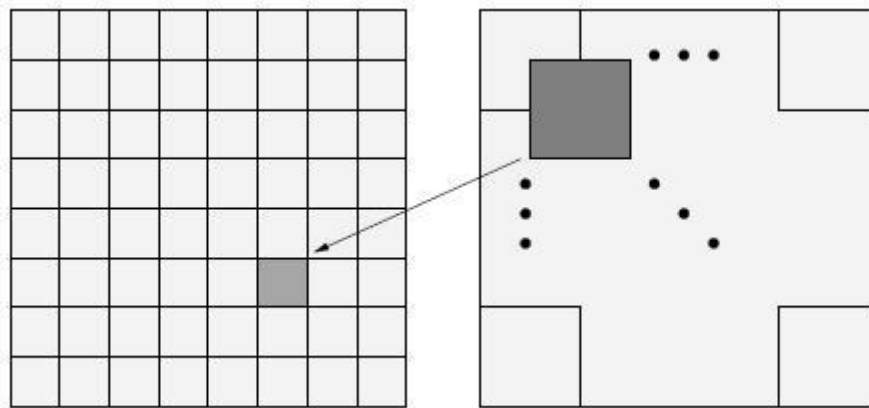


Figura 4.2 – Range block e domain block [13].

Com a finalidade de encontrar similaridade entre cada um dos *range blocks* e os *domain blocks*, nestes são aplicadas transformações geométricas (rotação, inversão, redução) e ajuste de brilho e contraste. Nessas transformações, primeiro os *domain blocks* são reduzidos para que fiquem do mesmo tamanho do *range block* e poderem ser comparados entre si. Depois são aplicadas 8 transformações: 4 rotações (0° , 90° , 180° , 270°) e 4 inversões (diagonal principal, diagonal secundária, vertical e horizontal) [14, 15]. A Figura 4.3 mostra as 8 transformações que são aplicadas a cada *domain block* depois do mesmo ser reduzido ao tamanho do *range block*.

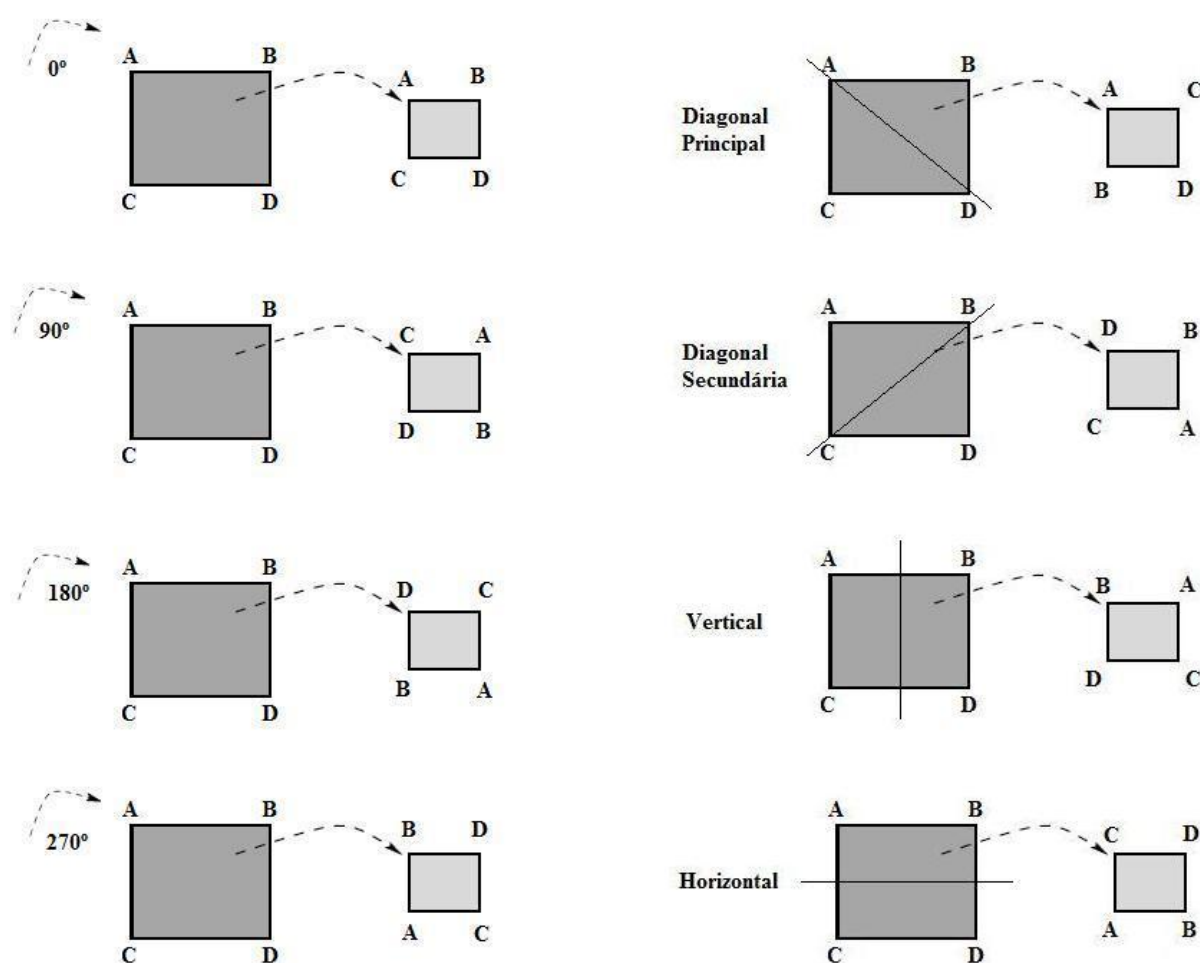


Figura 4.3 – 8 transformações (4 rotações e 4 inversões) [12].

A métrica utilizada para fazer a comparação entre cada *range block* extraído da imagem original e todos os *domain blocks* também extraídos dessa imagem e posteriormente transformados é chamada raiz quadrática da média dos valores, do inglês *Root Mean Square* (RMS), denominada erro RMS. Essa métrica é a medida da distância (erro) entre o *range block* e o *domain block* em questão. Matematicamente, para fazer a comparação entre os dois blocos, usa-se a métrica RMS simplificada mostrada na Equação (4.1) [3].

$$RMS = \sum_{k=1}^{n \times n} (s_i \cdot d_k + o_i - r_k)^2 \quad (4.1)$$

onde:

s – contraste;

o – brilho;

d – *domain block*;

r – *range block*;

n – tamanho do *domain block* e do *range block*;

i – índice do *domain block* e do *range block* em questão;

k – índice do pixel dentre os $n \times n$ existentes.

Com a finalidade de se ter um menor valor de erro RMS entre ele e o *range block*, encontra-se os valores ótimos do contraste e do brilho para o *domain block* transformado. O valor mínimo para RMS calculado a partir da Equação (4.1) ocorre quando as derivadas parciais em relação a s e o são zero e isto ocorre quando s_i e o_i se comportam como nas Equações (4.2) e (4.3) [3].

$$s_i = \frac{n \times n (\sum_{k=1}^{n \times n} d_k r_k) - [(\sum_{k=1}^{n \times n} d_k)(\sum_{k=1}^{n \times n} r_k)]}{n(\sum_{k=1}^{n \times n} d_k^2) - (\sum_{k=1}^{n \times n} d_k)^2} \quad (4.2)$$

$$o_i = \frac{(\sum_{k=1}^{n \times n} r_k) - s_i (\sum_{k=1}^{n \times n} d_k)}{n \times n} \quad (4.3)$$

Em cada *domain block* transformado é realizado um ajuste de brilho e contraste utilizando-se as Equações (4.2) e (4.3) a partir da métrica RMS. Pode-se verificar nas equações que para encontrar os valores ótimos de brilho e contraste para fazer o ajuste do *domain block* leva-se em conta os valores dos pixels tanto do *domain block* em questão quanto do *range block* em questão, pois esse ajuste é feito para minimizar o erro entre o *domain block* e o *range block*. Esse ajuste é calculado através pela Equação (4.1) que encontra o erro RMS.

Após as transformações geométricas e o ajuste de brilho e contraste para cada um dos *domain blocks*, cada *range block* extraído da imagem original é comparado com cada *domain block* transformado e é encontrado então o melhor *matching* entre cada *range block* e os *domain blocks* transformados. O melhor *matching* significa o menor erro RMS entre cada *range block* extraído da imagem original e os *domain blocks* transformados. A cada *range block* é associado a posição do *domain block* que obteve o melhor *matching*, o coeficiente da transformação geométrica, e o ajuste de brilho e contraste. Esses coeficientes associados a cada parte da imagem original (*range block*) são chamados de *CÓDIGO FRACTAL*. A Figura 4.4 mostra o esquema de compressão fractal utilizando o método da força bruta.

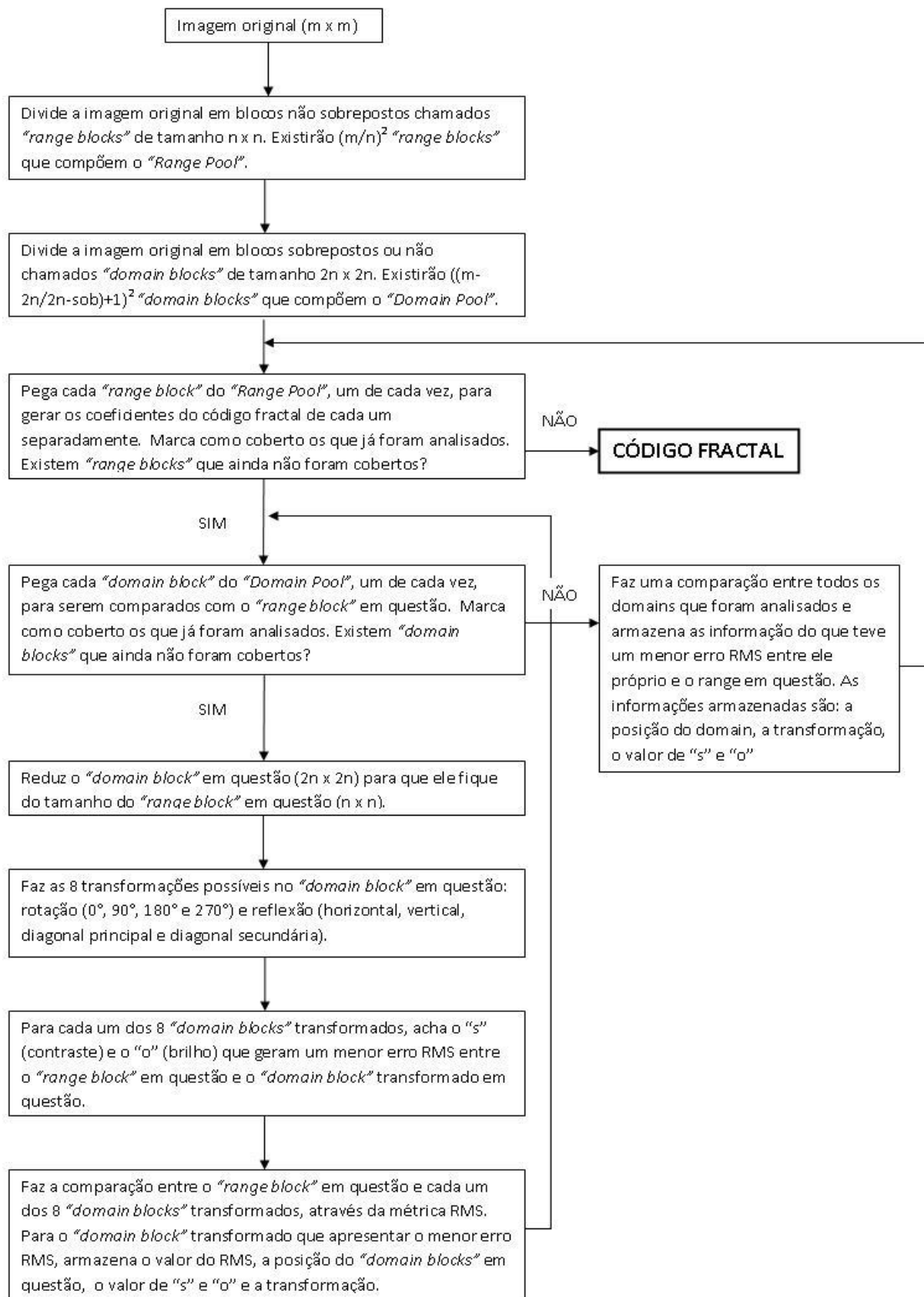


Figura 4.4 – Esquema de compressão fractal pelo método da força bruta.

Os parâmetros de entrada para a compressão fractal de uma imagem baseada no método da força bruta são:

- Tamanho do *range block*;
- Sobreposição;
- Máximo valor para o contraste;
- Quantidade de bits para representar o contraste;
- Quantidade de bits para representar o brilho.

O parâmetro “tamanho do *range block*” determina não só o tamanho dos *range blocks* como também o tamanho dos *domain blocks*, pois neste trabalho eles são considerados sempre o dobro do tamanho dos *range blocks*. O parâmetro “sobreposição” determina quantos pixels serão sobrepostos na construção do *Domain Pool*. Quanto maior é a sobreposição, maior é o *Domain Pool*. O parâmetro “máximo valor do contraste”, determina o valor máximo estipulado para o contraste (s_{max}). Os parâmetros “quantidade de bits para representar o contraste” e “quantidade de bits para representar o brilho” são utilizados na etapa de quantização do valor do contraste e do brilho.

Após todas as comparações entre cada um dos *range block* que compõem a imagem original e todos os *domain blocks* transformados, é realizada a escolha do melhor *matching* para cada *range block*, ou seja, o que tem o menor erro RMS. Então, para cada *range block*, armazena-se os coeficientes que definem o *domain block* que gerou o menor erro RMS. O conjunto de coeficientes armazenados referentes à imagem original de entrada obtidos pela compressão de imagem utilizando o método da força bruta, que é o próprio código fractal referente à imagem de entrada é composto pelos seguintes coeficientes:

- Posição do *domain block* (que gerou o menor RMS);
- A transformação geométrica do *domain block* (que gerou o menor RMS);
- O valor de s (ajuste de contraste);
- O valor de o (ajuste de brilho).

O coeficiente “posição do *domain block*” indica a posição, dentro da imagem original, do *domain block* que teve o menor erro RMS. O coeficiente “transformação” indica qual a

transformação realizada em tal *domain block* que gerou o menor erro RMS. Os coeficientes “valor de contraste” e “valor de brilho” indicam os valores ótimos de contraste e brilho que deverão ser feitas para aquele determinado *range block*.

Para ser armazenado cada coeficiente necessita de uma quantidade de bits específica. O coeficiente “transformação geométrica” é representado utilizando 3 bits, pois existem apenas 8 transformações possíveis. A quantidade de bits utilizada para armazenar o coeficiente “posição do *domain block*” depende do tamanho dos *range blocks* e da sobreposição dos *domain blocks*, pois esses parâmetros de entrada determinam o tamanho do *Domain Pool*.

Os coeficientes, valor de *s* (contraste) e valor de *o* (brilho), são valores não-quantizados e a quantização desses coeficientes é realizada com a finalidade de diminuir o tamanho do arquivo do código fractal aumentando assim a taxa de compressão. A escolha do número de bits que são utilizados para quantizar esses coeficientes foi feita em cima de um estudo dos testes realizados. Os testes foram realizados utilizando várias hipóteses de número de bits e os mesmos são apresentados nesse capítulo. Os números de bits escolhidos para quantizar esses coeficientes são parâmetros de entrada para a compressão fractal.

Para realizar a descompressão fractal basta aplicar os coeficientes do código fractal em uma imagem inicial qualquer e então a aproximação da imagem original é reconstruída. Essa aproximação da imagem original reconstruída é relacionada com o atrator que foi visto no Capítulo 3 deste trabalho onde foi apresentada a teoria dos fractais.

A qualidade da imagem depende do número de iterações aplicadas à imagem inicial qualquer e também dos parâmetros de entrada. Em relação aos parâmetros de entrada, quanto menores são os *range blocks*, melhor é a qualidade da imagem reconstruída, mas menos eficiente é a compressão, no sentido de taxa e tempo de compressão. Quanto mais pixels sobrepostos na etapa de construção do *Domain Pool*, melhor é a qualidade da imagem reconstruída, mas menos eficiente é a compressão.

Em relação ao número de iterações na descompressão, normalmente é necessário 10 iterações para que a imagem reconstruída estabilize em relação a PSNR [3].

4.2.1 Testes e resultados utilizando o método da força bruta

Foi desenvolvido um algoritmo em Matlab, baseado no método da força bruta. Foram realizados testes variando os cinco parâmetros de entrada: tamanho dos *range blocks*, sobreposição na construção do *Domain Pool*, quantidade de bits utilizados na quantização do contraste e do brilho e valor máximo do contraste. Para os testes, foi utilizada a imagem da “Lena” 64 x 64 pixels como imagem original a ser comprimida pelo método da força bruta. Abaixo é descrito como a variação de cada um dos parâmetros de entrada afeta a qualidade da imagem comprimida.

Variação do tamanho dos range blocks

O tamanho dos blocos utilizados na etapa de compressão fractal de imagem influencia bastante a qualidade da imagem reconstruída. Para o teste variando o tamanho dos *range blocks* foram utilizados tamanhos diferentes para os *range blocks* e sempre o tamanho dos *domain blocks* é o dobro do tamanho dos *range blocks*. Quanto menores são os blocos melhor é a qualidade da imagem reconstruída, porém, maior será a taxa de bits, já que o tamanho do *Domain Pool* aumenta, sendo necessária uma maior quantidade de bits para representar o coeficiente “posição do *domain block*” (que gerou o menor RMS). Com o aumento da quantidade de blocos, quantidade de comparações realizadas até que se encontre o melhor *matching* é maior fazendo com que mais tempo seja dispendido na etapa de compressão.

Variação da sobreposição

O aumento da sobreposição dos *domain blocks* na construção do *Domain Pool* afeta positivamente a qualidade da imagem reconstruída. Por outro lado, quanto maior é a sobreposição, maior é *Domain Pool*, aumentando a taxa de bits e fazendo com que o tempo dispendido na etapa de compressão seja maior. A Tabela 4.1 a seguir mostra a variação dos valores de PSNR das imagens reconstruídas utilizando a imagem original da “Lena” a ser comprimida com tamanho 64 x 64 pixels, variando o valor de sobreposição e o tamanho do *range block*, onde “*n*” corresponde ao tamanho desse *range block*. Os outros parâmetros de

entrada foram fixados em: quantidade de bits para representar o contraste = 5, quantidade de bits para representar o brilho = 7 e valor máximo do contraste = 1,2.

Tabela 4.1 – Valores de PSNR variando a sobreposição e o tamanho do *range block*.

Sobreposição	PSNR (dB) para n = 4	PSNR (dB) para n = 8
0	25,55	19,37
25%	26,17	19,65
50%	26,81	19,89
75%	28,22	20,46

Ao analisar os resultados da Tabela 4.1, pode-se verificar que quanto maior é a sobreposição, maior é o PSNR da imagem reconstruída em relação a imagem original, ou seja, melhor é a qualidade da imagem reconstruída. Em relação ao tamanho do *range block*, pode-se observar que quanto menor é esse tamanho, melhor é a qualidade da imagem reconstruída, pois o valor da PSNR foi maior quando foi utilizado o menor tamanho para o *range block*.

Variação do número de bits utilizados na quantização dos valores de s (contraste) e o (brilho)

A quantização dos valores de contraste e brilho é realizada com a finalidade de diminuir o número de bits utilizados para representá-los, reduzindo assim a taxa de bits da imagem comprimida. Essa quantização é realizada baseada nos dois parâmetros de entrada, “quantidade de bits para representar o contraste” e “quantidade de bits para representar o brilho”.

Foram realizados testes utilizando diferentes quantidades de bits para representar os valores de contraste (s) e brilho (o). Primeiro, utilizou-se s_{bits} igual a 5 e variou-se o_{bits} de 5 a 9. Depois utilizou-se o_{bits} igual a 7 e variou-se s_{bits} de 3 a 7. A Tabela 4.2 mostra os valores de PSNR para a imagem reconstruída, variando os valores dos bits utilizados para representar o contraste e o brilho. Os outros parâmetros de entrada foram fixados em: tamanho do *range block* = 4, sobreposição = 0 e valor máximo do contraste = 1,2.

Tabela 4.2 – Valores de PSNR variando a quantidade de bits para representar o contraste (s_{bits}) e o brilho (o_{bits}).

Sbits	Obits	PSNR (dB)
5	5	23,77
5	6	24,68
5	7	25,55
5	8	25,98
5	9	26,14
3	7	25,05
4	7	25,40
5	7	25,55
6	7	25,52
7	7	25,50

Pode-se observar na Tabela 4.2 que o valor de PSNR é maior quanto mais bits são utilizados na quantização dos coeficientes, de contraste e de brilho. Porém, quanto mais bits são utilizados, maior é a taxa de bits da imagem comprimida.

Considerando *Fischer* [3] e os resultados da Tabela 4.2, é utilizado nos testes deste trabalho 5 bits para quantizar o contraste e 7 bits para quantizar o brilho.

Variação do valor máximo do contraste

Considerando o número de bits para representar o contraste (s) igual a 5 e para representar o brilho (o) igual a 7, os testes agora são para definir o máximo valor do contraste (s_{max}). O intervalo para os valores de s é restrito a valores positivos ($s > 0$) e com a finalidade de reduzir a taxa de bits define-se o máximo valor do contraste [3, 16]. A Tabela 4.3 mostra os valores de PSNR para a imagem reconstruída, variando o valor máximo do contraste. Os outros parâmetros de entrada utilizados foram: tamanho do *range block* = 4, sobreposição = 0, quantidade de bits para representar o contraste = 5 e quantidade de bits para representar o brilho = 7.

Tabela 4.3 – Valores de PSNR variando o valor máximo para o contraste.

$S_{\max}$	PSNR (dB)
0,5	23,44
1	25,52
1,2	25,55
1,5	25,48
2	25,61
2,5	25,39

Pode-se verificar na Tabela 4.3 que o valor de PSNR é maior quanto maior é o valor máximo do contraste. Considerando *Fischer* [3] e os resultados da Tabela 4.3, é utilizado nos testes deste trabalho o valor máximo do contraste igual 1,2.

Neste trabalho os parâmetros de entrada que são alterados a fim de alterar a taxa de bits da imagem comprimida são: o tamanho dos *range blocks* e a sobreposição na construção do *Domain Pool*. De acordo com *Fischer* [3]

Para demonstrar a etapa de descompressão a Figura 4.5 mostra as 5 primeiras iterações do processo de descompressão com os respectivos valores de PSNR para cada iteração. A imagem inicial utilizada na descompressão foi denominada neste trabalho de “menina”. Nesta imagem foram aplicados os coeficientes do código fractal referente à imagem da “Lena” 256 x 256 pixels, obtidos pela compressão fractal utilizando o método da força bruta. Na construção do *Domain Pool* não foi utilizada sobreposição e o tamanho dos *range blocks* é de 4×4 ($n = 4$). A Figura 4.6 mostra as 5 primeiras iterações do processo de descompressão com os respectivos valores de PSNR para cada iteração, alterando-se o parâmetro de entrada sobreposição na construção do *Domain Pool* de 0% para 75%.



Imagem inicial



PSNR = 14,58 dB



PSNR = 18,49 dB



PSNR = 22,26 dB



PSNR = 28,69 dB



PSNR = 31,59 dB

Figura 4.5 – Imagem inicial e as 5 primeiras iterações do processo de descompressão de método força bruta.



Imagem inicial



PSNR = 14,70 dB



PSNR = 19,17 dB



PSNR = 24,62 dB



PSNR = 29,74 dB



PSNR = 32,35 dB

Figura 4.6 – Imagem inicial e as 5 primeiras iterações do processo de descompressão do método força bruta.

4.3 Método da Partição Quadtree

O inconveniente do método da força bruta na partição da imagem em *range blocks* e em *domain blocks* homogêneos (tamanho dos blocos constante) é a quantidade de comparações entre cada um dos *range blocks* e todos os possíveis *domain blocks* transformados que formam o *Domain Pool*. A busca de cada *range block* pelo melhor *domain block* torna-se exaustiva e dispende um tempo muito grande na etapa de codificação.

Por exemplo, para uma imagem $m \times m$, com $m = 64$, ou seja, 64×64 pixels, considerando os *range blocks* de tamanho 4×4 , tem-se um *Range Pool* com 256 *range bocks* $\left(\frac{m}{n} * \frac{m}{n}\right)$. Considerando os *domain blocks* tamanho 8×8 e que o *Domain Pool* é montado utilizando sobreposição de 50%, tem-se um total de 225 *domain blocks*. $\left(\left(\frac{m-2n}{2n*0,5}\right) + 1\right)^2$. Se para cada *domain block* são possíveis 8 transformações para depois serem comparados com cada *range block* ($8 * 225 = 1800$), cada um dos 256 *range blocks* é comparado com 1800 *domain blocks*.

Muitas imagens possuem alta diversidade em relação a riqueza de detalhes e existem áreas nestas imagens que utilizando-se blocos menores consegue-se um bom resultado. A partição Quadtree divide a imagem de forma não homogênea com blocos menores nas áreas com detalhes finos e com blocos maiores nas áreas mais uniformes da imagem [12, 17].

A partição quadtree gera os *range blocks* de tamanhos diferentes conforme a riqueza de detalhes de cada parte da imagem. Ela representa a imagem como uma árvore, onde a cada nível a imagem é dividida em quatro sub-imagens. Estas sub-imagens também podem ser divididas em quatro outras sub-imagens e assim sucessivamente, como mostra a Figura 4.7. O que determina se a sub-imagem será novamente subdivida ou não é um limiar de erro pré-estabelecido entre o *range block* em questão (que é a própria sub-imagem) e os *domain blocks* possíveis para aquele *range block*.

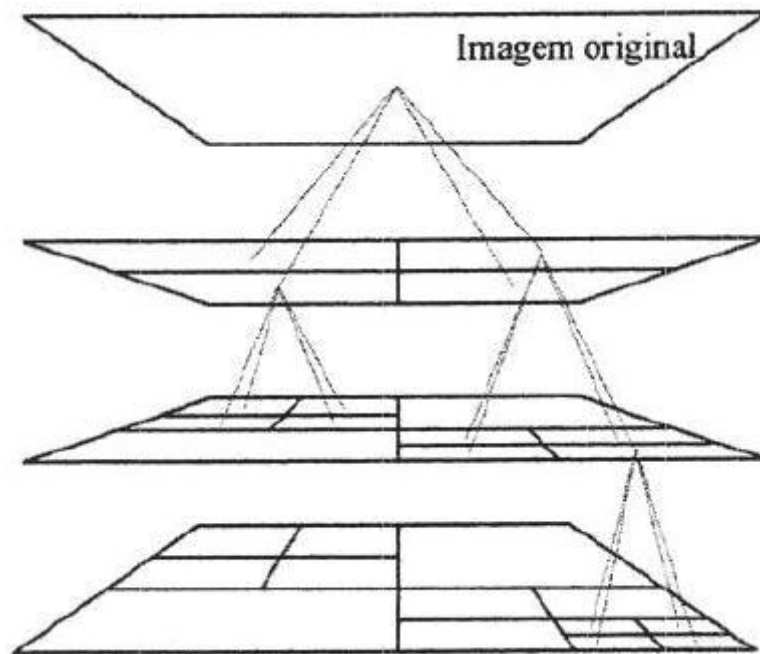


Figura 4.7 – Representação em árvore da partição Quadtree [3].

Primeiramente a imagem inicial é dividida em quatro sub-imagens que são os quatro *range blocks* iniciais. Cada um desses quatro *range blocks* é analisado com a finalidade de encontrar o *domain block* que possui o melhor *matching*. Essa primeira divisão da imagem é chamada de nível 1 de partição. Para cada nível de partição tem-se um tamanho de *range block*, e então é construído o *Domain Pool* contendo *domain blocks* com o dobro do tamanho dos *range blocks*.

Cada *range block* do nível 1 de partição é comparado com cada *domain block* transformado do atual *Domain Pool*. Encontrado o melhor *matching* que gera o menor erro RMS, este é comparado com o limiar de erro RMS pré-estabelecido. Se o menor erro RMS é menor que o limiar, os coeficientes para o *range block* em questão são armazenados. Se é maior, o *range block* em questão é dividido em 4 sub-imagens que serão os novos *range blocks* do nível 2 de partição e assim por diante até cobrir a imagem toda.

O limiar de erro RMS é um parâmetro de entrada da compressão pelo método da partição quadtree, assim como o nível máximo de partição. Como mencionado neste trabalho, a primeira divisão da imagem é chamada de nível 1 de partição, na próxima subdivisão ela é chamada de nível 2 de partição e assim sucessivamente. Quando o valor máximo do nível de partição é atingido as sub-imagens não poderão mais ser subdivididas.

Portanto, a partição dos *range blocks* deve levar em conta tanto o limiar de erro RMS quanto o nível máximo de partição. Então, os *range blocks* são subdivididos enquanto o menor RMS encontrado é maior do que o limiar pré-estabelecido ou que a profundidade da partição é menor ou igual a máxima pré-estabelecida. Quando o menor RMS encontrado para um determinado *range block* é menor do que o limiar, e/ou a profundidade da partição é menor ou igual o limiar, a posição do *domain block* que gerou esse menor RMS é armazenada, assim como a transformação afim, os parâmetros s (contraste) e o (brilho) relativos ao *domain block* que gerou o menor erro RMS e o nível atual de partição. Esses coeficientes armazenados são o CÓDIGO FRACTAL para a imagem original de entrada. A Figura 4.8 mostra o esquema de compressão fractal pelo método da partição quadtree.

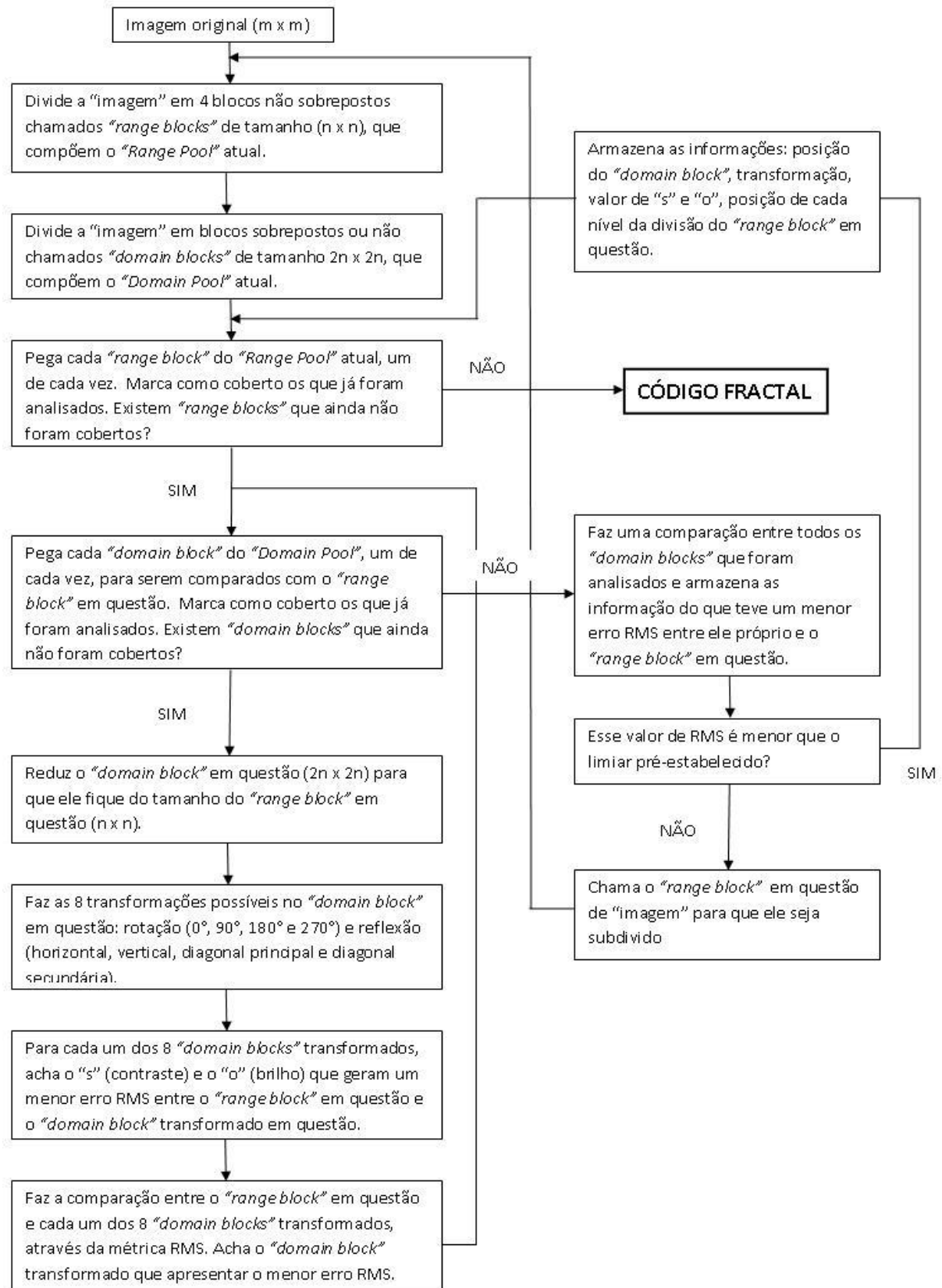


Figura 4.8 – Esquema de compressão fractal pelo método quadtree.

Utilizando a partição quadtree, as partes uniformes da imagem são cobertas por blocos maiores e as partes da imagem que possuem riqueza de detalhes maior são cobertas por blocos menores. As partes da imagem onde foram utilizados blocos maiores fazem com que a taxa de bits diminua, pois o número de blocos de comparação é menor. E as partes da imagem onde foram utilizados blocos menores fazem com que a qualidade da imagem reconstruída seja melhor, pois o número de blocos de comparação é maior.

A principal característica da partição quadtree é que nas partes da imagem que tem uma grande riqueza de detalhes, o nível de partição utilizado é maior, utilizando assim blocos de comparação menores, fazendo com que a qualidade da imagem nessas áreas seja melhor. Nas partes da imagem que são uniformes, sem muitos detalhes, o nível de partição utilizado é menor, utilizando blocos de comparação maiores, já que nessas áreas pode-se conseguir uma boa qualidade de reconstrução sem a necessidade de utilizar blocos muito pequenos. Dessa maneira consegue-se diminuir a taxa de bits e o tempo de compressão.

Os parâmetros de entrada para a compressão fractal de uma imagem baseada no método da partição quadtree são:

- Sobreposição;
- Limiar de erro (erro máximo permitido);
- Nível máximo de partição;
- Máximo valor para o contraste;
- Quantidade de bits para representar o contraste;
- Quantidade de bits para representar o brilho.

Os parâmetros “sobreposição”, “máximo valor do contraste”, “quantidade de bits para representar o contraste” e “quantidade de bits para representar o brilho” afetam a qualidade da imagem comprimida pelo método da partição quadtree da mesma maneira que no método da força bruta.

O parâmetro “limiar de erro” tem uma ligação direta com o nível de partição. Quando cada *range block* é comparado com todos os possíveis *domain blocks*, e o erro entre eles é maior do que o limiar de erro pré-estabelecido, esse *range block* em questão é subdividido em 4, passando para o próximo nível de partição.

O parâmetro “nível máximo de partição” depende do tamanho da imagem original que se deseja comprimir. Por exemplo, uma imagem de 64 x 64 pixels, normalmente usa-se como nível máximo de partição o valor 4, pois no nível 1 tem-se os *range blocks* de tamanho 32 x 32, no nível 2 tem-se *range blocks* de tamanho 16 x 16, no nível 3 tem-se *range blocks* de tamanho 8 x 8 e no nível 4 tem-se *range blocks* de tamanho 4 x 4.

Assim como no método da força bruta, ao final da comparação de cada *range block* com todos os possíveis *domain blocks*, e escolhido o que gerou o menor erro RMS, armazenam-se os coeficientes que definem o *domain block* que gerou esse menor erro. O conjunto de coeficientes armazenados referentes à imagem original de entrada pela compressão de imagem utilizando o método da partição quadtree, que é o próprio código fractal referente à imagem de entrada é composto pelos seguintes coeficientes:

- Posição do *domain block* (que gerou o menor RMS);
- A transformação geométrica do *domain block* (que gerou o menor RMS);
- O valor de *s* (ajuste de contraste);
- O valor de *o* (ajuste de brilho).
- O nível atual de partição.

O coeficiente “posição do *domain block*” indica a posição, dentro da imagem original, do *domain block* que teve o menor erro RMS. O coeficiente “transformação” indica qual a transformação realizada no *domain block* que gerou esse menor erro RMS. Os coeficientes “valor de contraste” e “valor de brilho” indicam os valores ótimos de ajuste de contraste e brilho que deverão ser feitos para aquele dado *range block*. O coeficiente “nível atual de partição” indica qual o nível de partição no momento da escolha do melhor *matching*, podendo ser extraído desse coeficiente também o tamanho do *range block* e consequentemente o tamanho do *domain block*.

Assim como no método da força bruta, cada coeficiente necessita de uma quantidade de bits específica para ser armazenado. A quantidade de bits utilizados para quantizar os valores ótimos de *s* (contraste) e *o* (brilho) é um parâmetro de entrada. A escolha desses valores é realizada em cima do estudo dos testes realizados que serão mostrados nesse capítulo.

A quantidade de bits utilizada para armazenar o coeficiente “posição do *domain block*” depende do máximo nível de partição, pois é ele que define quantos *domain blocks* fazem parte do *Domain Pool*. O coeficiente “transformação geométrica” é representado utilizando 3 bits, pois são possíveis apenas 8 transformações. A quantidade de bits utilizada para a representação do coeficiente “nível atual de partição” depende do parâmetro de entrada “nível máximo de partição”.

No momento do cálculo do erro RMS entre cada *range block* e cada *domain block* em questão para encontrar o melhor *matching*, os valores dos coeficientes “contraste” e “brilho” que são utilizados são os valores quantizados, pois são esses valores que são utilizados na etapa da reconstrução.

4.3.1 Classificação de Média e Variância

A fase de comparação entre cada *range block* e todos os *domain blocks* possíveis é bastante demorada em função da quantidade de blocos a serem comparados, principalmente se o número de *domain blocks* é maior dependendo da “sobreposição” escolhida na construção do *Domain Pool*. Para acelerar essa etapa de comparação, a idéia proposta por Fischer [3], é classificar os *range blocks* e os *domain blocks* utilizando a média e a variância dos blocos. A partir dessa classificação, são comparados apenas os *range blocks* e os *domain blocks* que estiverem dentro de uma mesma classe [3, 18].

A classificação dos *range blocks* e dos *domain blocks* é realizada antes da etapa de codificação começar, e durante a etapa de codificação a classe do *range block* em questão e a classe de cada *domain block* é analisada, e somente se pertencerem a mesma classe é que eles são comparados. Caso contrário, passa-se para o próximo *domain block* do *Domain Pool*.

A classificação dos blocos é realizada dividindo-os em quatro sub-blocos. Esses sub-blocos são numerados da seguinte maneira: os sub-blocos superiores da esquerda e da direita são os sub-blocos de número 1 e 2, respectivamente e os sub-blocos inferiores da esquerda e da direita são os sub-blocos de número 3 e 4, respectivamente. A Figura 4.9 mostra a divisão dos blocos. A média e a variância são calculadas para cada sub-bloco.

1	2
3	4

Figura 4.9 – Numeração dos blocos para classificação [3].

Fischer dividiu a classificação em superclasses e subclasses. As superclasses são baseadas na classificação da média dos 4 quadrantes e as subclasses são baseadas na classificação da variância dos 4 quadrantes.

Considerando $(r_1^i, r_2^i, \dots, r_n^i)$ a intensidade de cada um dos n pixels, de cada um dos i quadrantes ($i = 1, 2, 3, 4$), a média de cada quadrante e a variância são calculadas utilizando as Equações (4.4) e (4.5) [3].

$$M_i = \frac{1}{n} \sum_{j=1}^n r_j^i \quad (4.4)$$

$$V_i = \frac{1}{n} \left(\sum_{j=1}^n r_j^{i^2} - M_i^2 \right) \quad (4.5)$$

onde:

M – média;

V – variância;

i – quadrante em questão;

j – pixel em questão;

n – quantidade de pixels do quadrante em questão.

Fischer classificou os blocos em 3 superclasses baseadas na média de cada quadrante do bloco como mostrado nas Equações (4.6) a (4.8).

$$\text{Superclasse 1: } M_1 > M_2 > M_3 > M_4 \quad (4.6)$$

$$\text{Superclasse 2: } M_1 > M_2 > M_4 > M_3 \quad (4.7)$$

$$\text{Superclasse 3: } M_1 > M_4 > M_2 > M_3 \quad (4.8)$$

A Figura 4.10 mostra as 3 superclasses classificadas por *Fischer*.

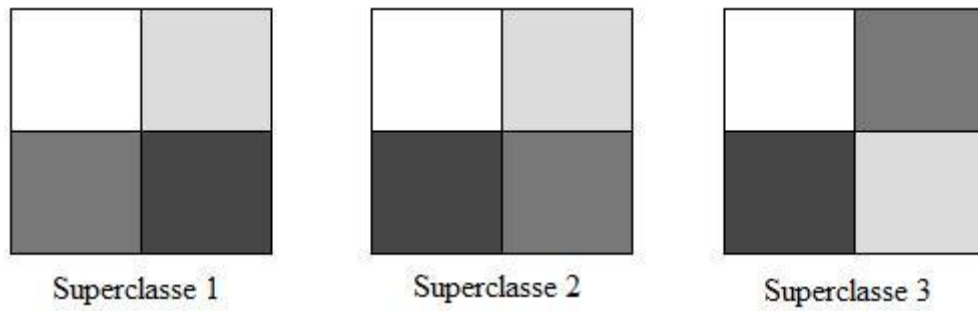


Figura 4.10 – Superclasses 1, 2 e 3 [3].

Dentro de cada uma das 3 superclasse, são realizadas as 8 transformações isométricas e cada bloco resultante é classificado como sendo de uma determinada superclasse resultando em 24 ordenações possíveis de média classificadas em apenas 3 superclasses [3]. A Figura 4.11 mostra todas as possíveis transformações que são classificadas em cada uma das 3 superclasses.

Levando em consideração a variância, *Fischer* [3] classificou em 24 subclasses que representam as 24 ordenações possíveis do valor da variância para cada quadrante (M_i) totalizando 72 classes.

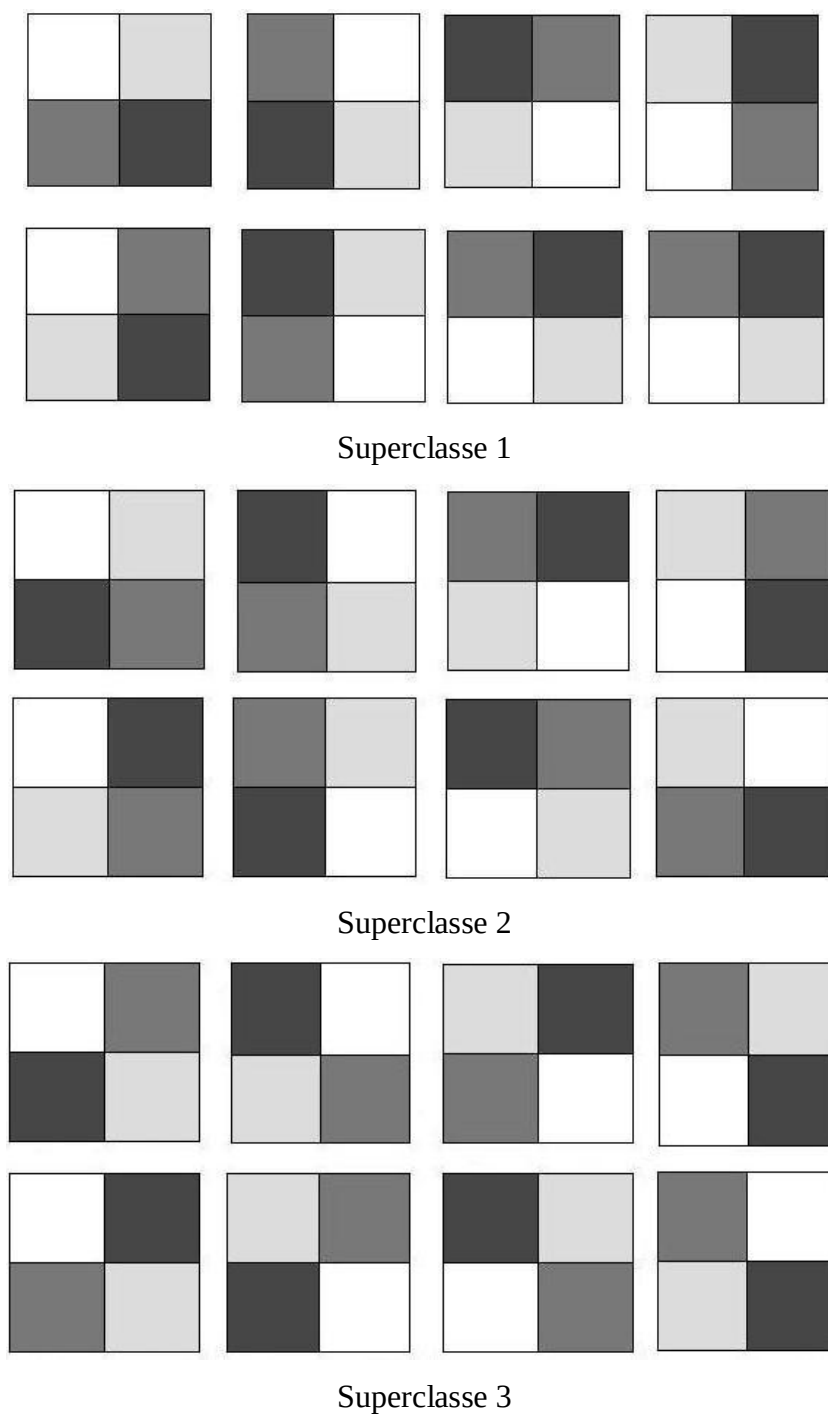


Figura 4.11 – 8 transformações que são classificadas dentro de cada superclasse.

No momento da classificação em relação a ordenação da média de cada bloco, deve-se fazer uma das 8 transformações isométricas possíveis, com a finalidade de classificar um determinado bloco em uma das 3 superclasses existentes. O que é proposto neste trabalho é a utilização dessa transformação no momento da comparação de um determinado *range block*

com um determinado *domain block*. Se as transformações necessárias para incluir um determinado *range block* e um determinado *domain block* em umas das 3 superclasses são armazenadas, pode-se fazer um estudo entre essas duas transformações e descobrir qual transformação é necessária fazer no *domain block* para que ele fique com a mesma ordenação de média do *range block*. Descobrendo-se essa transformação, no momento da comparação não é necessário a comparação do *range block* com todas as 8 transformações do *domain block*. É necessário apenas comparar com o *domain block* transformado utilizando aquela transformação da classificação de média. Com isso, diminuindo-se o número de comparações e também o tempo de compressão.

Assim como no método da força bruta, para obter a imagem reconstruída a partir dos coeficientes do código fractal, é necessário aplicá-los em uma imagem inicial qualquer e então é reconstruída uma aproximação da imagem original. Esta aproximação é relacionada com o atrator da teoria dos fractais. A qualidade da imagem depende principalmente do nível máximo de partição, do limiar de erro e da sobreposição. A qualidade da imagem final reconstruída depende também do número de iterações aplicadas à imagem original qualquer. Normalmente, o valor de PSNR é estabilizado a partir da 10ª iteração.

4.3.2 Testes e resultados utilizando o método da partição quadtree

Foi desenvolvido neste trabalho um algoritmo em Matlab, baseado no método da partição quadtree utilizando a classificação de média e variância. Foram realizados testes variando a sobreposição na construção do *Domain Pool*, o limiar de erro, o nível máximo de partição, o máximo valor do contraste e o número de bits utilizados na quantização dos coeficientes contraste e brilho. Para os testes, foi utilizada uma imagem da “Lena” 64 x 64 pixels como imagem original a ser comprimida pelo método da partição quadtree. A seguir descreve-se como a variação de cada um desses parâmetros de entrada influencia a qualidade da imagem comprimida.

Variação da sobreposição

Assim como foi visto no método da força bruta, a variação da sobreposição da etapa de compressão influencia tanto a qualidade da imagem reconstruída quanto o tempo de compressão fractal. Quanto maior é a sobreposição, maior é a PSNR da imagem reconstruída. Por outro lado, quanto maior é a sobreposição, maior é o número de *domain blocks* que fazem parte do *Domain Pool*, fazendo com que a etapa de codificação seja mais demorada. Para os testes não é utilizado sobreposição na construção do *Domain Pool*.

Variação do limiar de erro

O limiar de erro está diretamente relacionado ao nível de partição e consequentemente ao tamanho dos blocos, pois é esse limiar que determina se um *range block* deve ou não ser particionado novamente. A Tabela 4.4 mostra os valores de PSNR para a imagem reconstruída, variando o limiar de erro. Os outros parâmetros de entrada utilizados foram: nível máximo de partição = 4, sobreposição = 0, quantidade de bits para representar o contraste = 5, quantidade de bits para representar o brilho = 7 e valor máximo do contraste = 1,2.

Pode-se verificar na Tabela 4.4 que quanto menor é esse limiar, melhor será a qualidade da imagem reconstruída, porém mais blocos são particionados com níveis mais altos de partição, fazendo com que a taxa de bits (tamanho do código fractal) seja maior.

Tabela 4.4 – Valores de PSNR variando o limiar de erro.

Limiar erro	PSNR (dB)
20	25,50
50	25,49
70	25,47
100	25,35
150	24,48
200	22,74

Considerando os resultados obtidos na Tabela 4.4, é utilizado nos testes deste trabalho o valor de limiar de 70, onde tem-se um equilíbrio entre a qualidade da imagem reconstruída e o tempo de compressão.

Variação do número de bits utilizados na quantização dos valores de “contraste” e “brilho”

Os coeficientes do código fractal, valor de s (contraste) e o valor de o (brilho), devem ser quantizados com a finalidade de diminuir a taxa de bits da imagem comprimida. Foram realizados testes utilizando diferentes quantidades de bits para representar os coeficientes contraste (s) e brilho (o). Primeiro fixou-se s_{bits} em 5 e variou-se o_{bits} de 5 a 9. Depois fixou-se o_{bits} em 7 e variou-se s_{bits} de 3 a 7.

A Tabela 4.5 mostra os valores de PSNR para a imagem reconstruída, variando a quantidade de bits utilizados na etapa de quantização do contraste e do brilho. Os outros parâmetros de entrada utilizados foram: nível máximo de partição = 4, sobreposição = 0, limiar de erro = 70 e valor máximo do contraste = 1,2.

Tabela 4.5 – Valores de PSNR variando a quantidade de bits para representar o contraste (s_{bits}) e o brilho (o_{bits}).

s_{bits}	o_{bits}	PSNR (dB)
5	5	23,66
5	6	24,50
5	7	25,47
5	8	25,89
5	9	26,04
3	7	24,97
4	7	25,31
5	7	25,47
6	7	25,43
7	7	25,40

Ao analisar a Tabela 4.5 e com base em *Fischer* [3], assim como no método da força bruta a quantidade escolhida de bits para representar o contraste e o brilho para os testes neste trabalho foi de 5 e 7, respectivamente.

Variação do valor máximo do contraste

Para o teste variando o valor máximo do contraste ($s_{\max}$), considerou-se o intervalo os valores positivos de contraste ($s > 0$) variando apenas o máximo valor do contraste.

A Tabela 4.6 mostra os valores de PSNR para a imagem reconstruída, variando o valor máximo do contraste. Os outros parâmetros de entrada utilizados foram: nível máximo de partição = 4, sobreposição = 0, limiar de erro = 70, quantidade de bits para representar o contraste = 5 e quantidade de bits para representar o brilho = 7.

Tabela 4.6 – Valores de PSNR variando o valor máximo do contraste ($s_{\max}$).

$s_{\max}$	PSNR (dB)
0,5	23.4072
1	25.4346
1,2	25.4754
1,5	25.2796
2	25.3840
2,5	25.0832

Ao analisar a Tabela 4.6 e de acordo com *Fischer* [3], assim como no método da força bruta o valor máximo do contraste escolhido para os testes neste trabalho foi de 1,2.

Variação do nível máximo de partição

Como citado anteriormente neste capítulo, a variação do nível de partição tem relação direta com o tamanho dos *range blocks*. Quanto maior é o nível de partição, menores são os *range blocks* fazendo com que as partes da imagem com riqueza de detalhes sejam melhor

reconstruídas. Por outro lado, quanto menores são os *range blocks* mais demorada é a etapa de compressão.

Para uma imagem original de 64 x 64 pixels a ser comprimida, o nível de partição 1 indica *range blocks* de 32 x 32, que são comparados com o único *domain block* existente de 64 x 64, que é a própria imagem. Se o erro RMS entre cada um desses *range blocks* e o *domain block* é maior do que o limiar de erro pré-estabelecido (parâmetro de entrada), o *range block* em questão é dividido em quatro (nível de partição 2), e agora tem-se *range blocks* de 16 x 16, que são comparados com os *domain blocks* de 32 x 32 do *Domain Pool* referente ao 2º nível de partição. Se nenhum dos *domain blocks* comparados apresentar um erro RMS menor do que o limiar, é realizada novamente a partição até que seja encontrado um *domain block* que comparado com o *range block* em questão apresente um erro RMS menor do que o limiar, ou até que o nível de partição atinja o limite máximo pré-estabelecido (parâmetro de entrada).

A escolha do nível máximo de partição deve considerar o tamanho da imagem original a ser comprimida. Para cada tamanho de imagem original a Tabela 4.7 mostra o tamanho dos *range blocks* para cada nível de partição.

Tabela 4.7 – Tamanho dos *range blocks* para cada nível de partição.

Tamanho imagem (pixels)	Nível de partição							
	1	2	3	4	5	6	7	8
64 x 64	32 x 32	16 x 16	8 x 8	4 x 4	2 x 2	1 x 1	-	-
128 x 128	64 x 64	32 x 32	16 x 16	8 x 8	4 x 4	2 x 2	1 x 1	-
256 x 256	128 x 128	64 x 64	32 x 32	16 x 16	8 x 8	4 x 4	2 x 2	1 x 1
512 x 512	256 x 256	128 x 128	64 x 64	32 x 32	16 x 16	8 x 8	4 x 4	2 x 2

Foram realizados testes variando o nível máximo de partição. A Tabela 4.8 mostra os valores de PSNR para a imagem reconstruída, variando esse parâmetro de entrada. Os outros parâmetros de entrada utilizados foram: sobreposição = 0, limiar de erro = 70, quantidade de bits para representar o contraste = 5, quantidade de bits para representar o brilho = 7 e valor máximo do contraste = 1,2.

Tabela 4.8 – Valores de PSNR variando o nível máximo de partição.

Nível max	PSNR (dB)
2	15,84
3	19,37
4	25,47
5	28,29

Ao considerar os valores de PSNR da Tabela 4.8, o nível máximo de partição escolhido para os testes para a imagem de 64 x 64 é de 4.

Para demonstrar a etapa de descompressão, a Figura 4.12 mostra as 5 primeiras iterações com os respectivos valores de PSNR para cada iteração. A imagem inicial utilizada na descompressão foi a “menina”. Nesta imagem foram aplicados os coeficientes do código fractal referente à imagem da “Lena” 256 x 256 pixels, obtidos pela compressão fractal utilizando o método da partição quadtree. Na construção do *Domain Pool* não foi utilizada sobreposição, o limiar de erro foi de 70 e o nível máximo de partição foi 6, gerando *range blocks* de até 4 x 4.

A Figura 4.13 mostra as 5 primeiras iterações do processo de descompressão com os respectivos valores de PSNR para cada iteração. A imagem inicial utilizada na descompressão foi a “menina”. Em cima dessa imagem foram aplicados os coeficientes do código fractal referente à imagem da “Lena” 256 x 256 pixels, obtidos pela compressão fractal utilizando o método da partição quadtree utilizando a idéia proposta neste trabalho de obter a transformação na etapa da classificação de média evitando a comparação entre o *range block* e todas as transformações do *domain block*. Na construção do *Domain Pool* foi utilizada a sobreposição de 50%, o limiar de erro foi de 70 e o nível máximo de partição foi 6, gerando *range blocks* de até 4 x 4.



Imagem Inicial



PSNR = 16,47 dB



PSNR = 22,18 dB



PSNR = 26,93 dB



PSNR = 28,41 dB



PSNR = 28,45 dB

Figura 4.12 – Imagem inicial e as 5 primeiras iterações do processo de descompressão.



Imagem Inicial



PSNR = 15,99 dB



PSNR = 20,68 dB



PSNR = 25,16 dB



PSNR = 27,24 dB



PSNR = 27,46 dB

Figura 4.13 – Imagem inicial e as 5 primeiras iterações do processo de descompressão.

4.4 Considerações Finais do Capítulo

Os trabalhos desenvolvidos por Fischer introduziram a teoria dos fractais no estudo da compressão de imagens. Dois métodos foram propostos por ele e a partir deles surgiram pesquisas em relação a essa técnica de compressão de imagens. Nesse capítulo foram apresentados o método da força bruta e o método da partição quadtree. Foi estudada a influência que cada parâmetro de entrada tem na qualidade da imagem reconstruída (PSNR), pelos testes realizados variando esses parâmetros de entrada. Os resultados dos testes realizados neste trabalho foram condizentes com o que é descrito por *Fischer* [3].

Com a finalidade de realizar uma comparação entre diferentes técnicas de compressão, o próximo capítulo apresenta a técnica de compressão de imagens conhecida como padrão JPEG2000.

CAPÍTULO 5

PADRÃO DE COMPRESSÃO JPEG2000

5.1 Introdução

Os membros da *International Telecommunication Union* (ITU) e da *International Organization for Standardization* (ISO) iniciaram um trabalho nos anos 80 com a finalidade de estabelecer um padrão internacional para a compressão de imagens, em níveis de cinza e imagens coloridas. O *Joint Photographic Experts Group* (JPEG) é o resultado deste trabalho entre a ITU e a ISO. O “*joint*” em JPEG refere-se a colaboração entre a ITU e a ISO. O JPEG corresponde ao padrão internacional 10928-1 da ISO/IEC ou a recomendação T.81 da ITU-T. O texto em ambos os documentos são exatamente iguais [19].

O JPEG2000 foi um padrão que começou a ser desenvolvido em 1997 com objetivo de codificar diferentes tipos de imagens, com diferentes características, permitindo diferentes modelos de imageamento, preferencialmente com um sistema unificado. Porém, esse novo padrão foi produzido realmente em 2000. O padrão JPEG2000 define um novo esquema de compressão de imagens baseado na transformada *wavelet* e pode ser utilizado em diversas aplicações [19].

A principal diferença entre o padrão JPEG e o padrão JPEG2000, é que no JPEG a transformada utilizada no processo de compressão é a Transformada Discreta do Cosseno, do inglês *Discrete Cosine Transform* (DCT), enquanto que no JPEG2000 é utilizada a Transformada Discreta *Wavelet*, do inglês *Discrete Wavelet Transform* (DWT).

Este capítulo apresenta uma introdução em relação a transformada *wavelet* e em seguida descreve o padrão de compressão JPEG2000 baseado nessa transformada.

5.2 Transformada *Wavelet*

Análise *wavelet* é um conjunto de ferramentas e técnicas para a análise de diversos sinais, inclusive imagens, pois as *wavelets* podem ser utilizadas em sinais bidimensionais. A análise *wavelet* permite o uso de longos intervalos de tempo para extrair informações mais precisas de baixas frequências e regiões menores (intervalos pequenos de tempo) para informações de alta frequência. Dessa forma, a análise *wavelet* tem a vantagem de analisar áreas localizadas de um sinal grande e é capaz de mostrar aspectos de dados que outras técnicas de análise de sinal omitem, tais como inclinações, descontinuidades em altas derivadas, pontos de quebra, entre outros [20].

Wavelets são ondas pequenas, de frequência variável e duração limitada. A análise *wavelet* consiste em representar um sinal em versões escaladas (comprimidas ou expandidas) e deslocadas (atrasadas ou adiantadas) da *wavelet* original. Enquanto a análise de Fourier consiste em quebrar um sinal em senos de várias frequências [20]. A Figura 5.1 mostra uma onda seno, utilizada na análise Fourier e uma onda *wavelet* utilizada em análise *wavelet*.

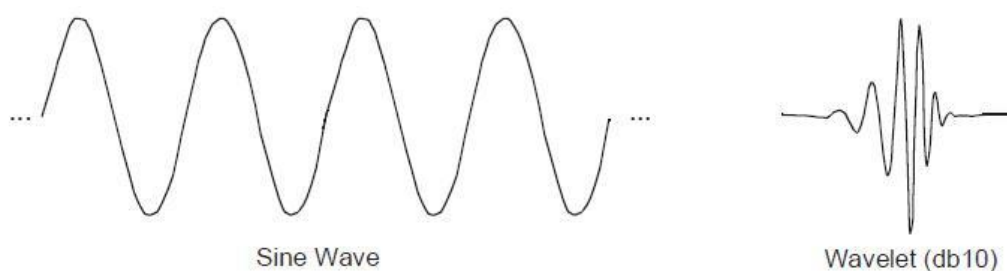


Figura 5.1 – Onda seno (análise fourier) e onda *wavelet* (análise *wavelet*) [20].

Representando o sinal em diferentes versões da *wavelet* original, as características locais dele podem ser descritas de uma melhor maneira. Na análise Fourier a informação

temporal é perdida ficando apenas a informação de frequência, enquanto que na análise *wavelet* tem-se tanto a informação em frequência quanto a informação temporal.

A transformada *wavelet* contínua, do inglês *Continuous Wavelet Transform (CWT)* é definida como a soma em todo o tempo do sinal multiplicada pela versão da função *wavelet*. O resultado da CWT são vários coeficientes *wavelet* (C) que são funções do escalar e da posição, como mostrado na Equação (5.1). Multiplicando cada coeficiente pela *wavelet* escalada e deslocada apropriada, as *wavelets* constituintes do sinal original são obtidas [20].

$$C(\text{escalar}, \text{posição}) = \int_{-\infty}^{\infty} f(t) \Psi(\text{escalar}, \text{posição}, t) dt \quad (5.1)$$

Passos para obter a CWT [20]:

- Passo 1: A *wavelet* é comparada com a seção inicial do sinal original;
- Passo 2: É calculado o coeficiente C que representa a correlação da *wavelet* com o sinal original, como mostrado na Figura 5.2. Quanto maior é o número C, mais similar é a *wavelet* do sinal.

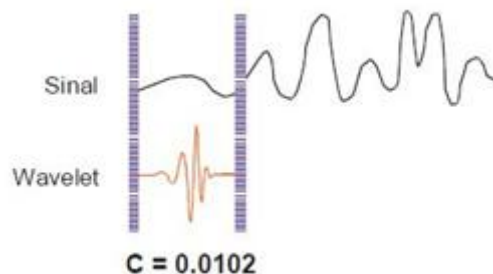


Figura 5.2 – Comparação da primeira parte do sinal com a *wavelet* [20].

- Passo 3: A *wavelet* é deslocada e o passo 1 e 2 são repetidos até a *wavelet* percorrer todo o sinal, como mostra a Figura 5.3;

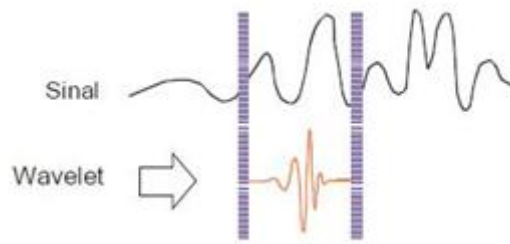


Figura 5.3 – Deslocamento da *wavelet* e comparação com a próxima parte do sinal [20].

- Passo 4: A *wavelet* é expandida e os passos de 1 a 3 são repetidos até percorrer todo o sinal, como mostra a Figura 5.4;

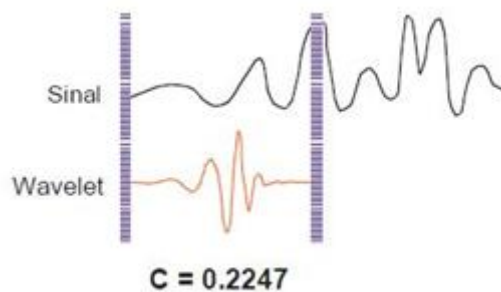


Figura 5.4 – Expansão da *wavelet* [20].

- Passo 5: Os passos de 1 a 4 são repetidos para todas as escalas.

Cada porção do sinal é comparada com as *wavelets* das várias escalas (comprimidas ou expandidas). Para cada comparação, é gerado um coeficiente que mede a correlação da porção do sinal original com a escala da *wavelet* atual. Quanto maior é a escala da *wavelet*, ou seja, mais expandida é a *wavelet* utilizada, maior é a porção do sinal com o qual ela é comparada, como mostra a Figura 5.5.



Figura 5.5 – Diferentes escalas da *wavelet* [20].

Os coeficientes constituem o resultado da regressão do sinal original realizada pelas *wavelets* [20]. A Figura 5.6 mostra os coeficientes resultantes da passagem de *wavelets* por um sinal qualquer.

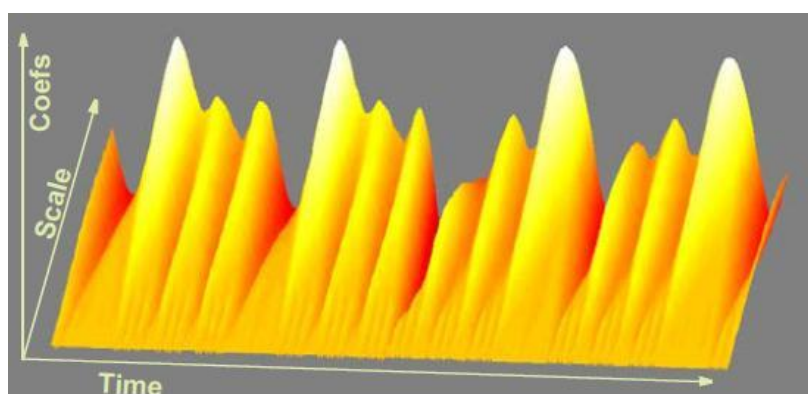


Figura 5.6 – Coeficientes gerados pela passagem de *wavelets* por um sinal qualquer [20].

Em análise *wavelet*, as aproximações são as altas escalas, ou seja, as componentes de baixa frequência do sinal e os detalhes são as baixas escalas que correspondem as componentes de alta frequência do sinal.

A DWT utiliza escala e posições baseadas em potência de dois para que a análise seja mais eficiente e mais rápida. Um caminho eficiente de implementar a DWT é a utilização de filtros. Este esquema desenvolvido em 1988 por Mallat [20].

A DWT é aplicada no sinal de entrada por meio de um conjunto de filtros, passa-baixa e passa-alta. Ao passar pelo conjunto de filtros o sinal original se transforma em dois novos sinais que são sub-amostrados e cada um contém metade das informações do sinal original. O

novo sinal que contém informações de aproximação é originado da passagem do sinal original pelo filtro passa-baixa e o novo sinal contendo informações de detalhes é originado da passagem do sinal original pelo filtro passa-alta. A Figura 5.7 mostra um sinal original (S) passando pelo conjunto de filtros e os dois novos sinais sub-amostrados. Um dos sinais sub-amostrados contém informações de detalhes (cD) e o outro contém informações de aproximação (cA) [20].

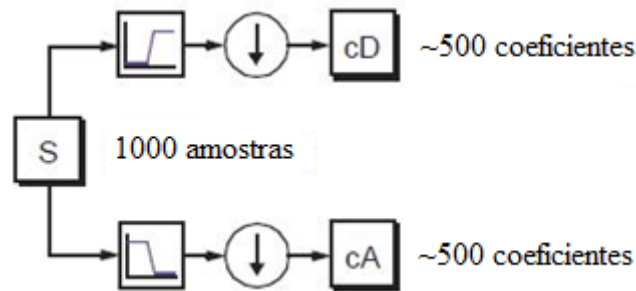


Figura 5.7 – Conjunto de filtros, passa-baixa e passa-alta [20].

5.3 Padrão de Compressão JPEG2000

No padrão de compressão JPEG2000, a DWT é aplicada na imagem de entrada por meio de um conjunto de filtros *wavelet*, com a finalidade de reduzir a quantidade de informação contida na imagem.

A imagem inicial ou parte dela passa por dois filtros, um passa-baixa e um passa-alta. Esses filtros são aplicados em uma dimensão arbitrária, que pode ser horizontal pelas colunas da imagem, ou vertical, pelas linhas da imagem. O resultado dessa filtragem são duas novas imagens que são sub-amostradas, ficando cada uma delas com a metade da quantidade de informação contida na imagem original. Cada uma dessas duas novas imagens passa novamente pelos dois filtros, agora na dimensão contrária e o resultado é novamente sub-amostrado [20, 21].

O resultado da filtragem pelos dois filtros nas duas dimensões são quatro novas imagens, onde cada uma tem um quarto da quantidade de informação contida na imagem original. Esses quatro conjuntos de dados são armazenados nos quatro blocos que fazem parte do primeiro nível de transformação, e são denominados como: LL, HL, LH e HH, onde L

representa “baixa”, do filtro passa-baixa, e H representa “alta”, do filtro passa-alta. A Figura 5.8 mostra passo a passo a aplicação do conjunto de filtros em um nível de transformação [20, 21, 22].

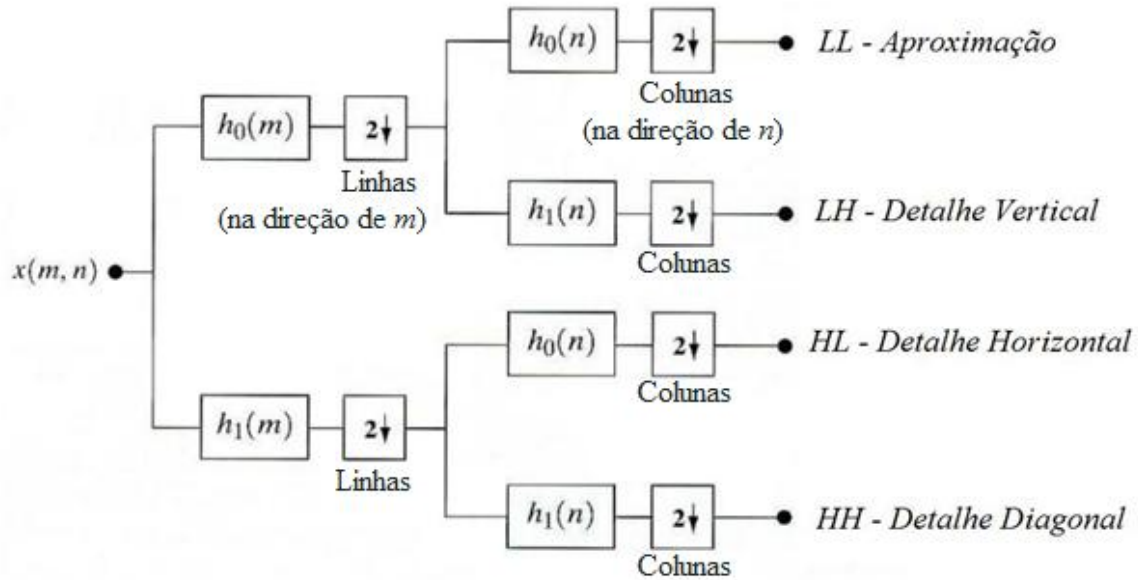


Figura 5.8 – Aplicação do conjunto de filtros em um nível de transformação [22].

O bloco LL é obtido pela aplicação do filtro passa-baixa nas duas dimensões e o bloco HH pela aplicação do filtro passa-alta nas duas dimensões. Os blocos LH e HL são obtidos pela aplicação do filtro passa-alta em uma dimensão e do filtro passa-baixa na outra dimensão.

Os blocos HL e LH são denominados também de detalhes horizontais ou verticais e o bloco HH é também denominado de detalhe diagonal. O bloco LL é o que contém a maior parte da energia e consequentemente a maior parte da informação da imagem inicial. Este bloco denominado de aproximação e é novamente submetido ao conjunto de filtros, gerando o próximo nível de transformação e obtendo-se novas subbandas [12, 21]. A Figura 5.9 mostra os três primeiros níveis de transformação.

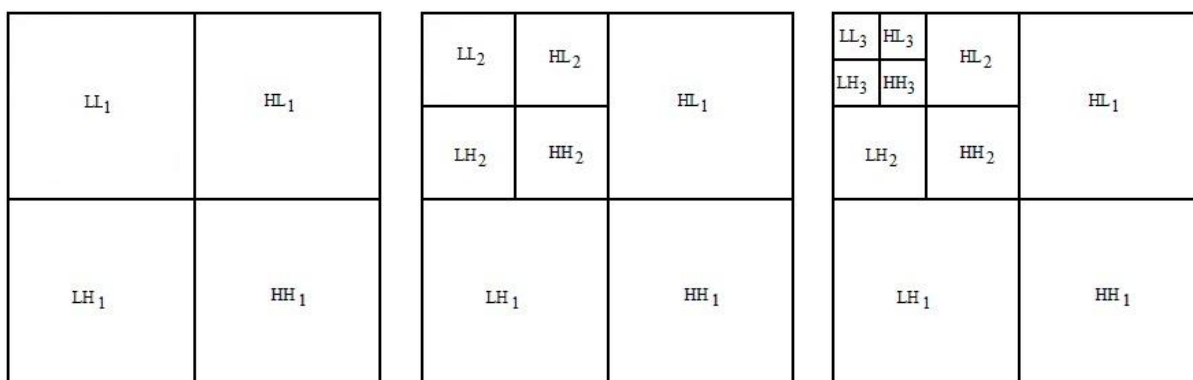


Figura 5.9 – 1º, 2º e 3º níveis de transformação [21].

Cada camada da subbanda suporta uma taxa de compressão particular, onde as primeiras camadas fornecem altas taxas de compressão e baixa qualidade de reconstrução da imagem. As últimas camadas contêm as frequências mais altas e suportam taxas de compressão mais baixas fornecendo uma melhor qualidade, pois essas áreas são as que possuem maior nível de detalhes.

Os coeficientes são quantizados e codificados para formar o *codestream* (*bitstream*) de saída. A imagem original pode ser particionada em blocos não-sobrepostos que são comprimidos separadamente.

No decodificador o *codestream* é decodificado, desquantizado e é então aplicada a transformada discreta *wavelet* inversa, resultando na imagem reconstruída. O JPEG2000 é um padrão de compressão que pode ser tanto com perdas quanto sem perdas, dependendo da transformada *wavelet* e da quantização aplicada.

5.3.1 Resultados da Compressão JPEG2000

O Projeto "*Jasper*" é um *software* de compressão baseado no padrão JPEG2000. Este projeto é coordenado pelo seu principal autor, *Michael Adams*, que é afiliado com o Grupo de Processamento Digital de Sinais, do inglês *Digital Signal Processing Group* (DSPG), no Departamento de Engenharia Elétrica e de Computação da Universidade de Victoria (British Columbia - Canadá) [28].

Utilizando o software “*Jasper*”, a imagem da “Lena” 256 x 256 pixels foi comprimida utilizando diferentes taxas de compressão. A Figura 5.10 mostra as imagens comprimidas com diferentes taxas de compressão, bem como a taxa de bits e o valor de PSNR de cada uma.

A compressão JPEG2000 é uma técnica bastante utilizada e consegue atingir altas taxas de compressão com uma boa qualidade da imagem comprimida, conforme as imagens mostradas na Figura 5.10.



8 bpp - Imagem original



0,48 bpp - PSNR = 49,64 dB



0,19 bpp - PSNR = 39,84 dB



0,1 bpp - PSNR = 34,79 dB



0,07 bpp - PSNR = 31,88 dB



0,05 bpp - PSNR = 30,13 dB

Figura 5.10 – Imagem inicial e 5 taxas de compressão JPEG2000.

5.4 Considerações Finais do Capítulo

A transformada *wavelet* é a base do padrão de compressão JPEG2000. Assim como a DWT é aplicada a um sinal de entrada por meio de um conjunto de filtros, a compressão de uma imagem utilizando o padrão JPEG2000 é realizada por meio de um conjunto de filtros *wavelet*, com a finalidade de reduzir a quantidade de informação contida na imagem.

Este capítulo apresentou a compressão JPEG2000, que é baseada na transformada *wavelet*. Utilizando o software "*Jasper*", a imagem da "Lena" foi comprimida utilizando diferentes taxas de compressão.

No próximo capítulo são realizados testes de compressão em imagens médicas utilizando duas técnicas de compressão: a compressão fractal de imagens e a compressão JPEG2000.

CAPÍTULO 6

RESULTADOS OBTIDOS

6.1 Introdução

Apesar do esforço do poder público e de empresas privadas no combate ao câncer de mama, ele continua sendo uma das principais causas de morte entre mulheres, tanto no Brasil como em diversos países. O auto-exame das mamas, o exame clínico e a mamografia são as principais formas de detecção precoce. “A mamografia é a radiografia simples das mamas, sendo considerado o método mais importante para o rastreamento do câncer neste órgão, por sua eficácia diagnóstica, praticidade e principalmente por ser de baixo custo.” [23]

A quantidade de informação gerada pelas imagens mamográficas pode ser um empecilho para algumas técnicas utilizadas em medicina como, por exemplo, a telemedicina ou mesmo no armazenamento dessas imagens. As técnicas de compressão de imagens desempenham um papel importante no processo de redução da quantidade de informação para fins de armazenamento, processamento e transmissão [24].

Com a finalidade de avaliar o desempenho da compressão fractal aplicada a imagens médicas, algumas imagens mamográficas retiradas de um banco de imagens são utilizadas para testes de compressão utilizando a compressão fractal. Com o objetivo de comparar o método de compressão fractal, as mesmas imagens são comprimidas utilizando o método de compressão JPEG2000.

6.2 Imagens mamográficas

O rastreamento mamográfico consiste tipicamente em obter duas vistas da mama, que são denominadas incidências básicas. Uma das incidências básicas é a vista superior da mama denominada vista crânio caudal (CC) e a outra incidência básica é uma vista angular da mama denominada vista médio lateral oblíqua (MLO). A vista MLO é obtida de um ponto de vista angular do centro da mama para a lateral. Normalmente, é utilizado um ângulo de 90°, pois uma maior parte do tecido mamário pode ser trabalhado no quadrante superior externo da mama e da axila. A vista CC é uma vista superior da mama e é obtida de cima para baixo [25]. A Figura 6.1 mostra as duas vistas de uma mesma mama.

Para estudar regiões específicas, as incidências complementares esclarecem situações detectadas nas incidências básicas. As incidências complementares são: crânio-caudal forçada (XCC), *cleavage* (CV), médio-lateral ou perfil externo (ML ou P), lateromedial ou perfil interno (LM ou *contact*) e caudocranial (RCC) [26].

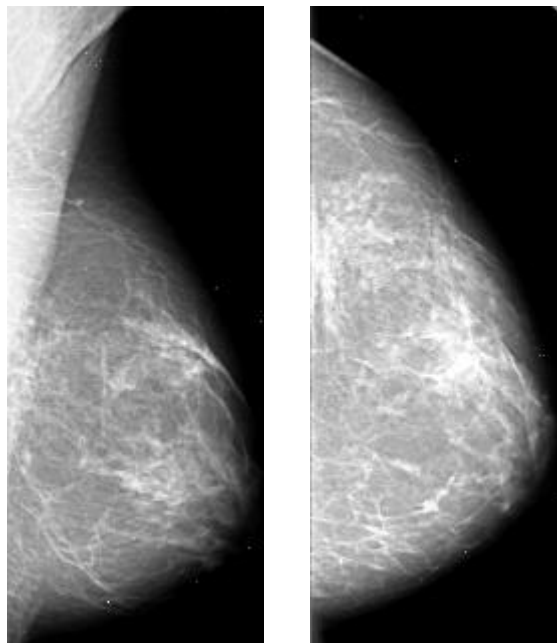


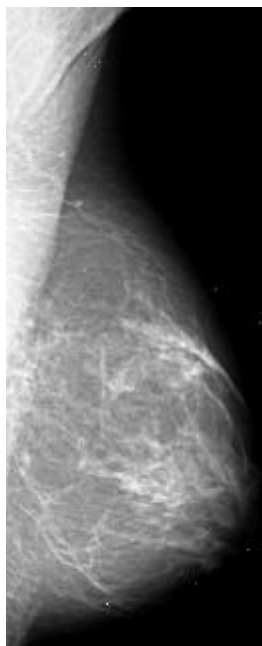
Figura 6.1 – Vistas MLO e CC [27].

As imagens mamográficas originais utilizadas neste trabalho foram selecionadas de um banco digital de imagens, *Digital Database for Screening Mammography*, da *University of South Florida* [27], disponibilizado para pesquisa. Foram escolhidos quatro casos diferentes, onde cada caso possui imagens das duas mamas (direita e esquerda) nas duas incidências básicas (MLO e CC).

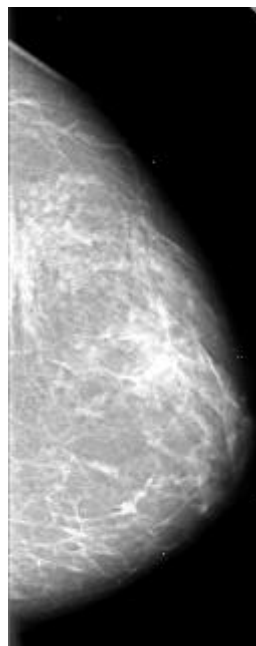
Dessa maneira, cada caso possui quatro imagens originais.

- Mama direita vista MLO;
- Mama direita vista CC;
- Mama esquerda vista MLO;
- Mama esquerda vista CC.

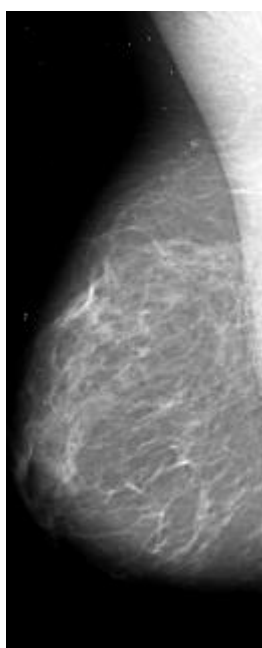
Considerando os quatro casos escolhidos, o total de imagens originais que é utilizado nos teste de compressão é de dezesseis. As Figura 6.2 a Figura 6.5 a seguir, mostram as imagens originais das duas incidências básicas (MLO e CC) de cada uma das duas mamas (direita e esquerda), de cada um dos quatro casos escolhidos.



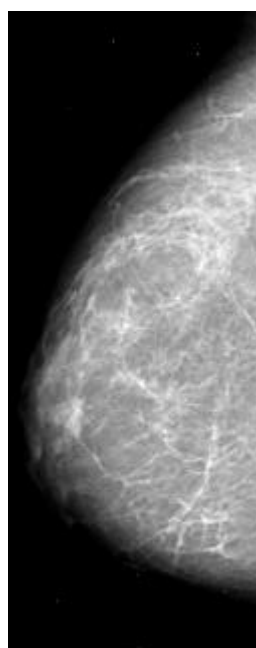
(a)



(b)

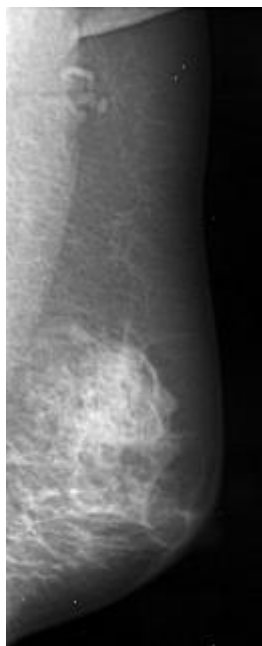


(c)

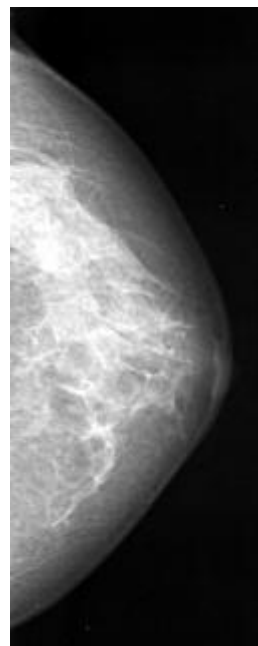


(d)

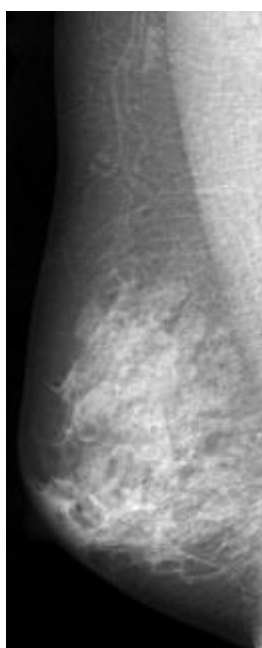
Figura 6.2 – Caso 1 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].



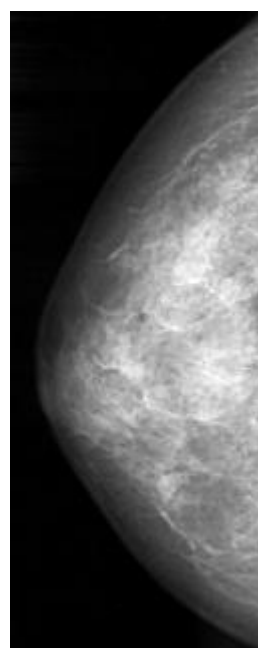
(a)



(b)

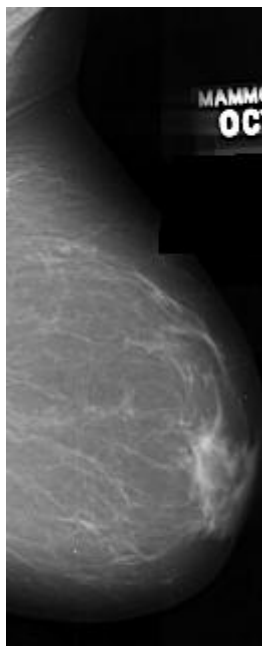


(c)

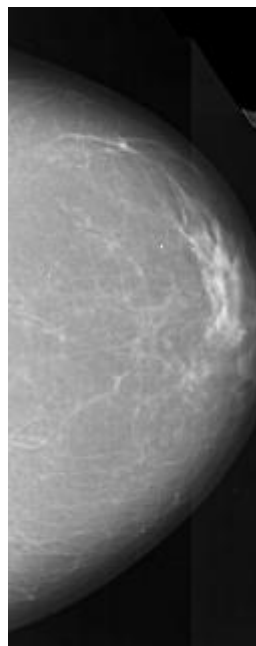


(d)

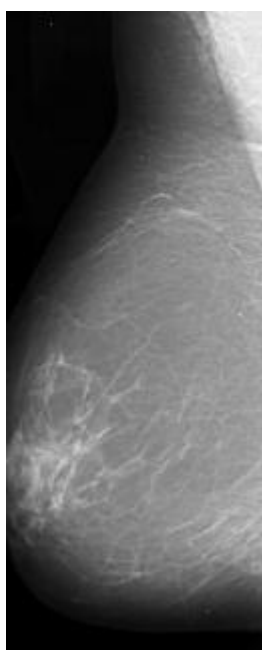
Figura 6.3 – Caso 2 (a) Mama Direita – vista MLO; (b) Mama Direita – vista; CC (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].



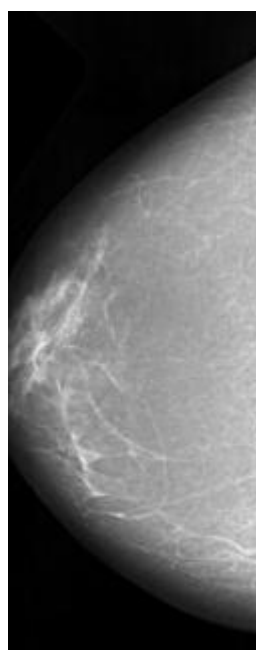
(a)



(b)

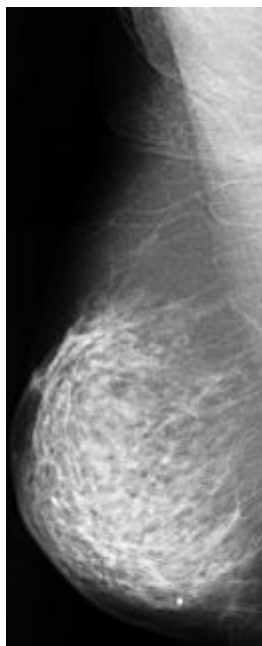


(c)

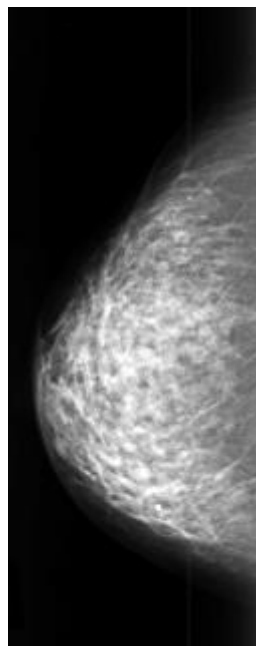


(d)

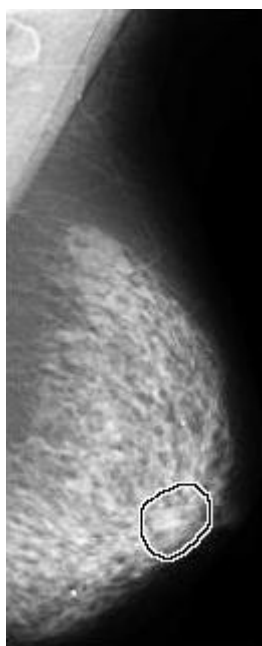
Figura 6.4 – Caso 3 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].



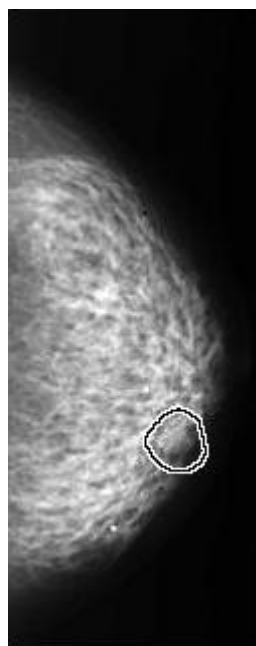
(a)



(b)



(c)



(d)

Figura 6.5 – Caso 4 (a) Mama Direita – vista MLO; (b) Mama Direita – vista CC; (c) Mama Esquerda – vista MLO; (d) Mama Esquerda – vista CC [27].

Foram realizados testes de compressão de imagens utilizando os dois métodos de compressão citados, compressão fractal e compressão JPEG2000. Para o método de compressão fractal, foi desenvolvido um algoritmo de compressão fractal de imagens em Matlab, baseado no método da partição quadtree desenvolvido por *Fischer* [3].

Os parâmetros de entrada utilizados foram: sobreposição de 25% na etapa de construção do *Domain Pool*, nível máximo de partição = 6, quantidade de bits para representar o contraste = 5, quantidade de bits para representar o brilho = 7 e máximo valor para contraste = 1,2.

Cada uma das imagens originais foi comprimida com cinco taxas de compressão diferentes. Estas taxas foram obtidas alterando-se apenas um dos possíveis parâmetros de entrada, o limiar do erro RMS. Quanto menor é o limiar de erro RMS, menor é a taxa de compressão e melhor é a qualidade da imagem comprimida. Para obtenção das cinco diferentes taxas de compressão foram utilizados cinco valores de limiar de erro RMS: 10, 20, 30, 50 e 70.

As mesmas imagens originais utilizadas na compressão fractal foram utilizadas na compressão JPEG2000. Nesta compressão, foi utilizado o software *Jasper*, desenvolvido por Michael Adams, afiliado ao *Digital Signal Processing Group* (DSPG) no Departamento de Engenharia Elétrica e de Computação da *University of Victoria*, Canadá [28]. As imagens originais foram comprimidas também com cinco taxas de compressão diferentes.

Para cada um dos dois métodos de compressão, considerando que existem dezesseis imagens originais, e que cada uma das imagens originais foi comprimida com cinco taxas de compressão diferentes, existe um total de oitenta imagens comprimidas.

Considerando as quatro imagens originais do Caso 1, as Figura 6.6 a Figura 6.9, mostram a imagem original e as cinco imagens comprimidas pelo método de compressão fractal e as Figura 6.10 a Figura 6.13, ilustram a imagem original e as cinco imagens comprimidas pelo método de compressão JPEG2000.

Considerando as quatro imagens originais do Caso 2, as Figura 6.14 a Figura 6.17, mostram a imagem original e as cinco imagens comprimidas pelo método de compressão

fractal e as Figura 6.18 a Figura 6.21, ilustram a imagem original e as cinco imagens comprimidas pelo método de compressão JPEG2000.

Considerando as quatro imagens originais do Caso 3, as Figura 6.22 a Figura 6.25, mostram a imagem original e as cinco imagens comprimidas pelo método de compressão fractal e as Figura 6.26 a Figura 6.29, ilustram a imagem original e as cinco imagens comprimidas pelo método de compressão JPEG2000.

Considerando as quatro imagens originais do Caso 4, as Figura 6.30 a Figura 6.33, mostram a imagem original e as cinco imagens comprimidas pelo método de compressão fractal e as Figura 6.34 a Figura 6.37, ilustram a imagem original e as cinco imagens comprimidas pelo método de compressão JPEG2000.

Em razão das imagens impressas perderem a qualidade e a nitidez, para possibilitar uma melhor análise, segue como Apêndice ao final desta dissertação um CD contendo todas as imagens das Figura 6.6 a Figura 6.37 para melhor análise.

6.2.1 Caso 1

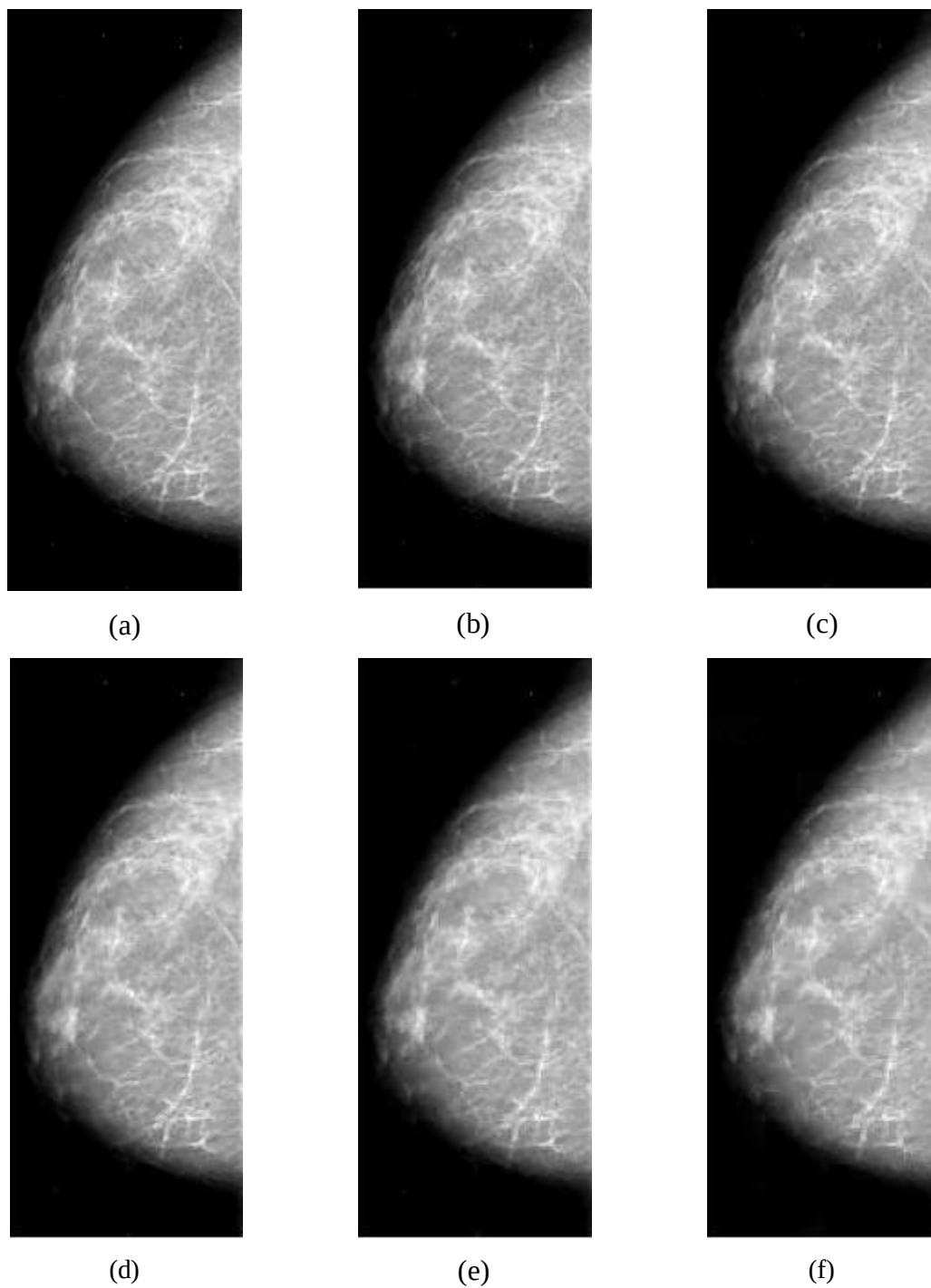
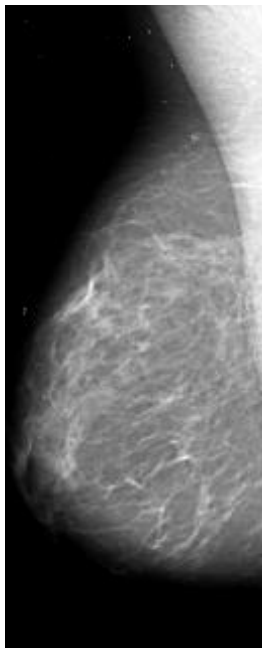
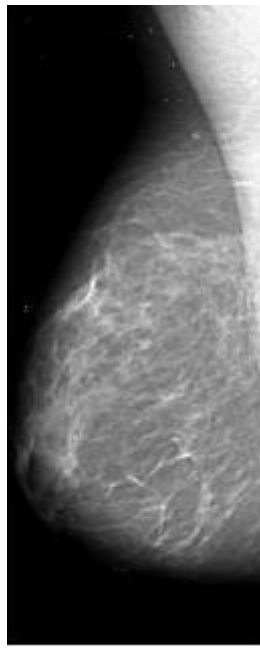


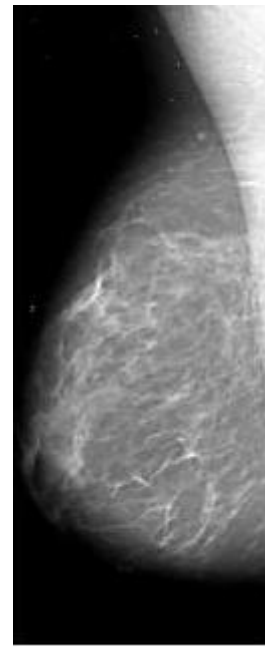
Figura 6.6 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 2,09 bpp; (c) 1,39 bpp; (d) 0,75 bpp; (e) 0,51 bpp; (f) 0,39 bpp.



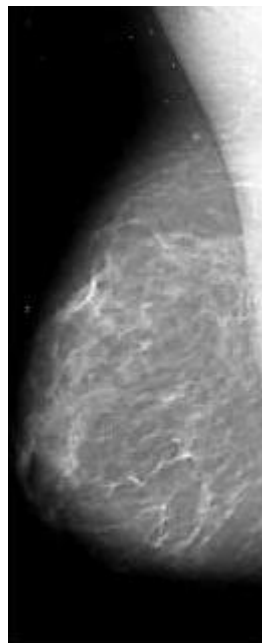
(a)



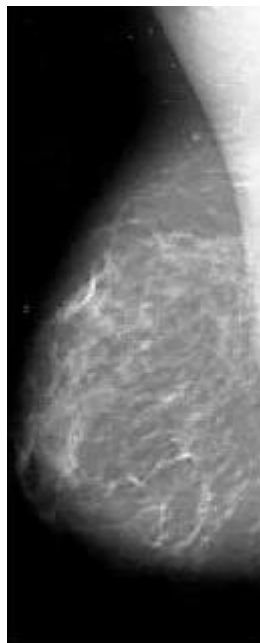
(b)



(c)



(d)

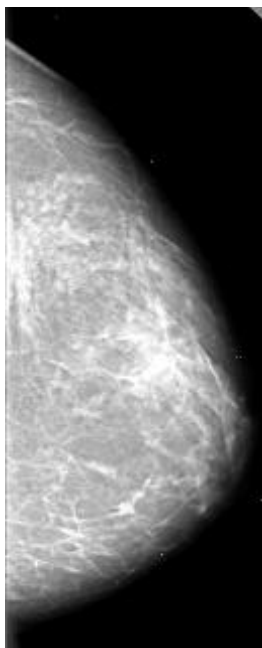


(e)

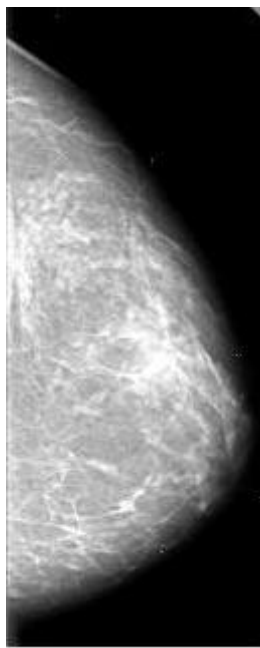


(f)

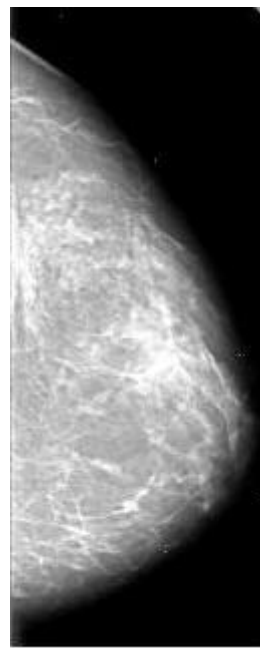
Figura 6.7 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,09 bpp; (c) 1,19 bpp; (d) 0,7 bpp; (e) 0,48 bpp; (f) 0,31 bpp.



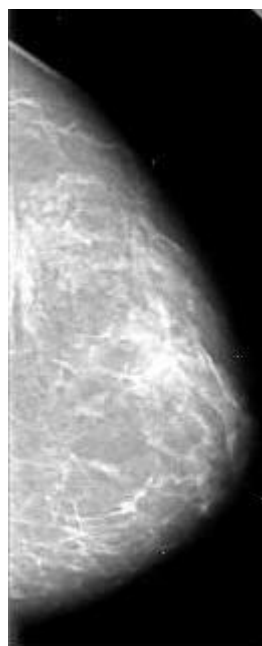
(a)



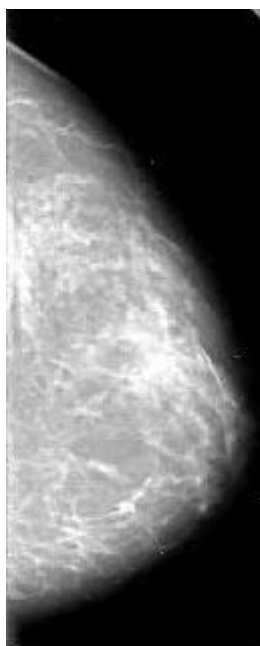
(b)



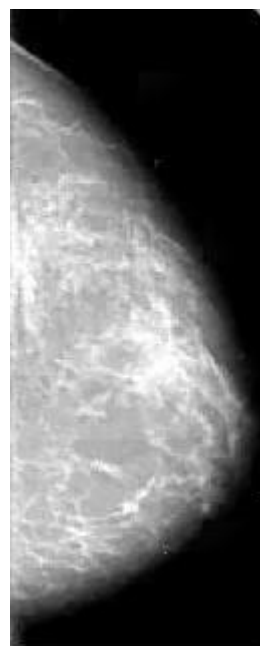
(c)



(d)

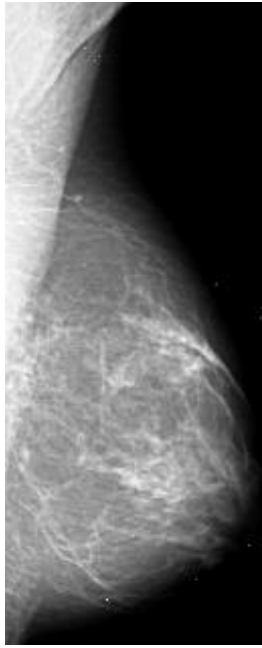


(e)

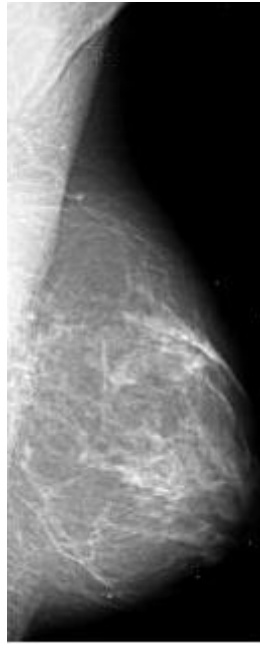


(f)

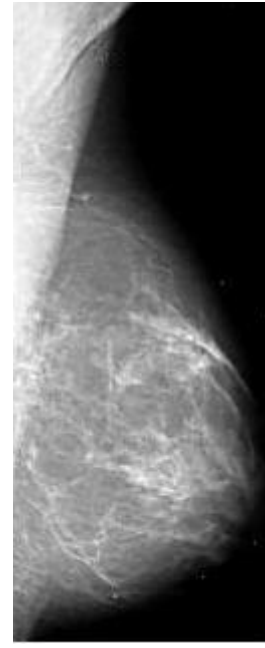
Figura 6.8 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 2,17 bpp; (c) 1,51 bpp; (d) 0,87 bpp; (e) 0,56 bpp; (f) 0,43 bpp.



(a)



(b)



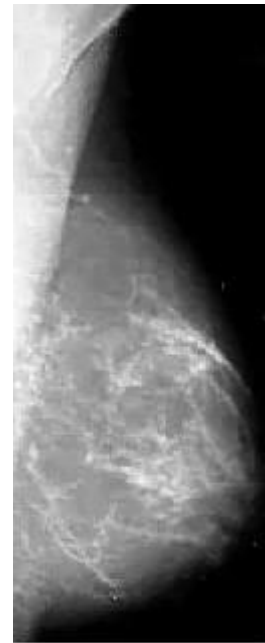
(c)



(d)



(e)



(f)

Figura 6.9 - Compressão Fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 2,19 bpp; (c) 1,34 bpp; (d) 0,78 bpp; (e) 0,51 bpp; (f) 0,36 bpp.

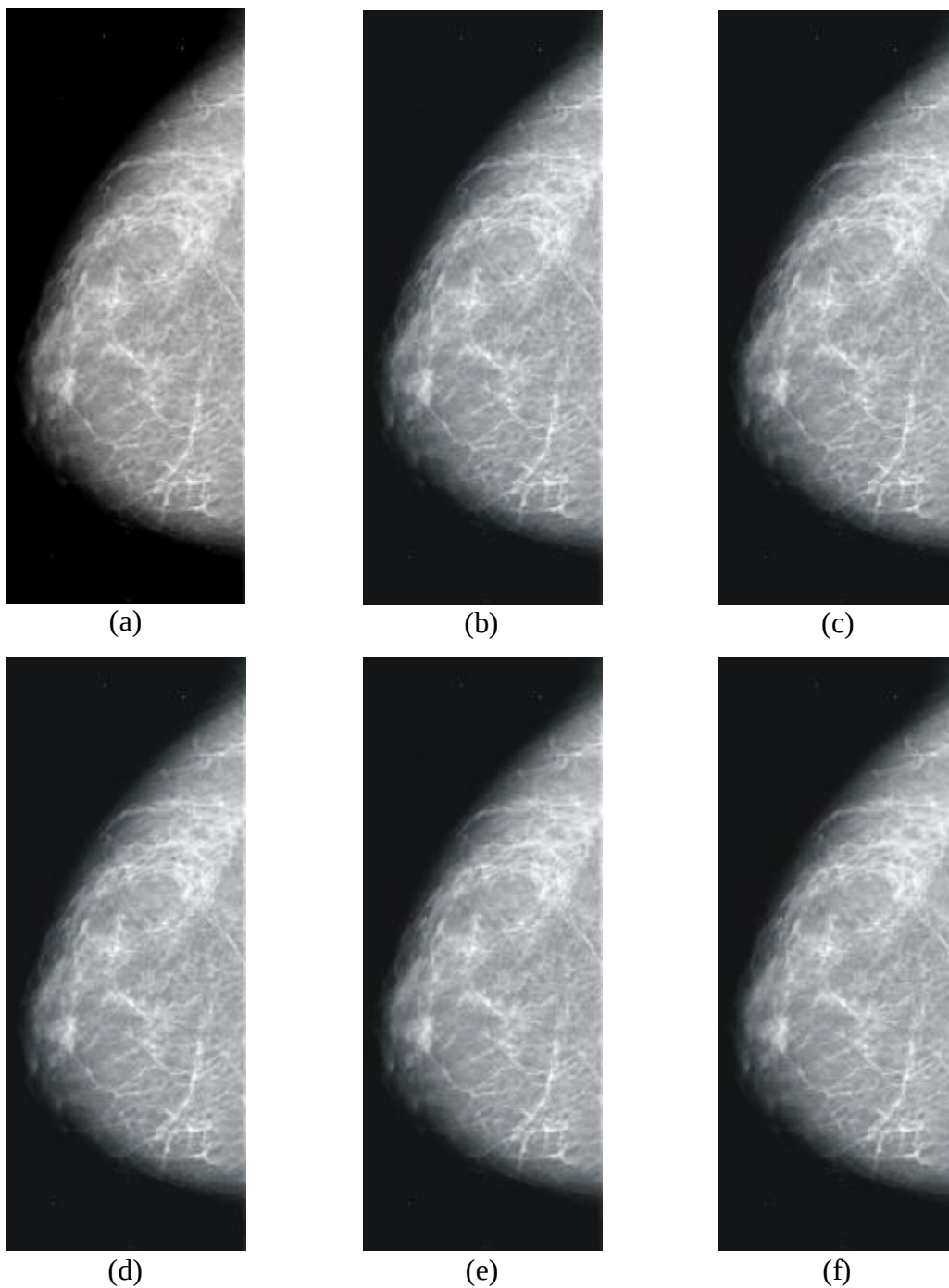


Figura 6.10 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,41 bpp; (c) 0,41 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

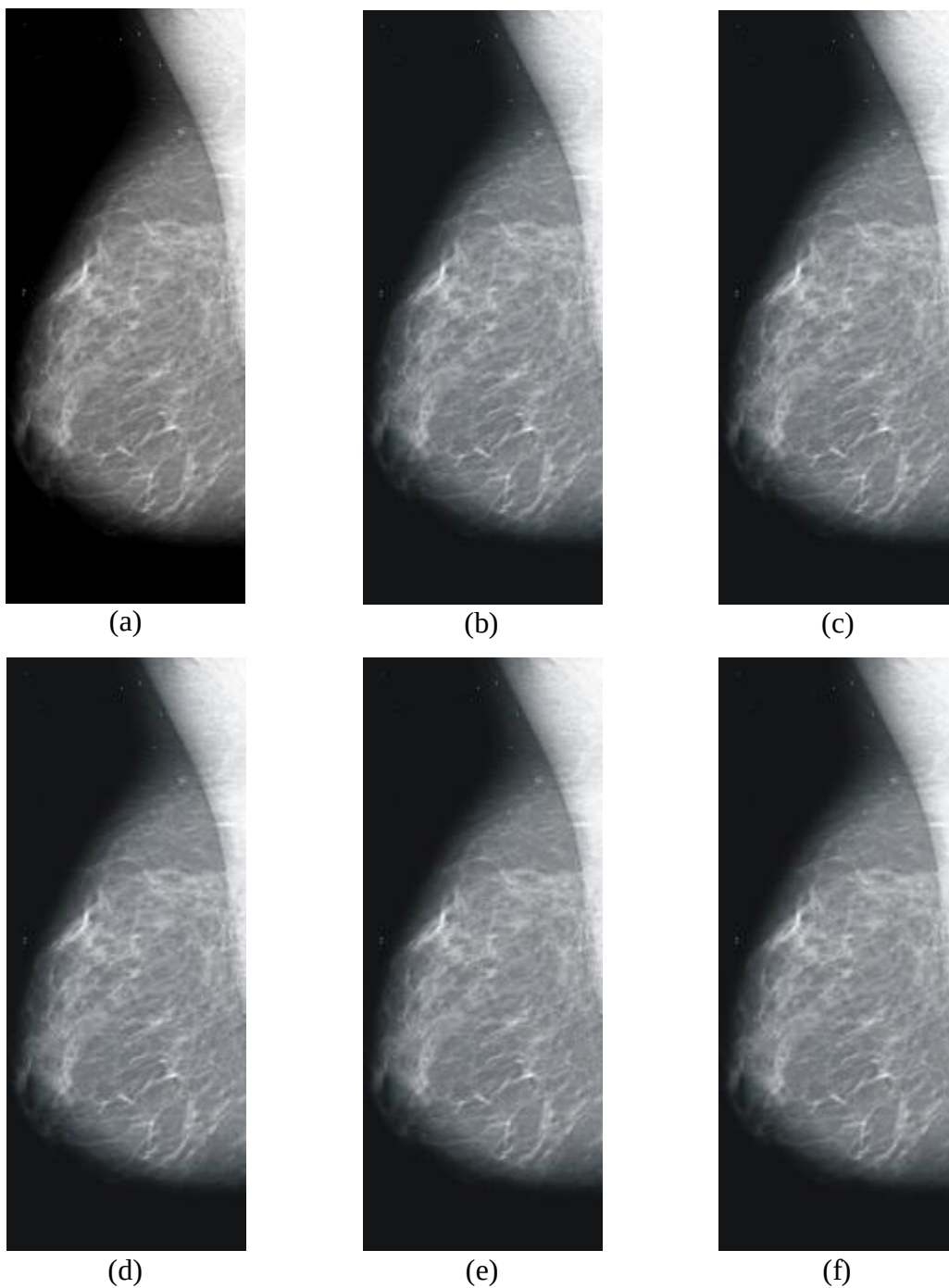


Figura 6.11 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,43 bpp; (c) 0,43 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

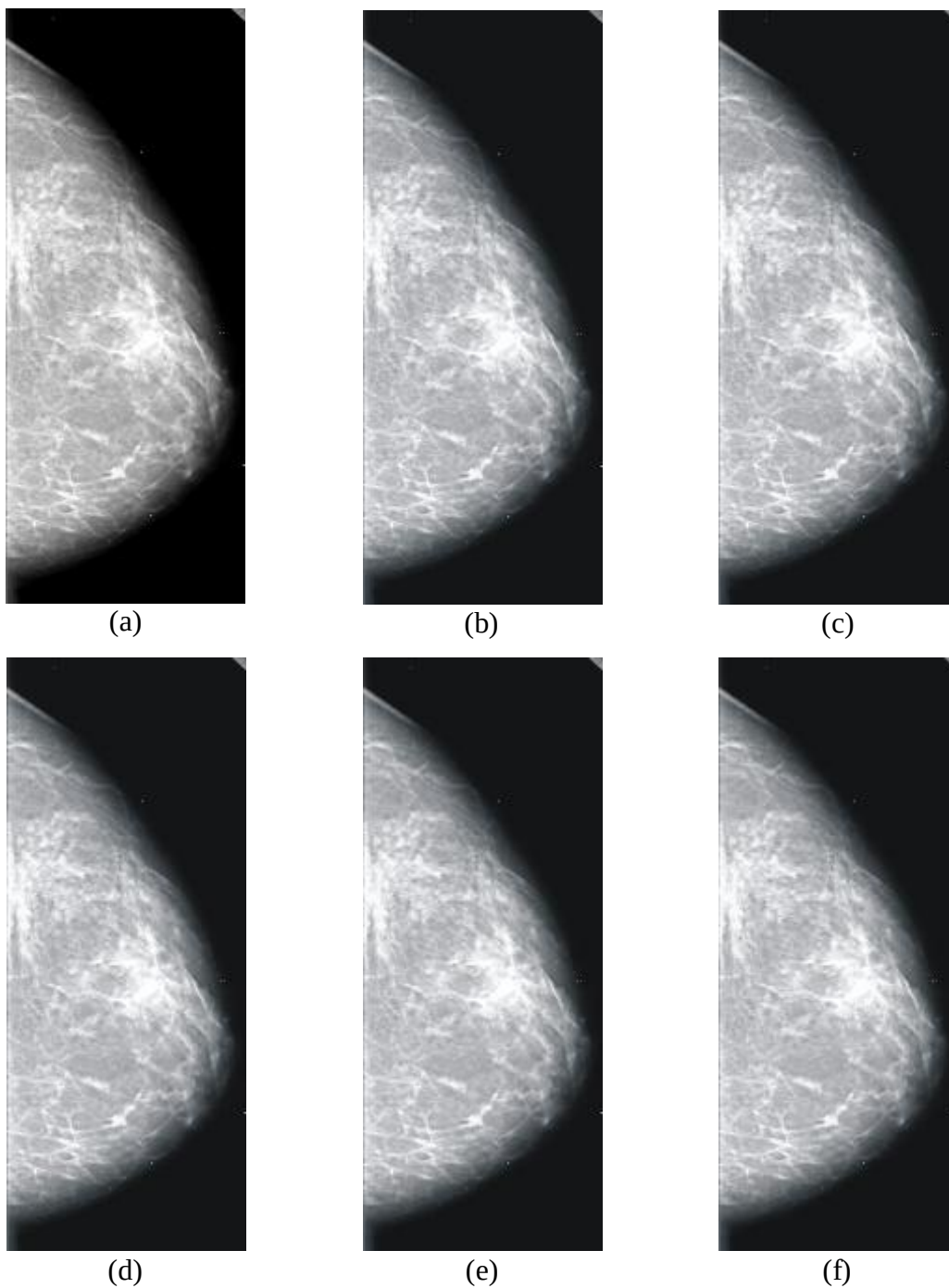
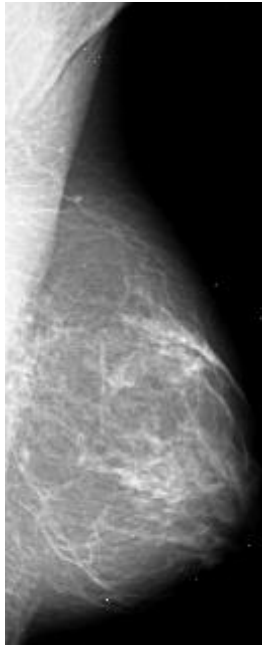
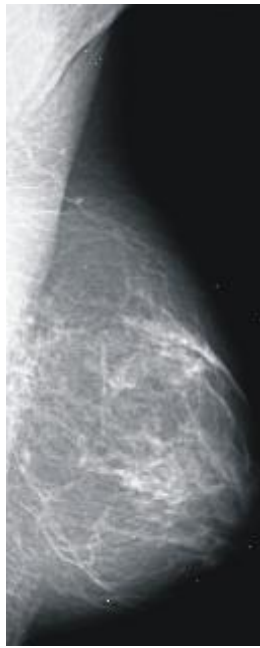


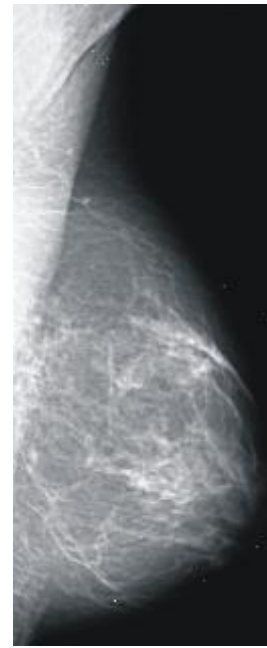
Figura 6.12 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,43 bpp; (c) 0,43 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.



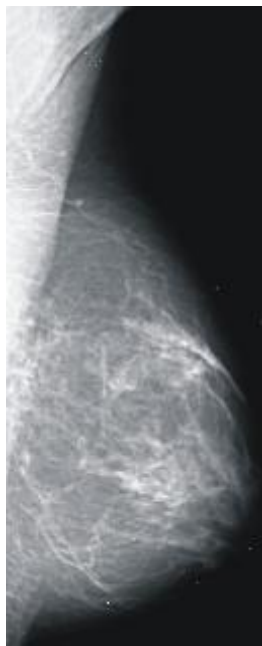
(a)



(b)



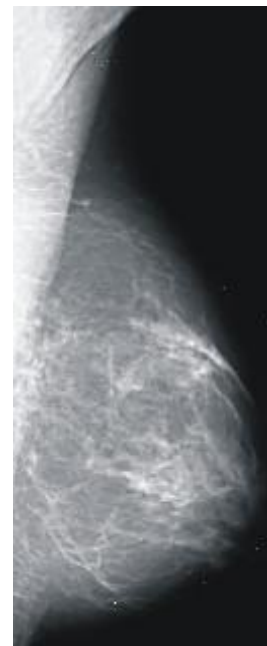
(c)



(d)



(e)



(f)

Figura 6.13 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

6.2.2 Caso 2

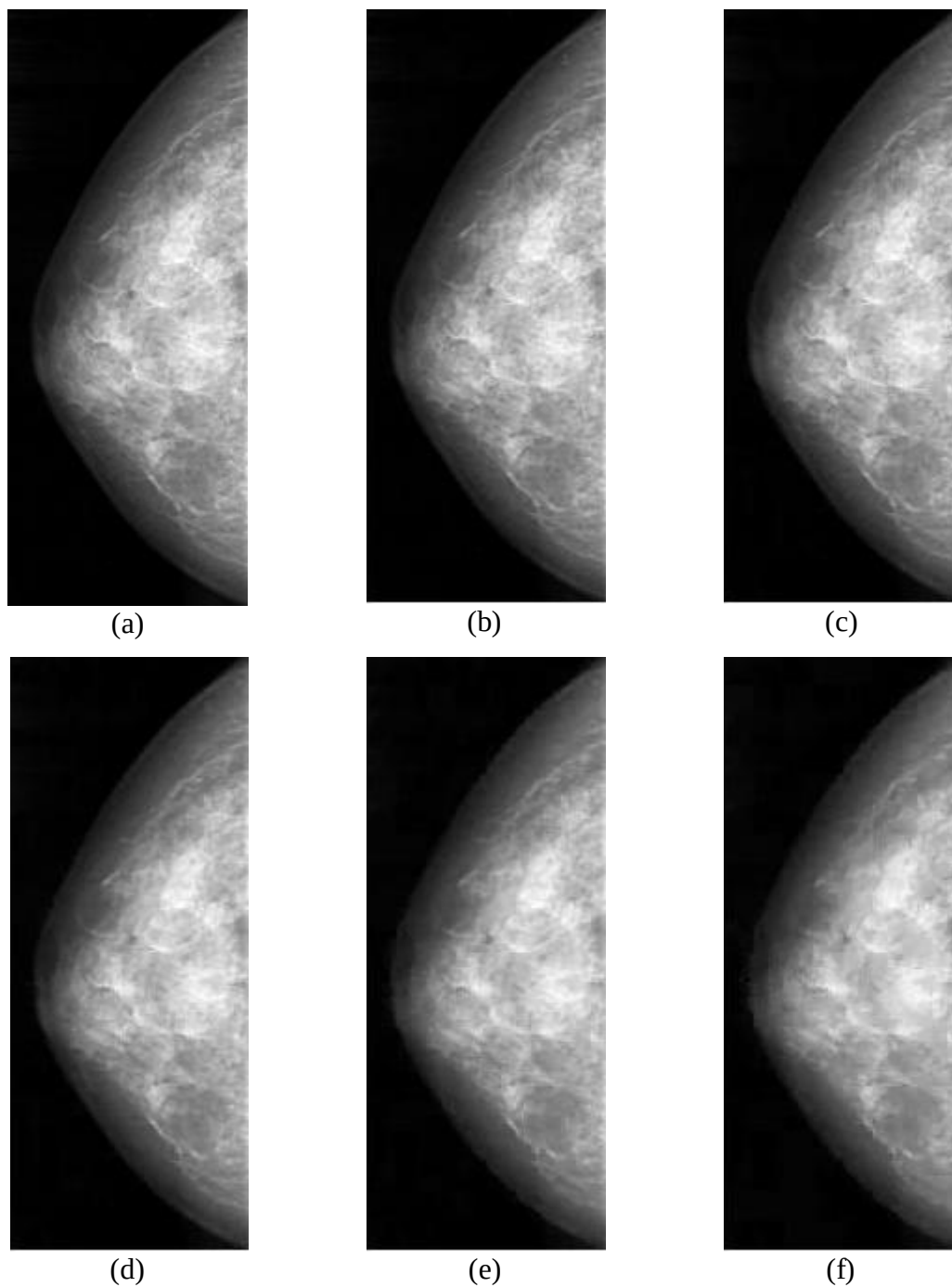


Figura 6.14 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,9 bpp; (c) 1 bpp; (d) 0,61 bpp; (e) 0,43 bpp; (f) 0,29 bpp.

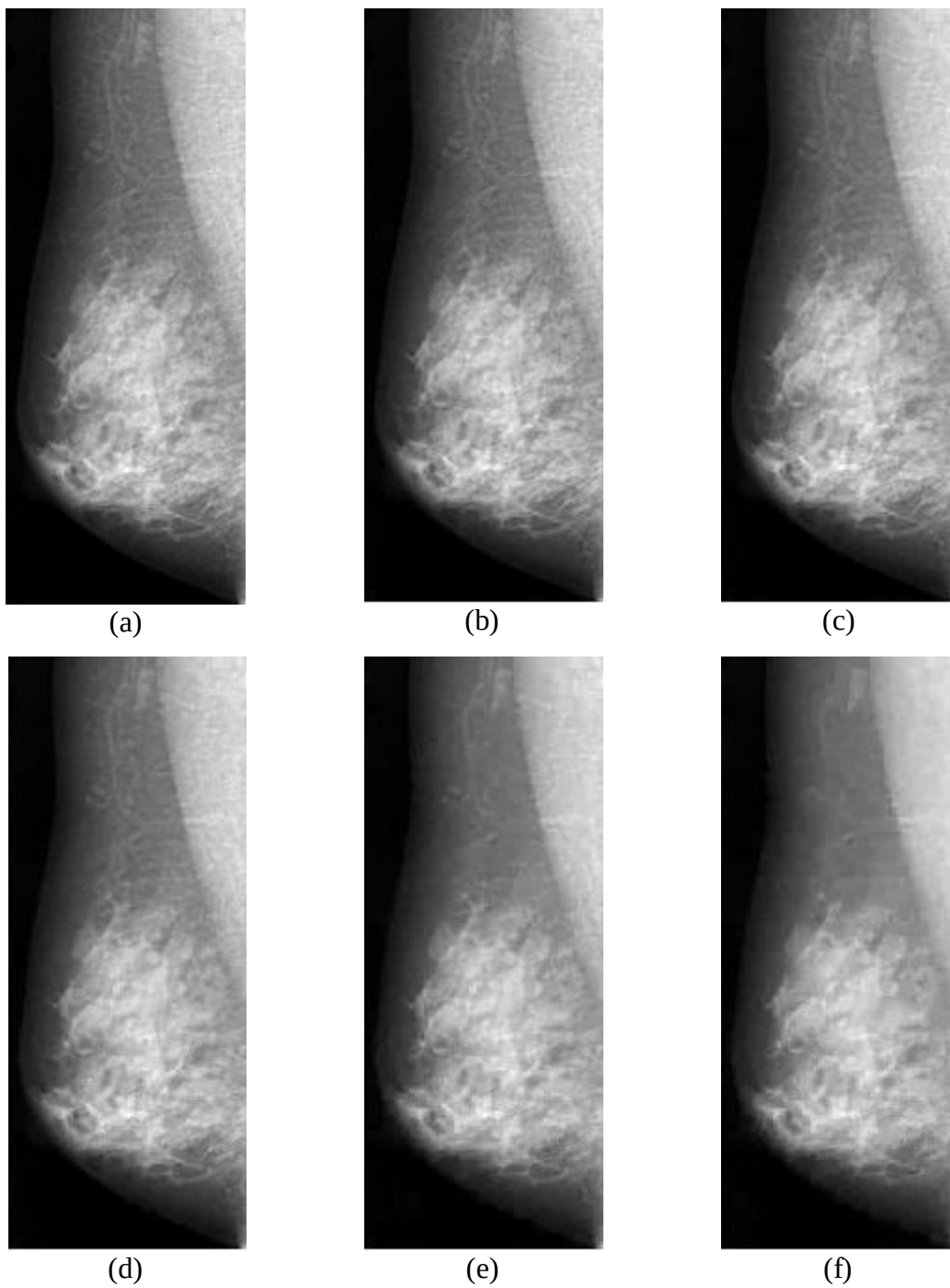
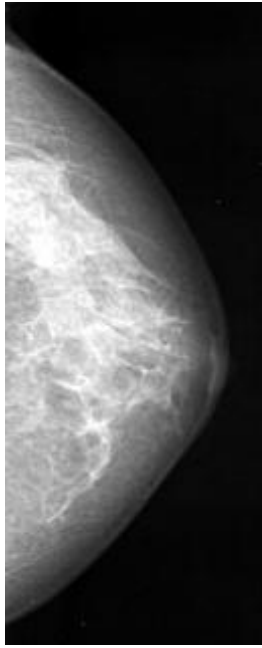
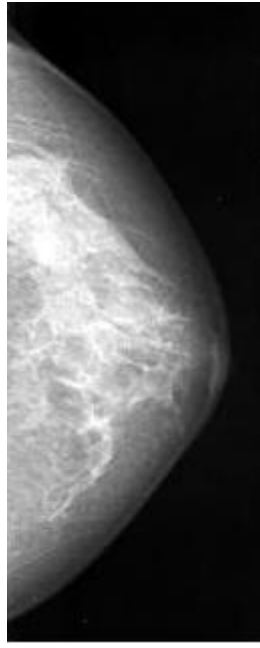


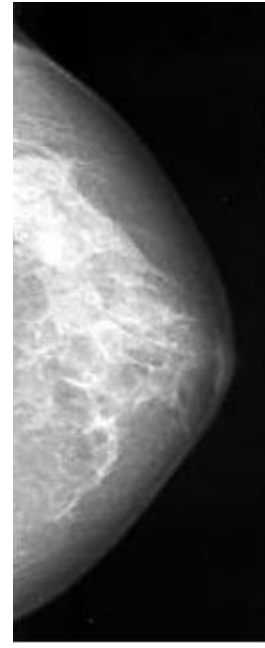
Figura 6.15 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,39 bpp; (c) 1,26 bpp; (d) 0,73 bpp; (e) 0,51 bpp; (f) 0,34 bpp.



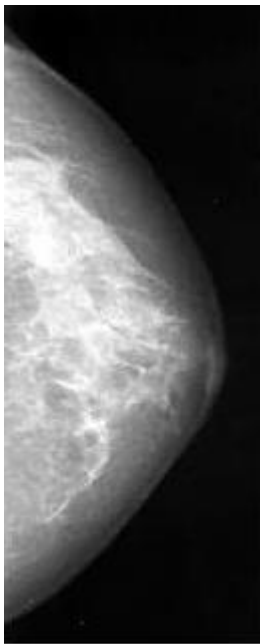
(a)



(b)



(c)



(d)

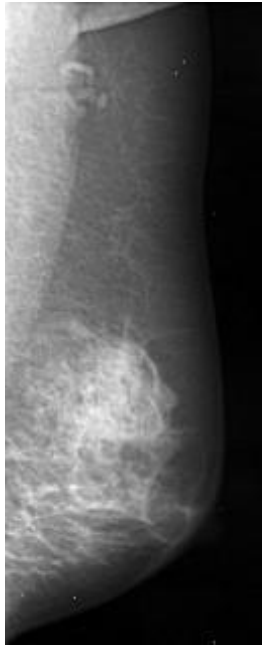


(e)



(f)

Figura 6.16 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,78 bpp; (c) 0,97 bpp; (d) 0,56 bpp; (e) 0,39 bpp; (f) 0,29 bpp.



(a)



(b)



(c)



(d)



(e)



(f)

Figura 6.17 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 1,97 bpp; (c) 1 bpp; (d) 0,63 bpp; (e) 0,41 bpp; (f) 0,29 bpp.

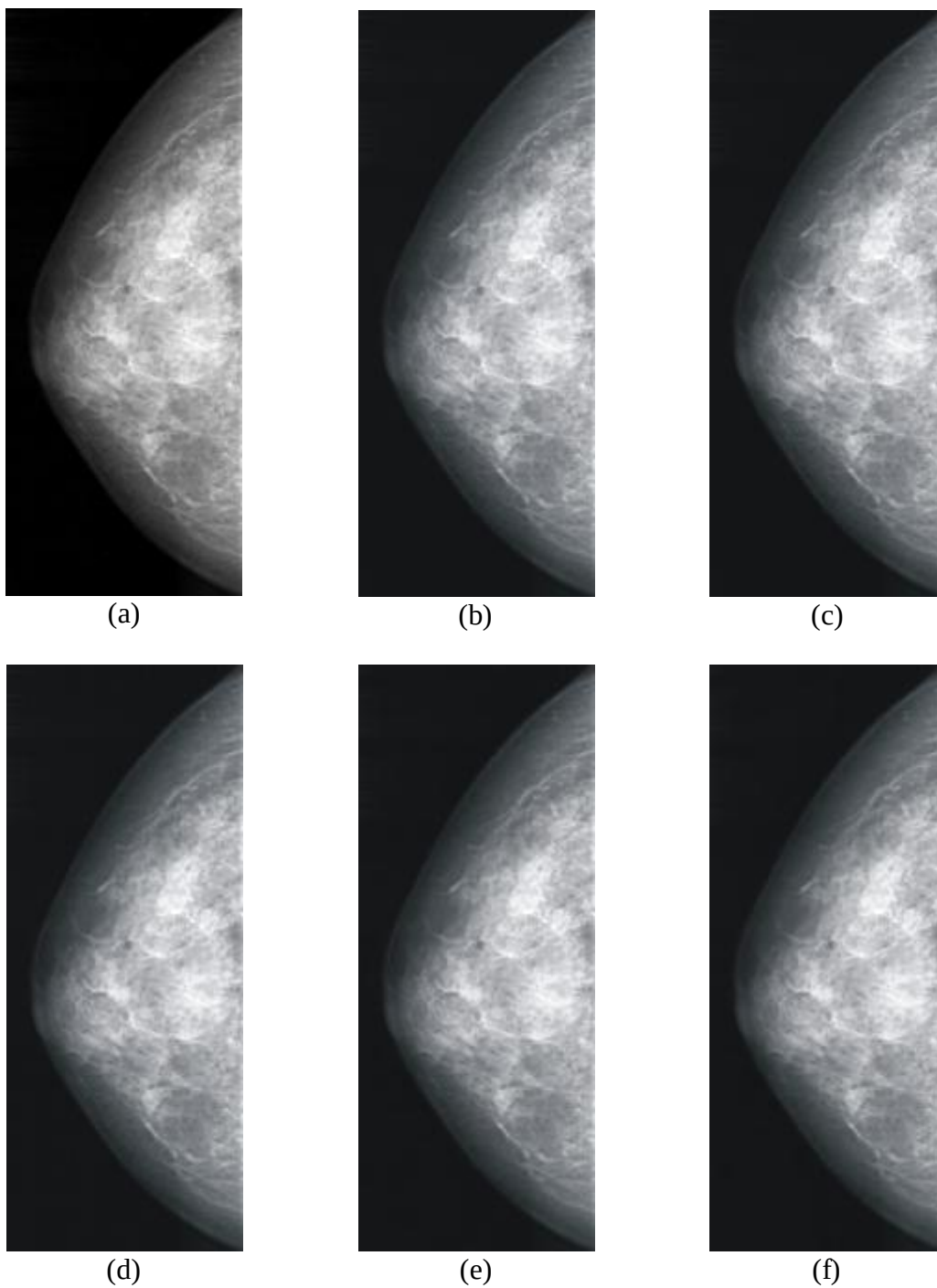


Figura 6.18 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

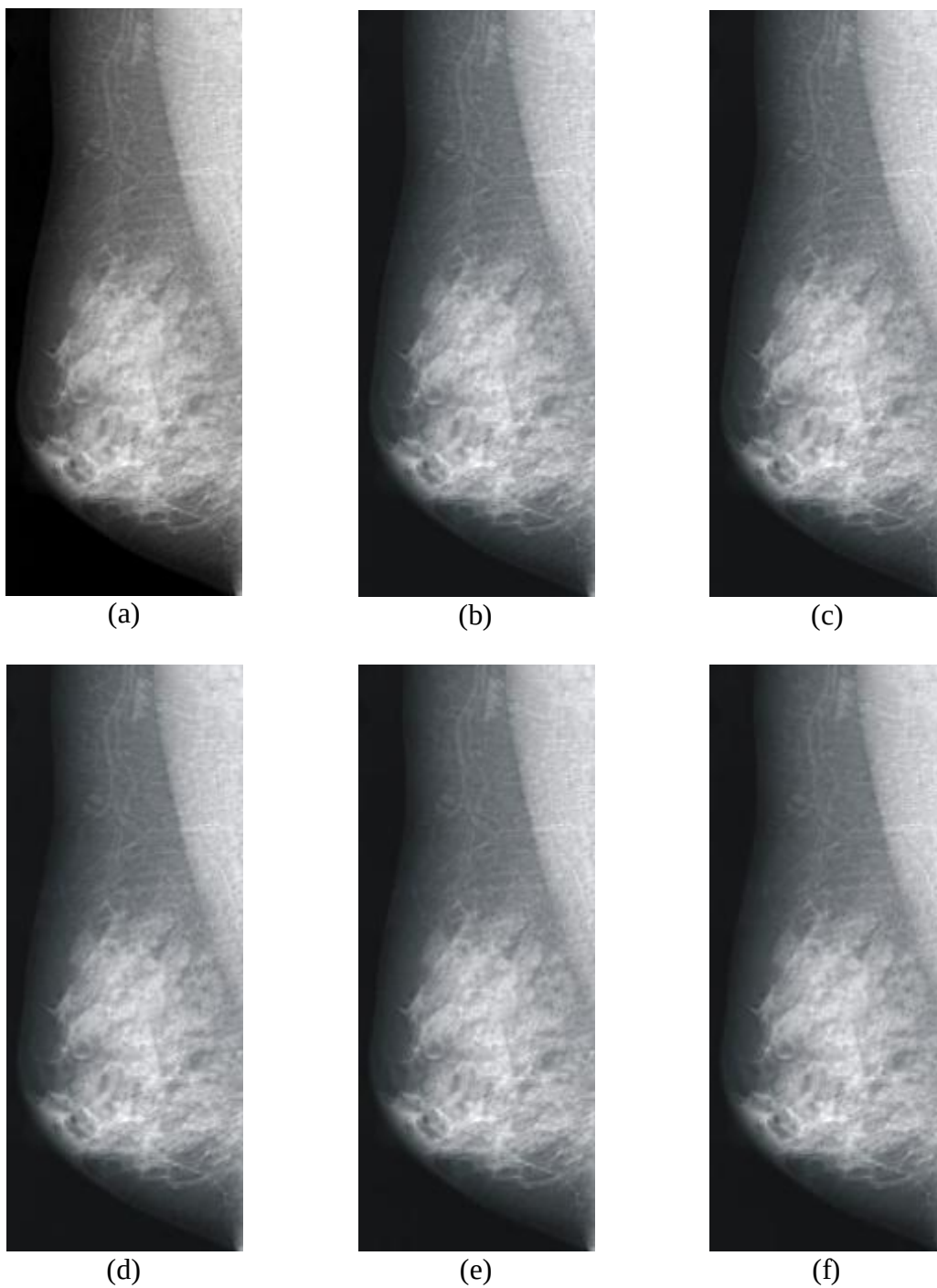


Figura 6.19 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,51 bpp; (c) 0,51 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

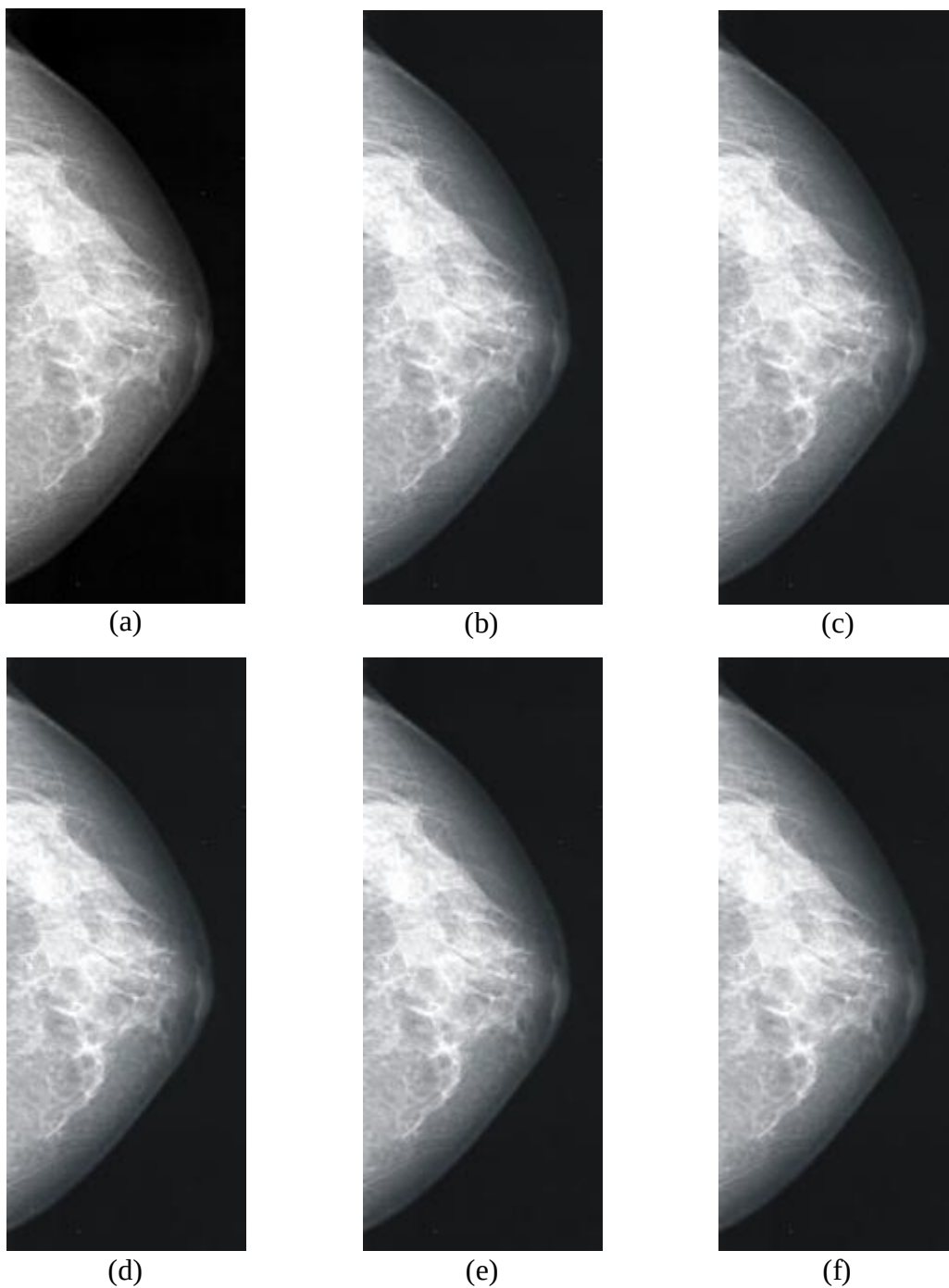
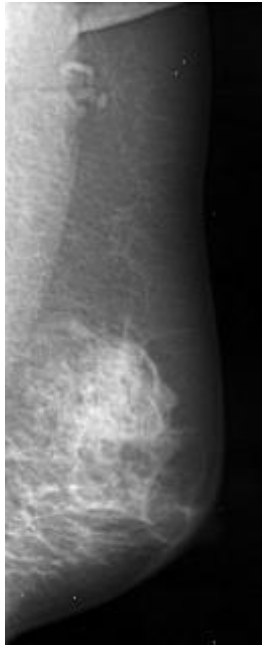
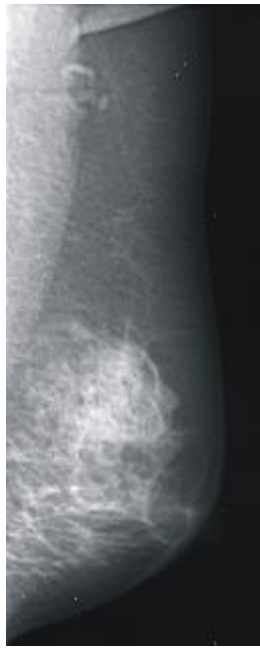


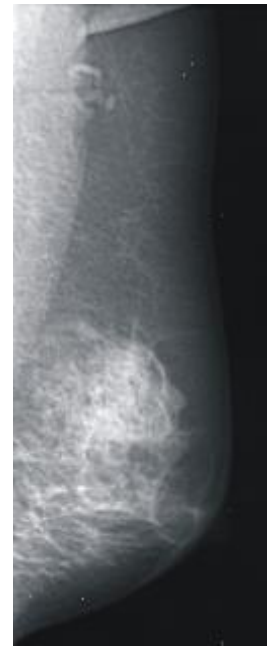
Figura 6.20 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.



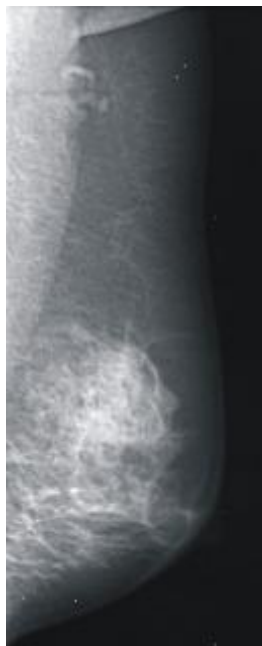
(a)



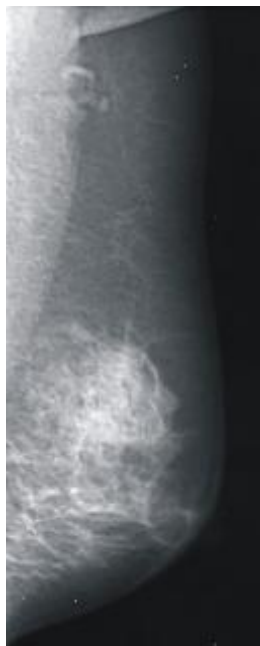
(b)



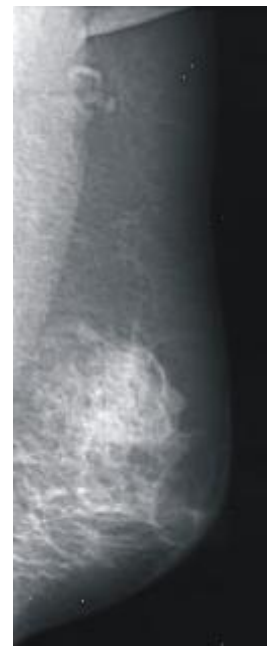
(c)



(d)



(e)



(f)

Figura 6.21 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

6.2.3 Caso 3

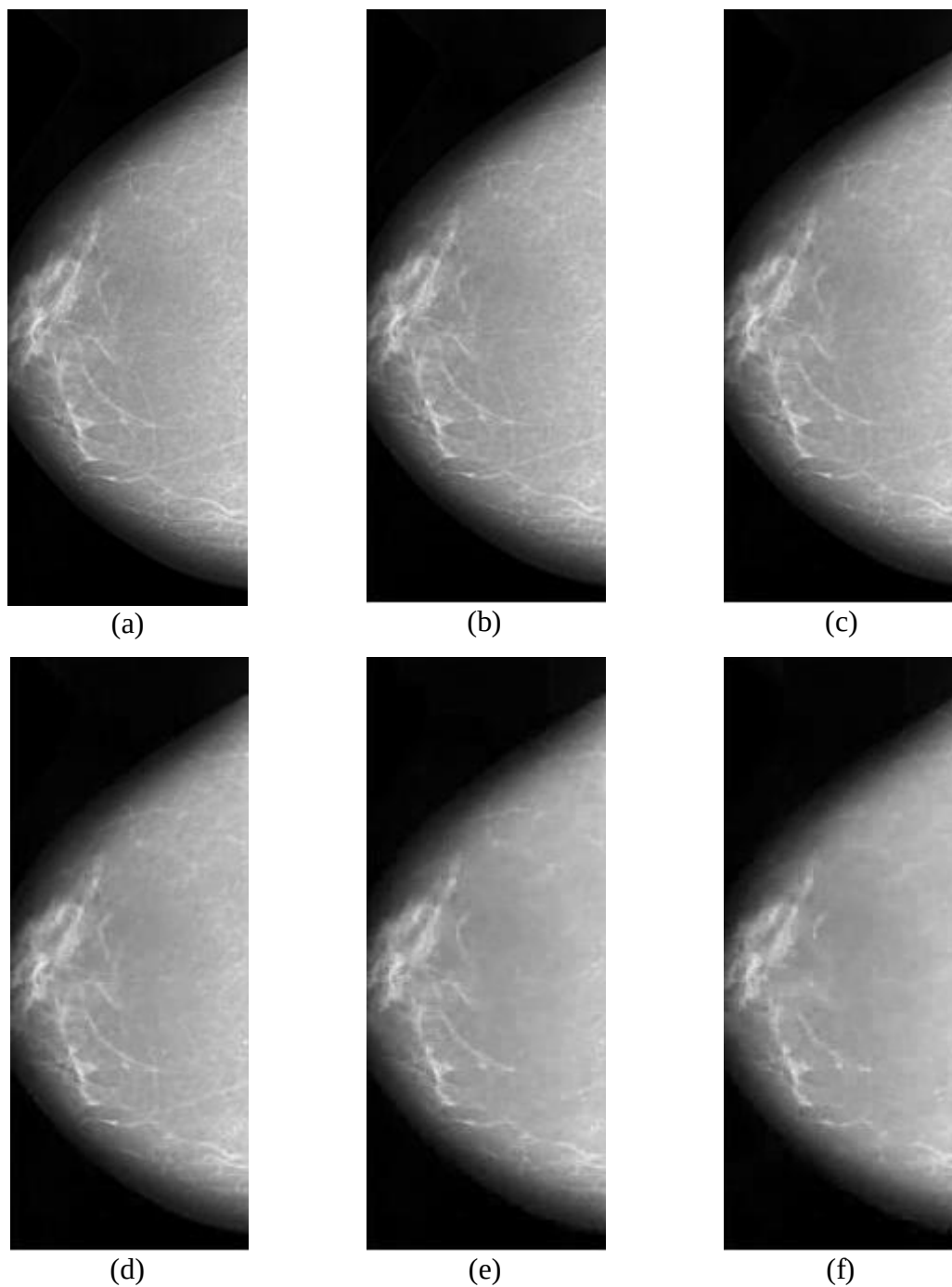


Figura 6.22 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,83 bpp; (c) 0,87 bpp; (d) 0,58 bpp; (e) 0,31 bpp; (f) 0,21 bpp.

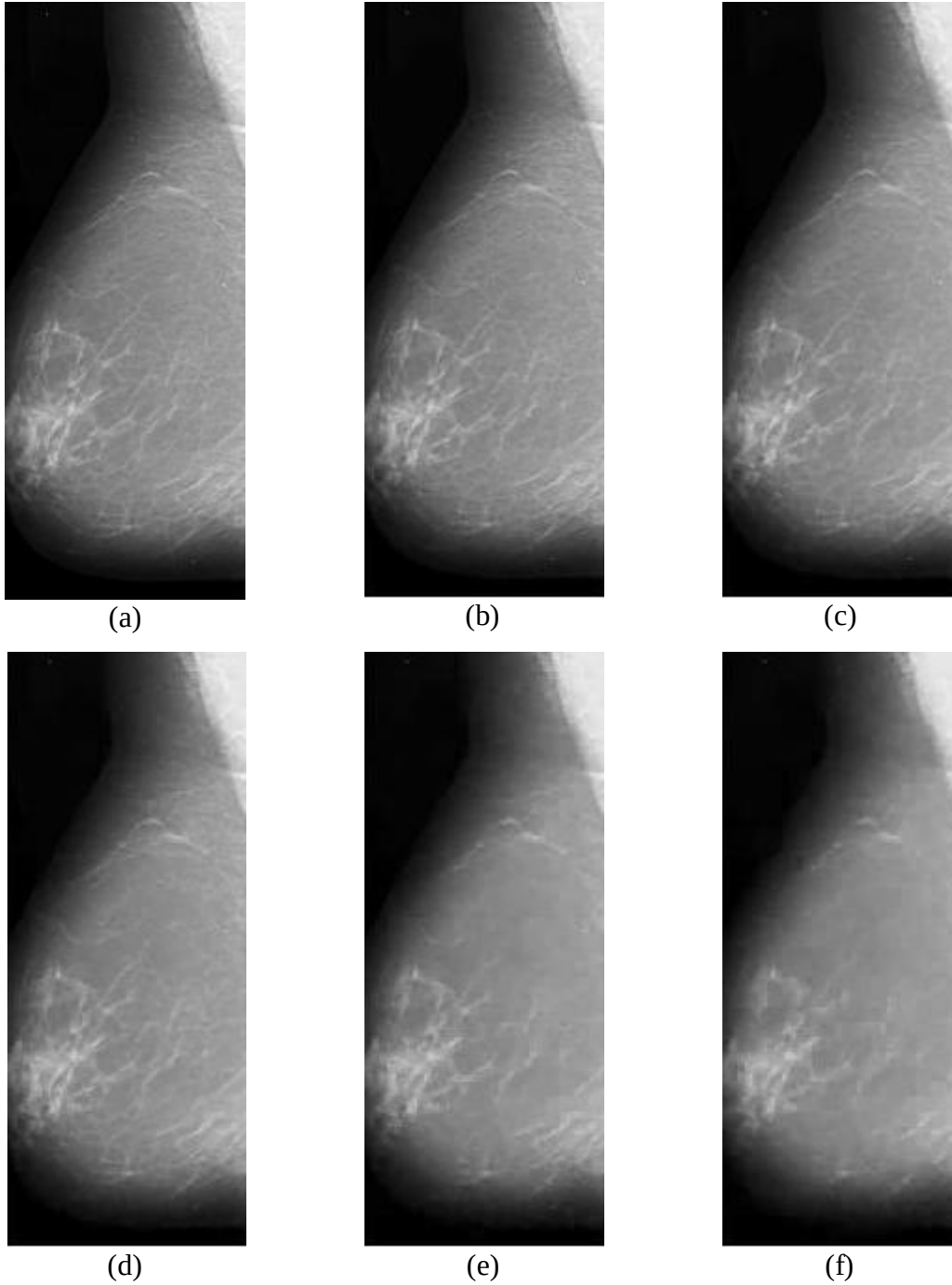
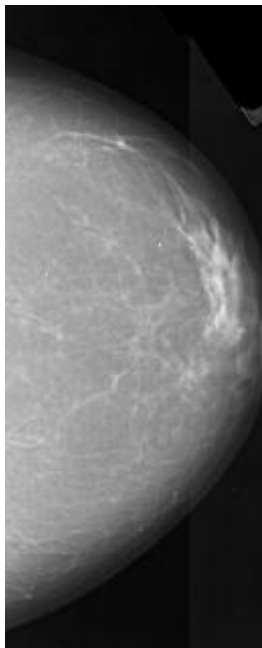
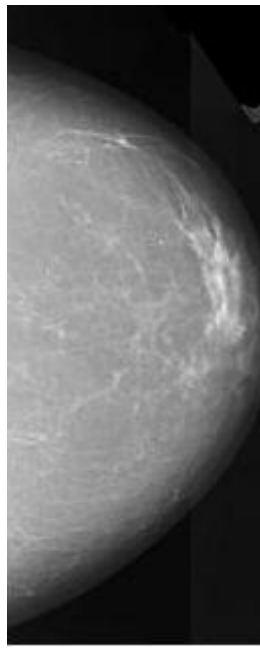


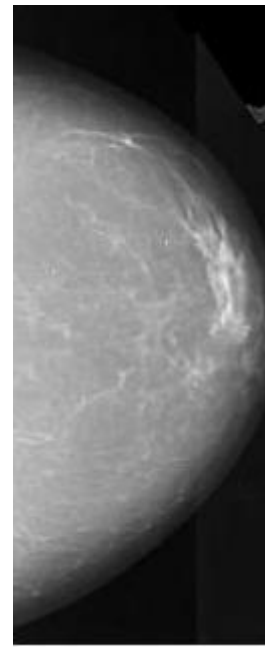
Figura 6.23 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 1,92 bpp; (c) 0,9 bpp; (d) 0,58 bpp; (e) 0,31 bpp; (f) 0,21 bpp.



(a)



(b)



(c)



(d)

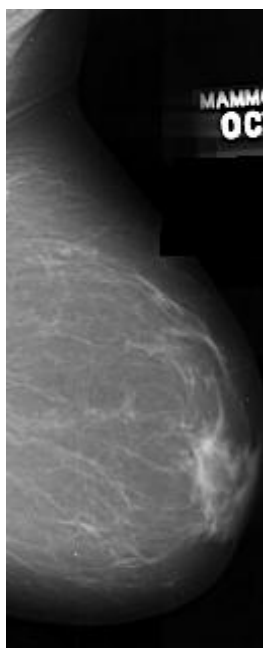


(e)

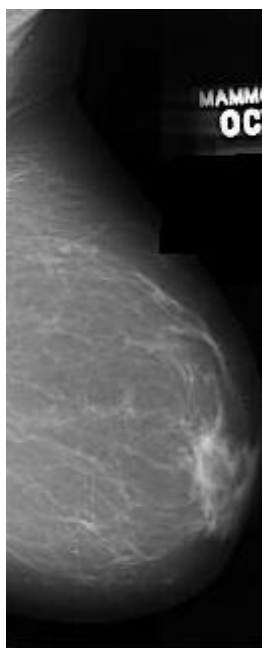


(f)

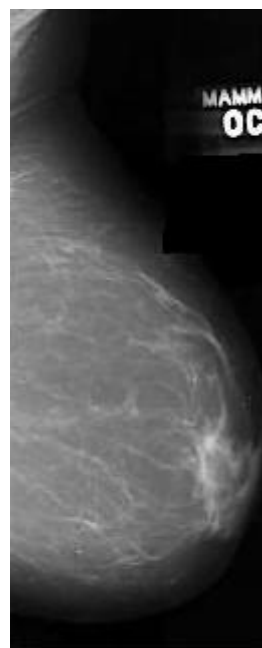
Figura 6.24 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,78 bpp; (c) 0,83 bpp; (d) 0,58 bpp; (e) 0,29 bpp; (f) 0,21 bpp.



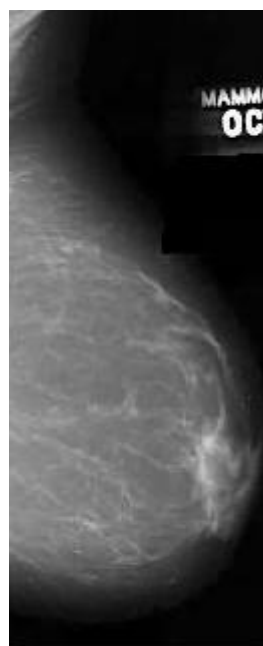
(a)



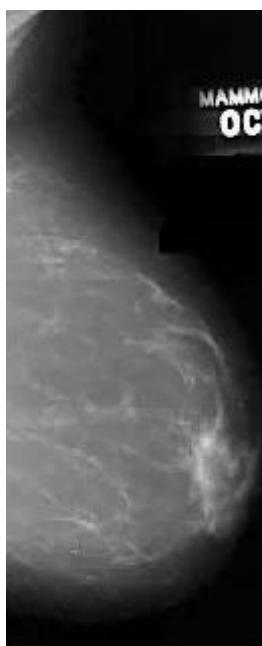
(b)



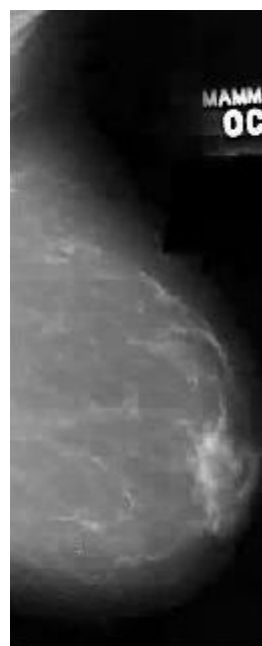
(c)



(d)



(e)



(f)

Figura 6.25 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 1,75 bpp; (c) 0,85 bpp; (d) 0,61 bpp; (e) 0,36 bpp; (f) 0,26 bpp.

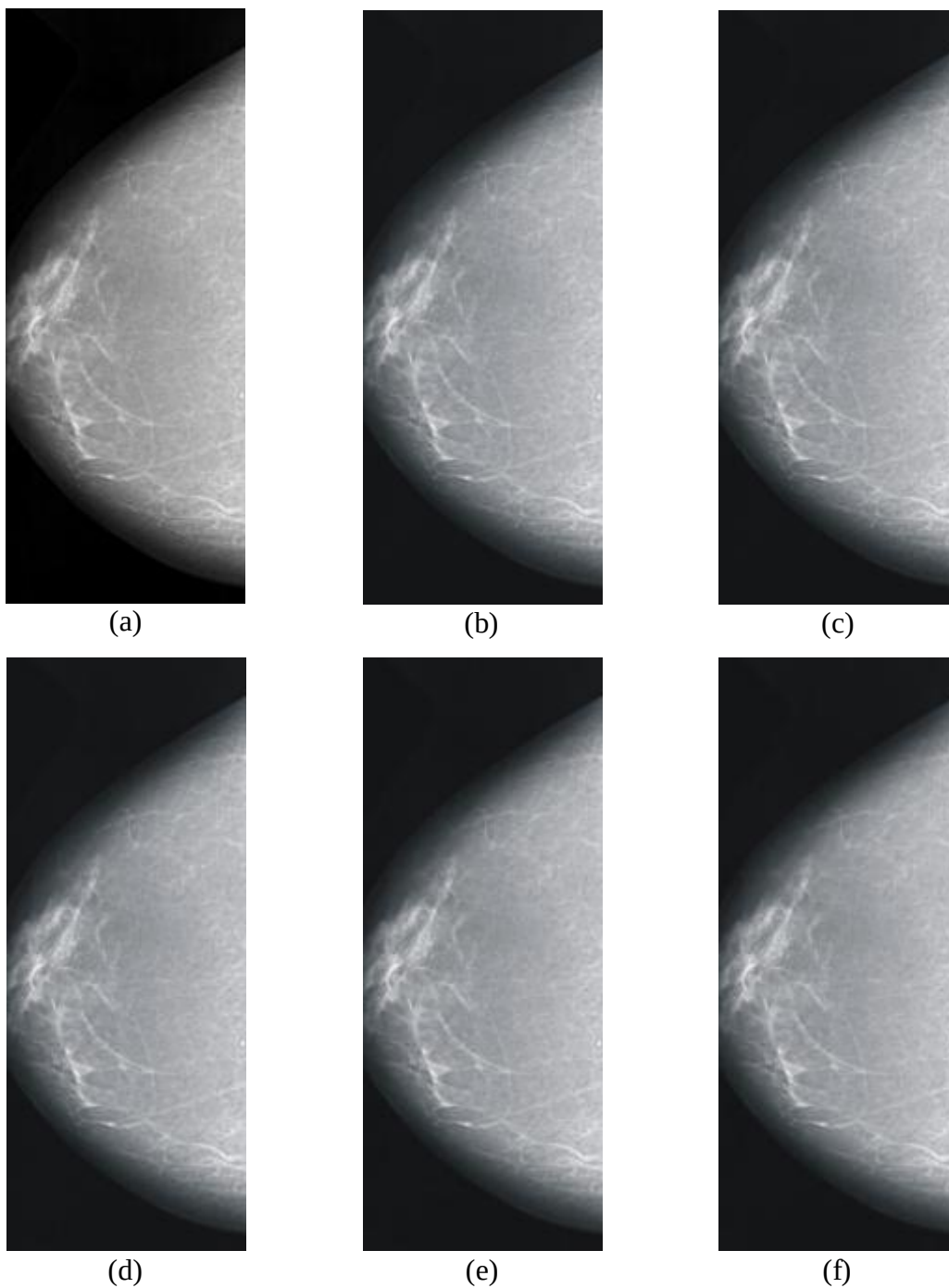


Figura 6.26 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

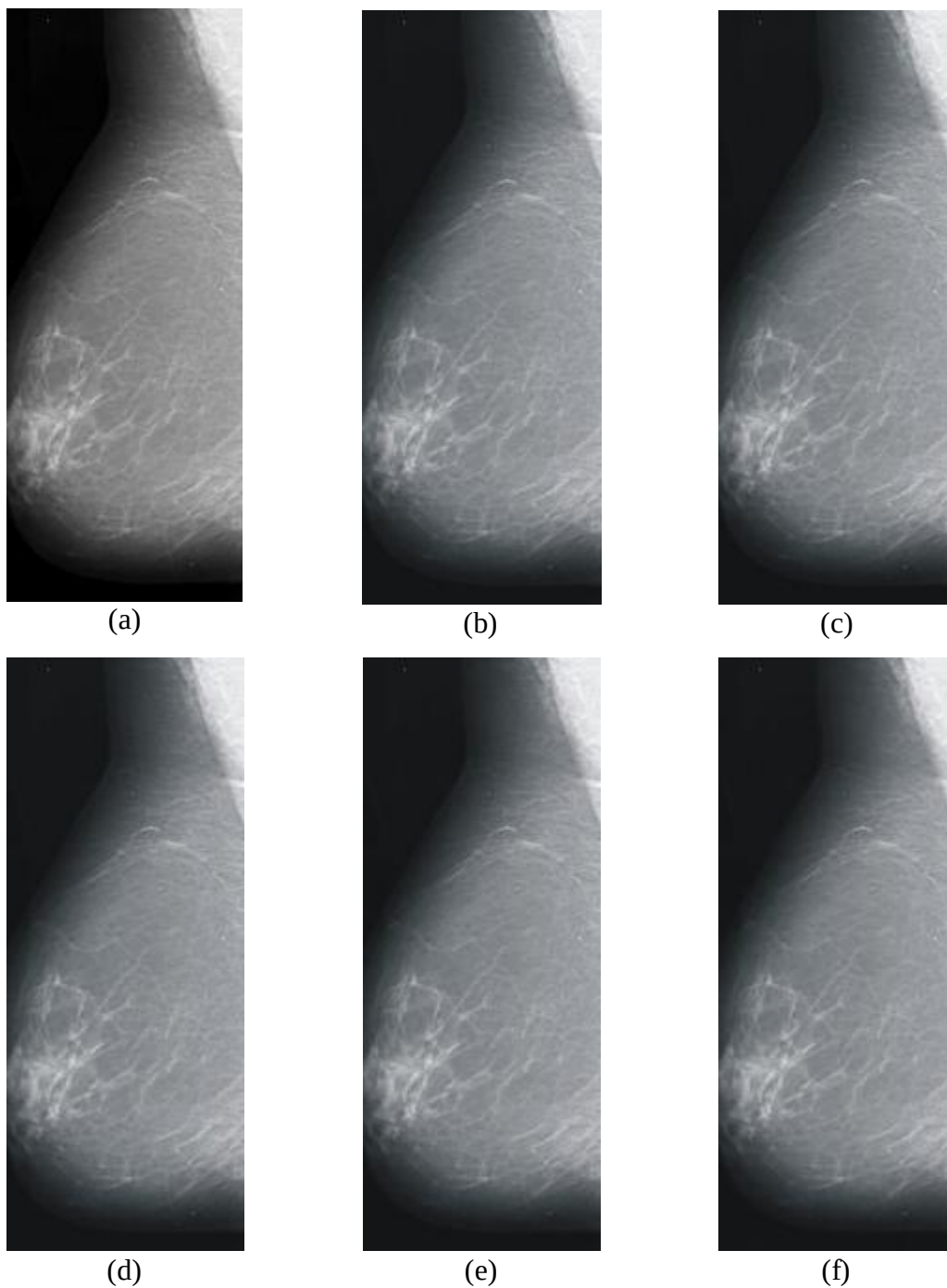


Figura 6.27 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,48 bpp; (c) 0,48 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

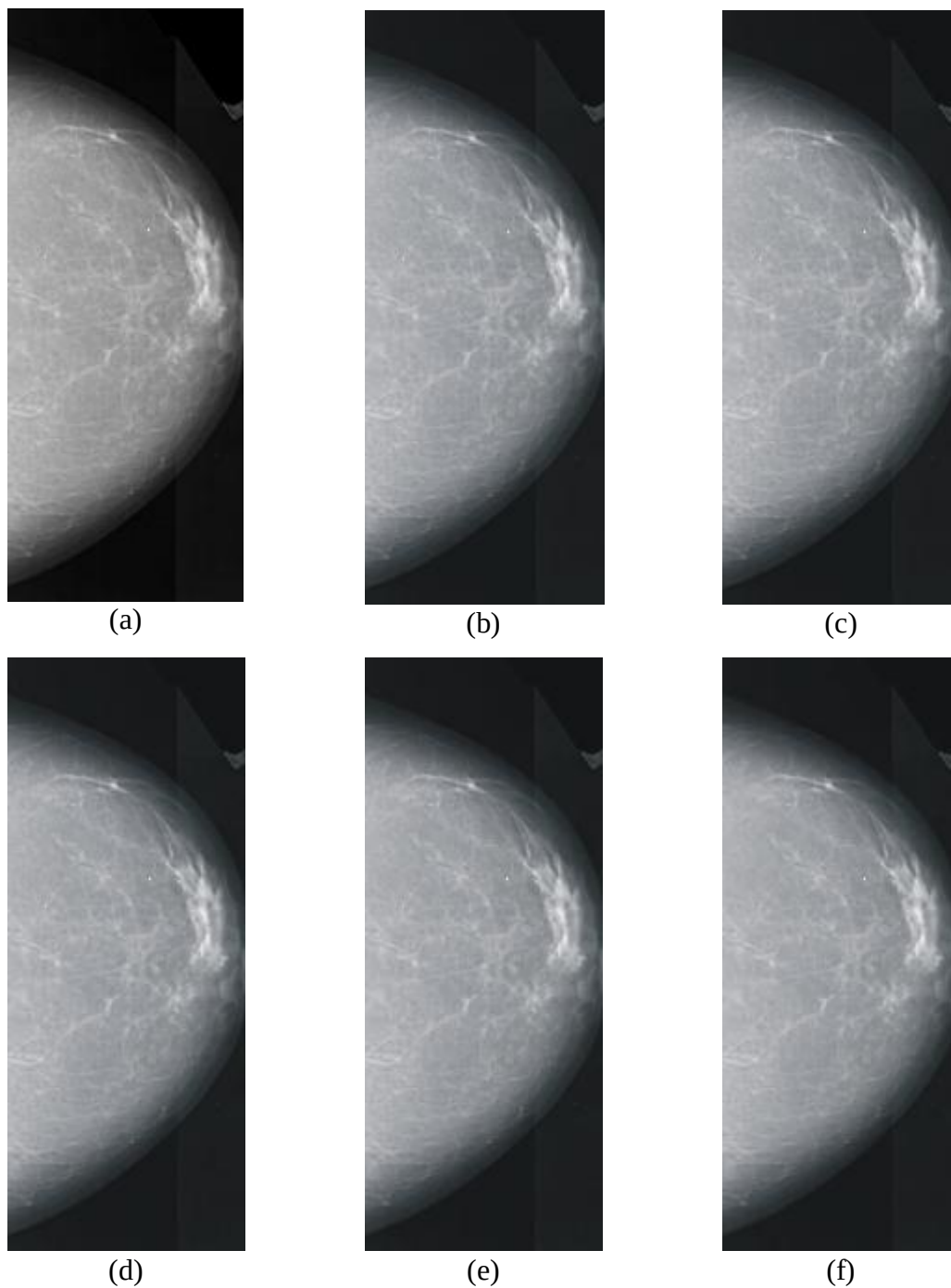


Figura 6.28 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

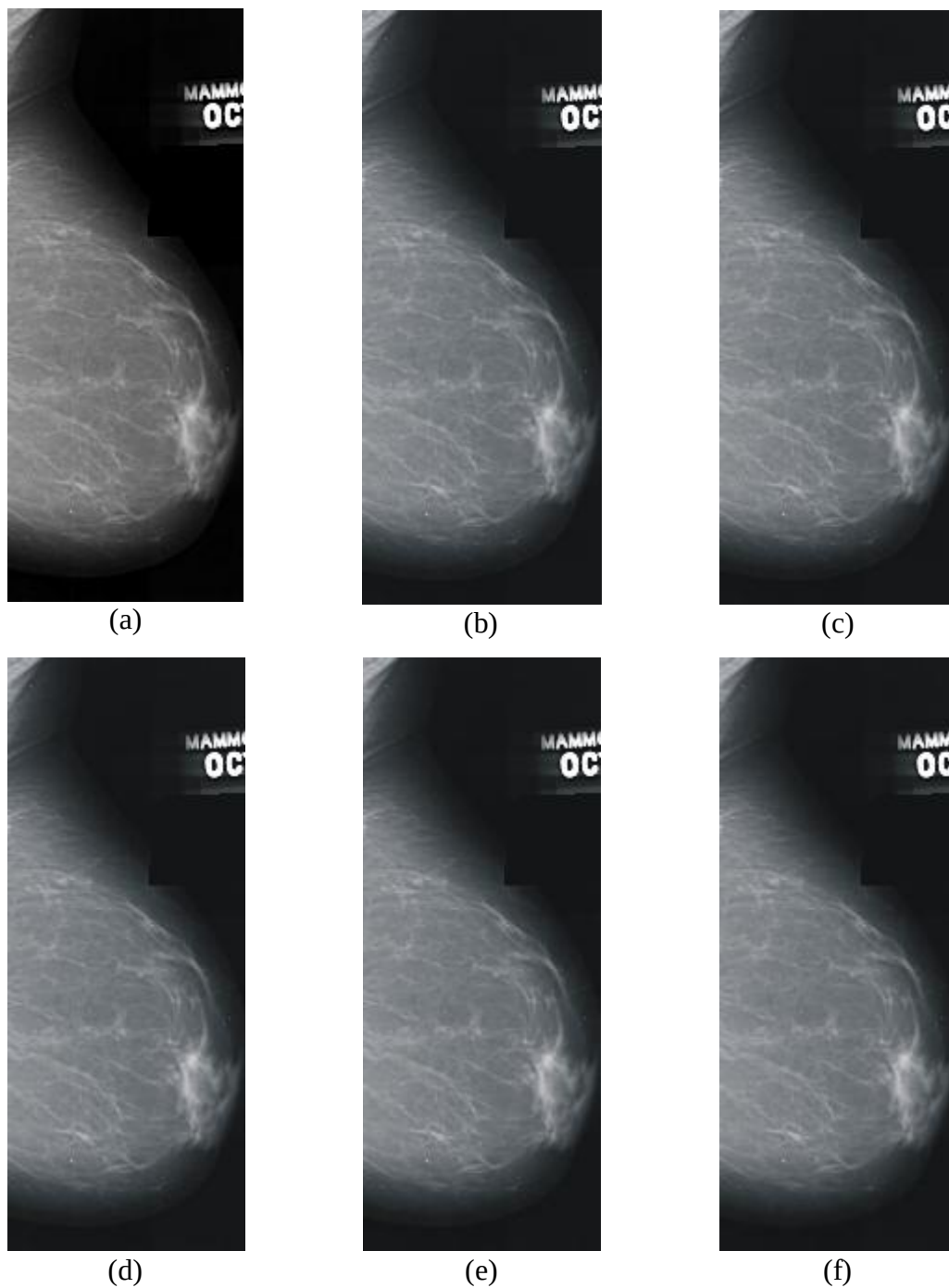


Figura 6.29 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

6.2.4 Caso 4

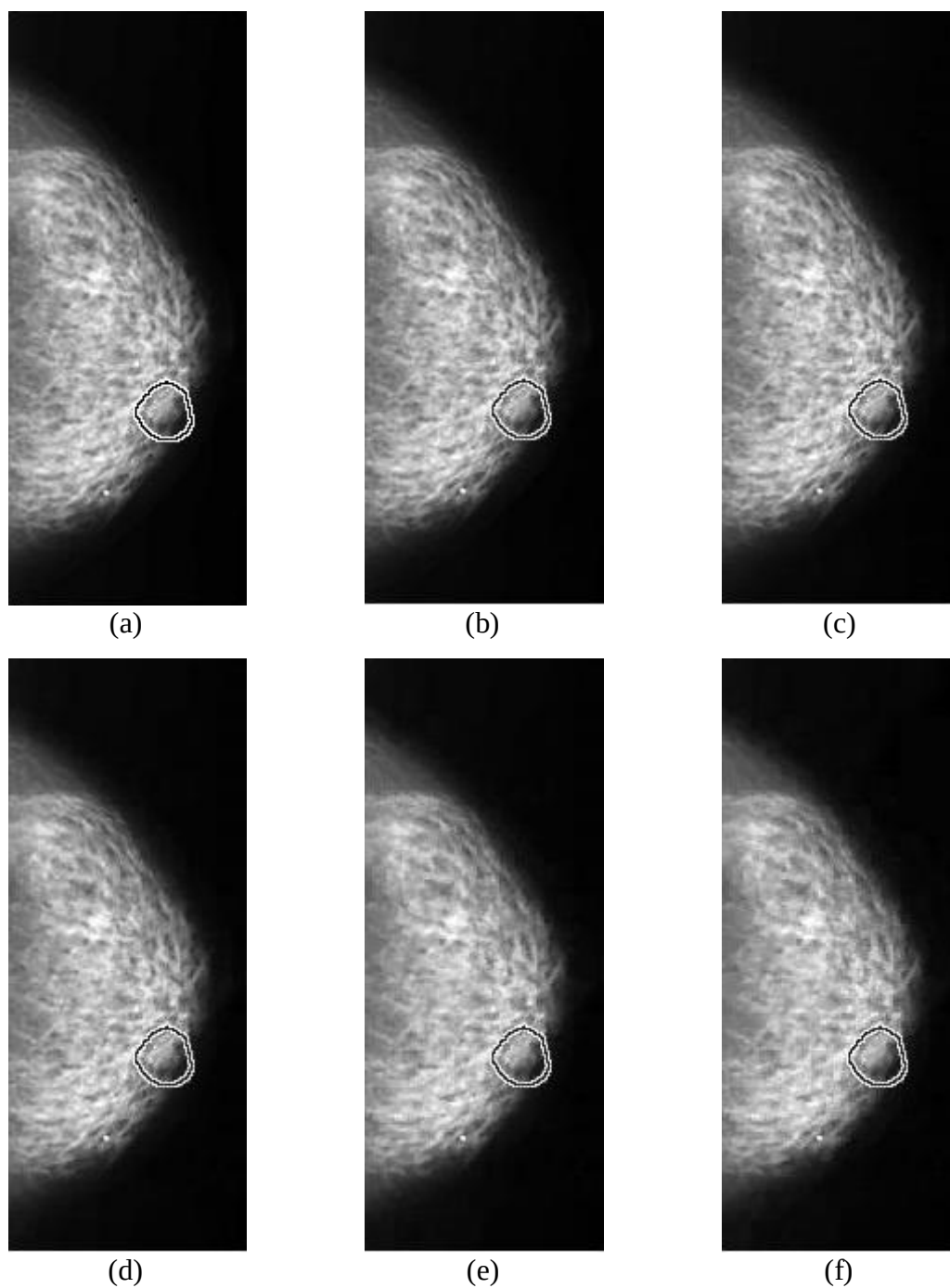


Figura 6.30 - Compressão fractal, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 1,8 bpp; (c) 1,48 bpp; (d) 1 bpp; (e) 0,53 bpp; (f) 0,43 bpp.

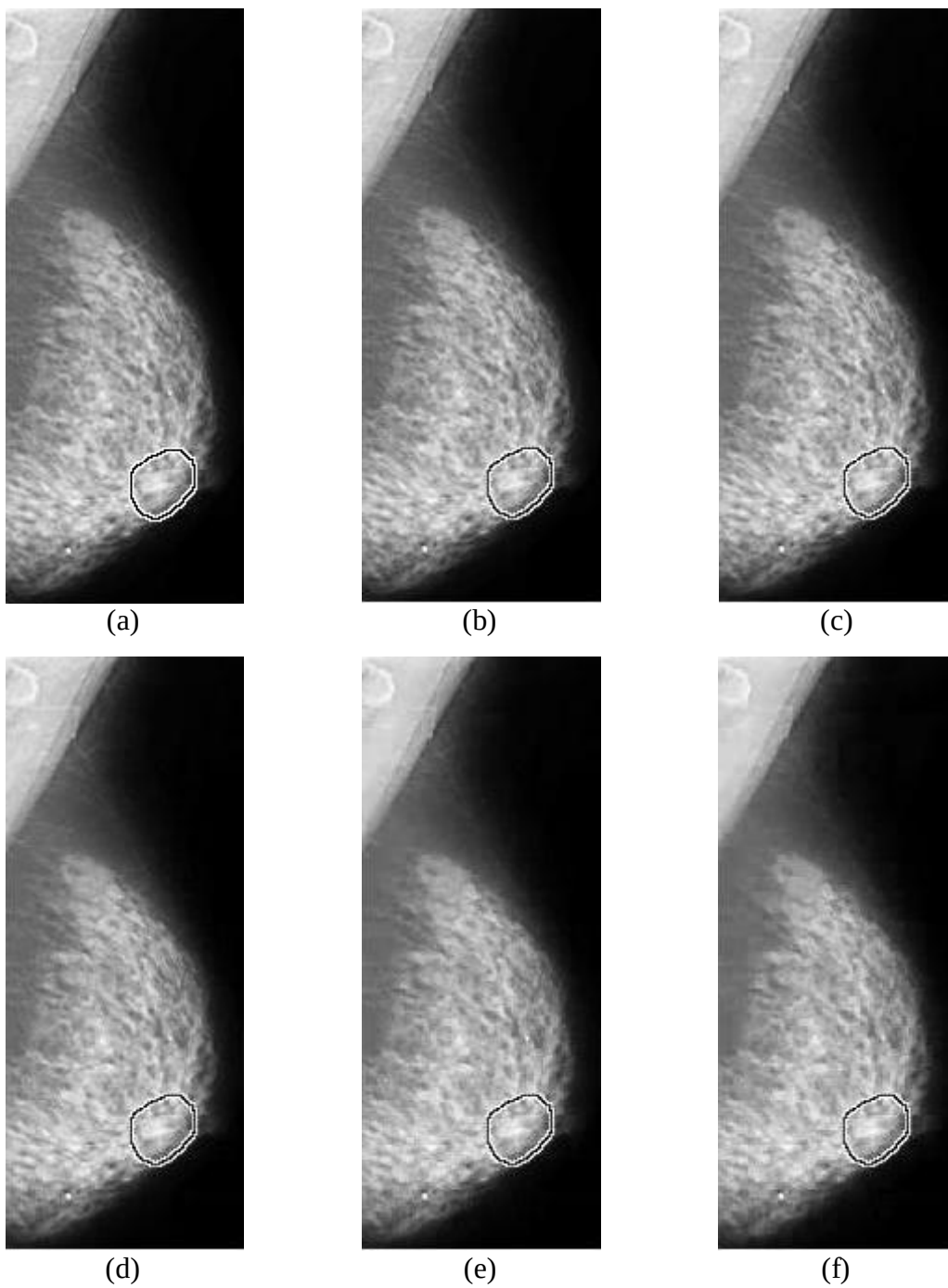


Figura 6.31 - Compressão fractal, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 2,22 bpp; (c) 1,63 bpp; (d) 1,02 bpp; (e) 0,56 bpp; (f) 0,48 bpp.

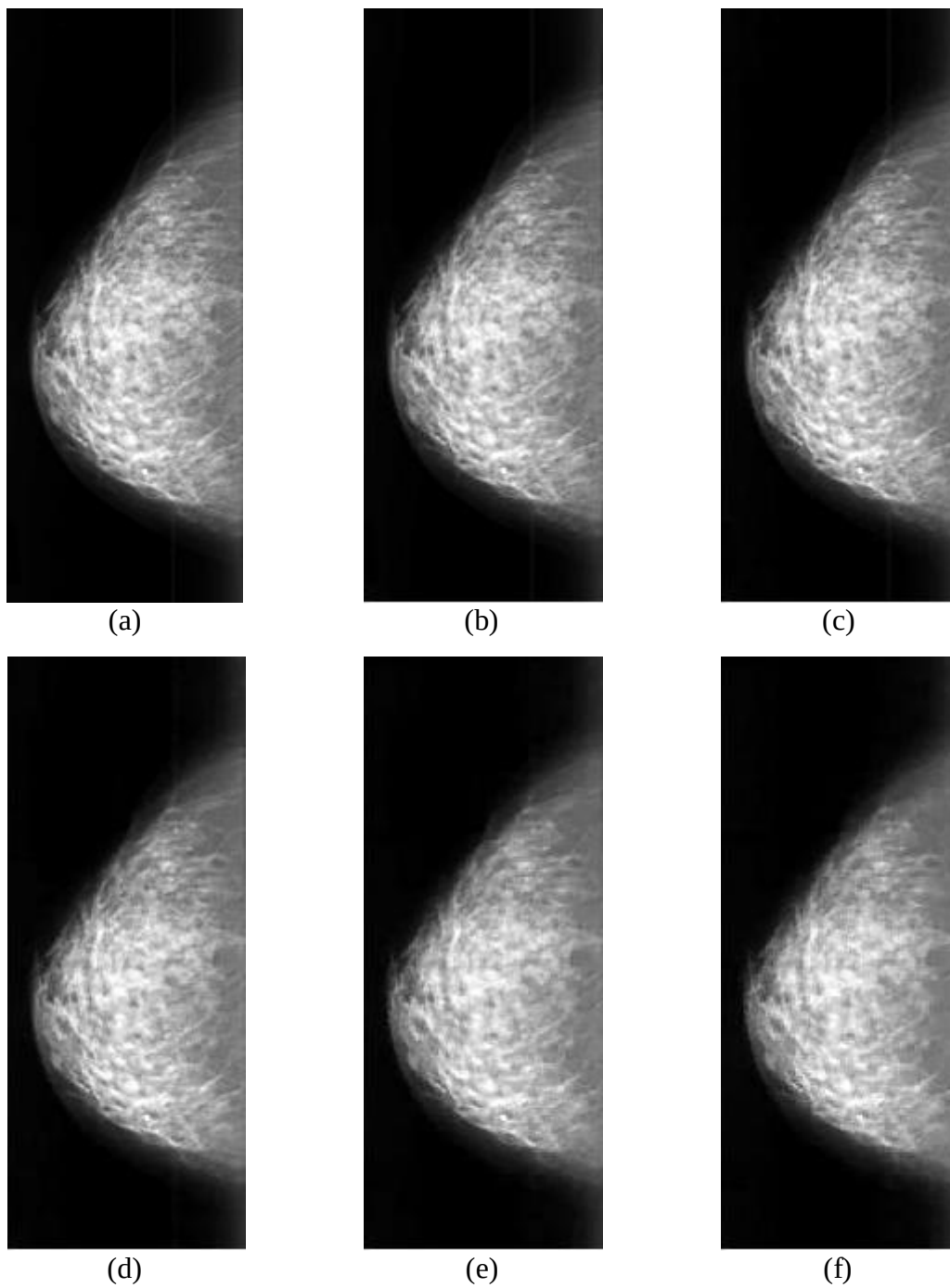


Figura 6.32 - Compressão fractal, seio direito, incidência CC (a) imagem original 8 bpp; (b) 1,8 bpp; (c) 1,36 bpp; (d) 0,87 bpp; (e) 0,46 bpp; (f) 0,36 bpp.

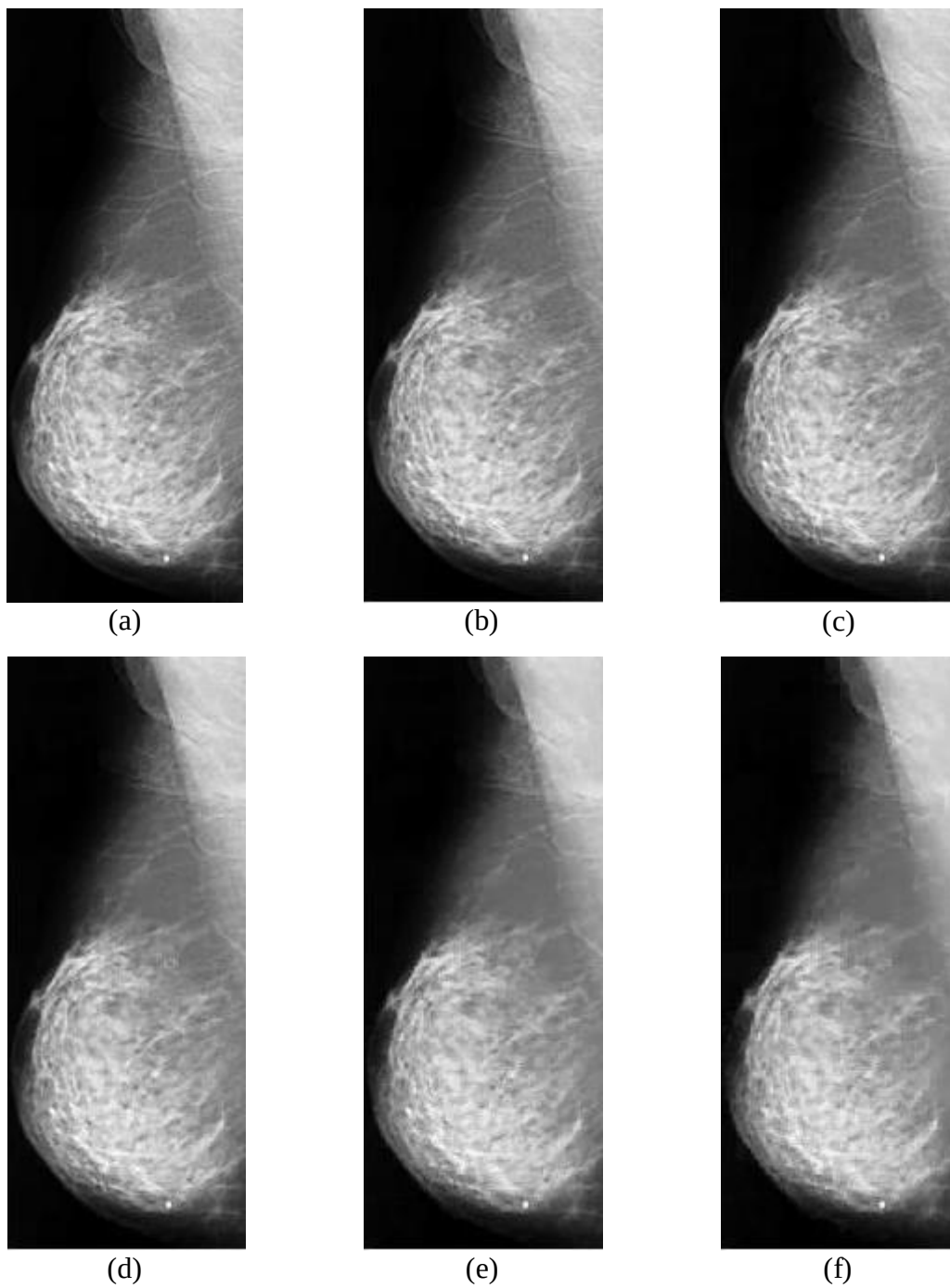


Figura 6.33 - Compressão fractal, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 2,39 bpp; (c) 1,53 bpp; (d) 1 bpp; (e) 0,56 bpp; (f) 0,41 bpp.

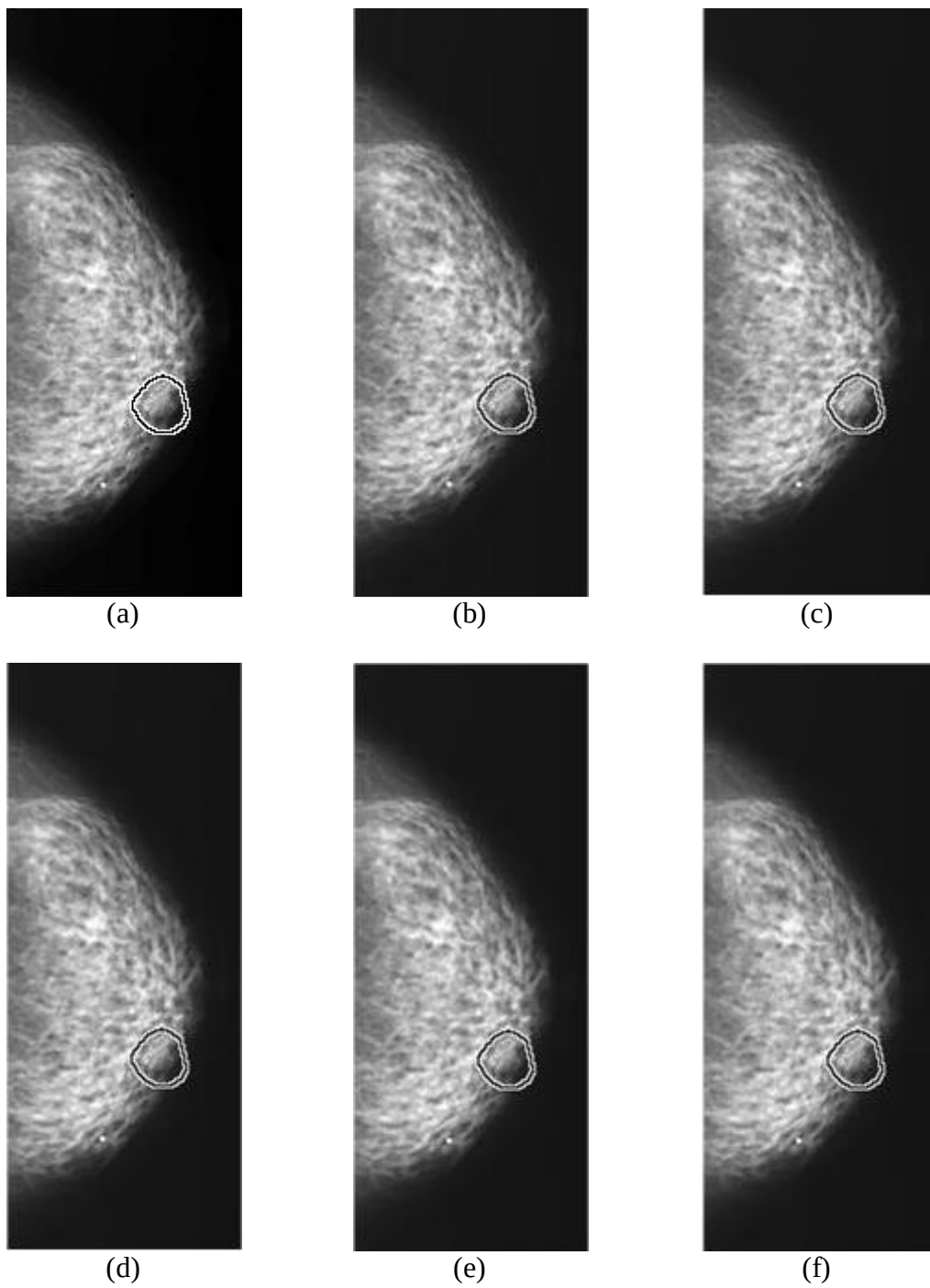


Figura 6.34 - Compressão JPEG2000, seio esquerdo, incidência CC (a) imagem original 8 bpp; (b) 0,46 bpp; (c) 0,46 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

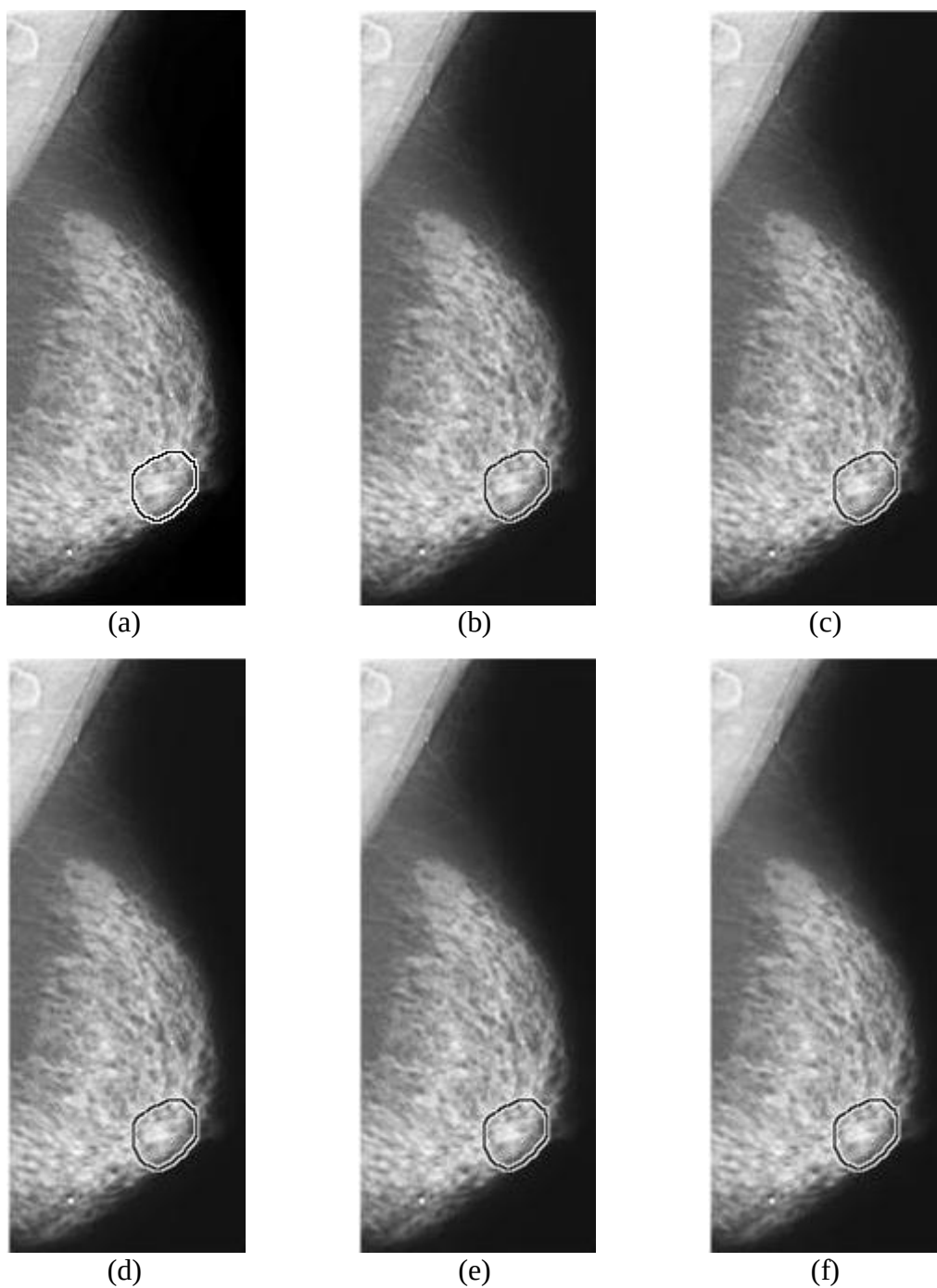


Figura 6.35 - Compressão JPEG2000, seio esquerdo, incidência MLO (a) imagem original 8 bpp; (b) 0,53 bpp; (c) 0,53 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

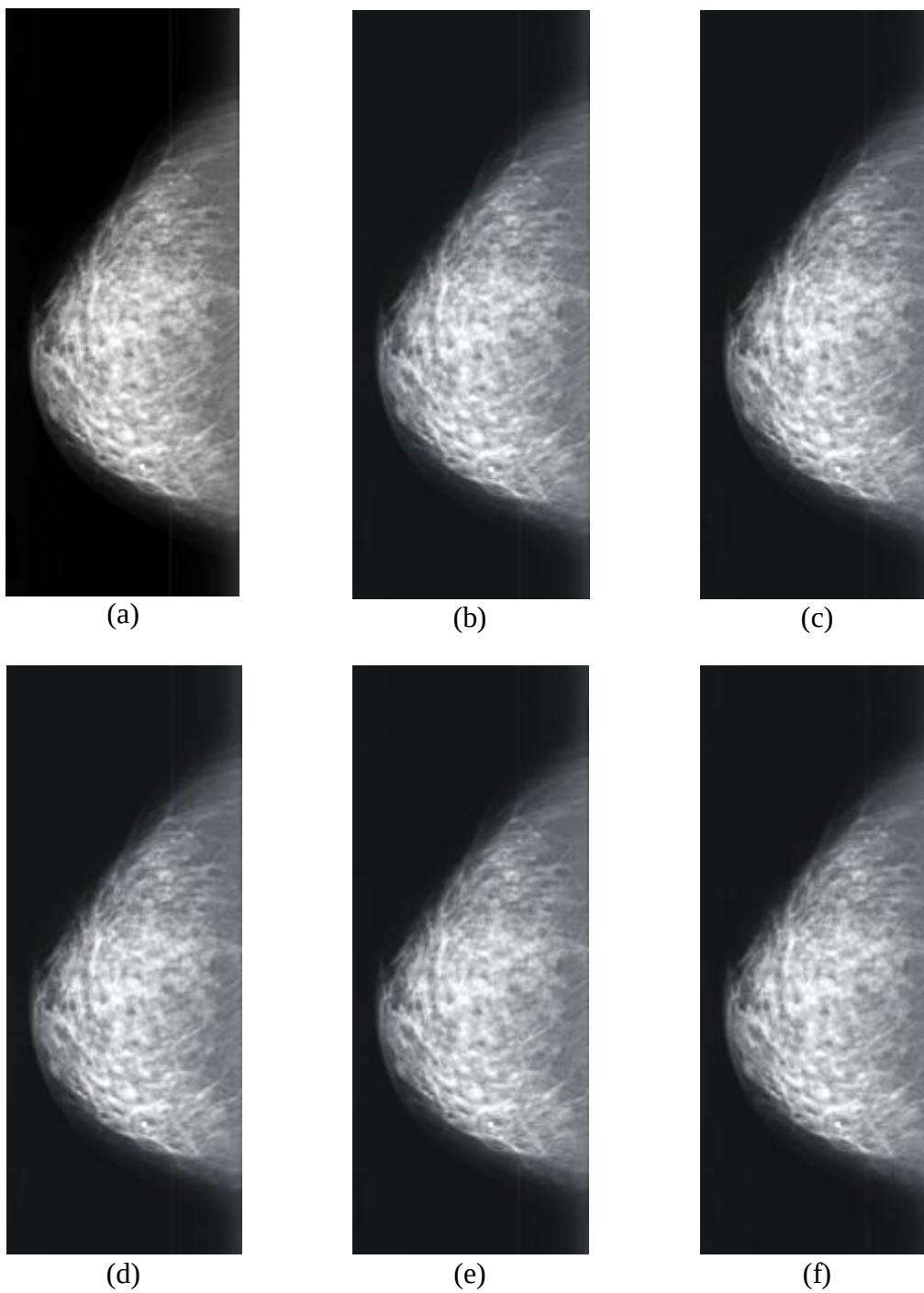


Figura 6.36 - Compressão JPEG2000, seio direito, incidência CC (a) imagem original 8 bpp; (b) 0,41 bpp; (c) 0,41 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

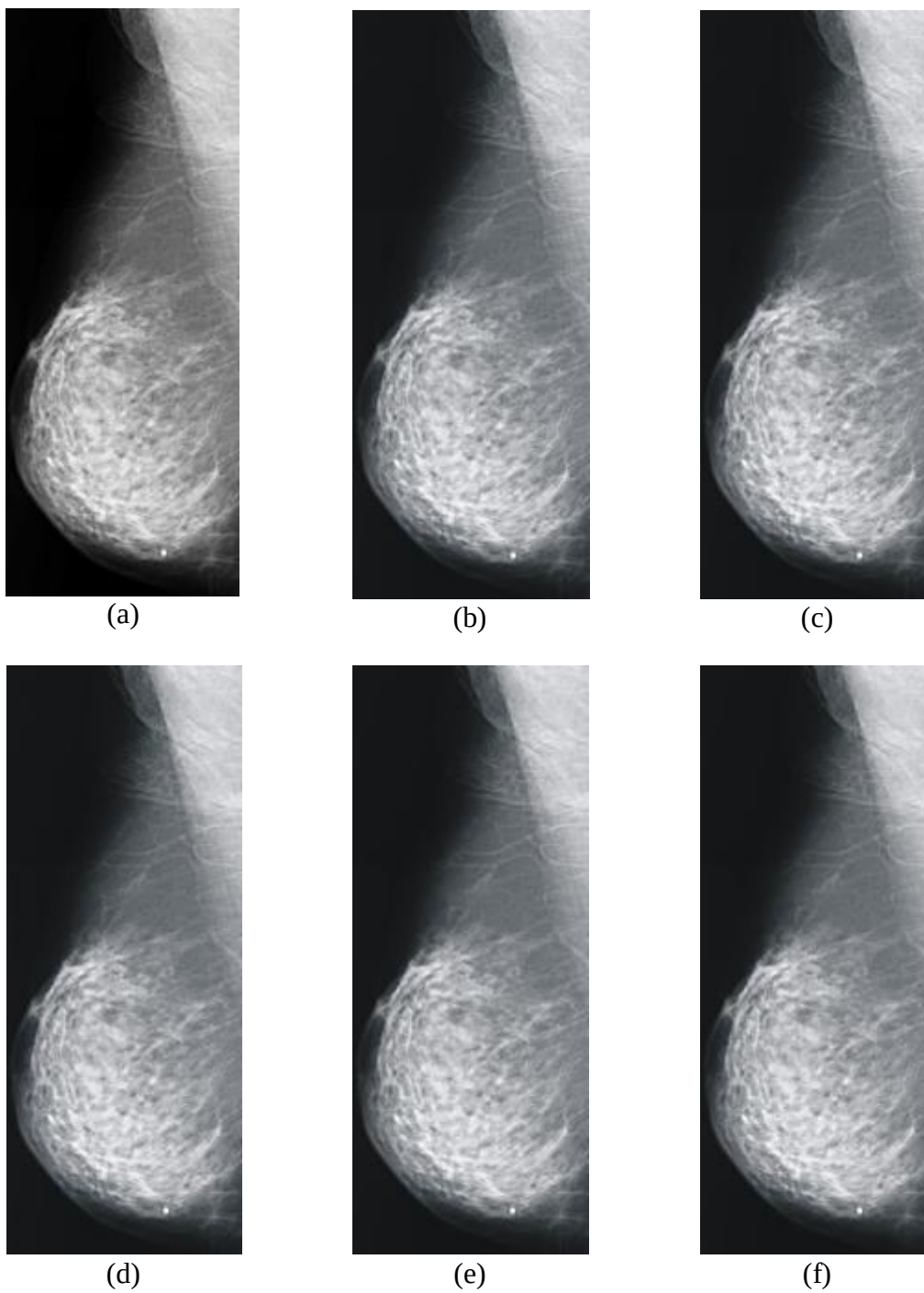


Figura 6.37 - Compressão JPEG2000, seio direito, incidência MLO (a) imagem original 8 bpp; (b) 0,51 bpp; (c) 0,51 bpp; (d) 0,29 bpp; (e) 0,19 bpp; (f) 0,14 bpp.

6.3 Análises dos Médicos Especialistas

Em imagens médicas, para definir se uma imagem comprimida pode ser considerada aceitável é necessário que um especialista da área analise essa imagem. Os especialistas não estão habituados a trabalhar com medidas objetivas (PSNR), dessa forma existe uma grande dificuldade em relacionar os critérios objetivo e subjetivo. Portanto, foi realizada neste trabalho uma associação entre esses critérios, para definir um limite em que as imagens mamográficas comprimidas utilizando o método da compressão fractal possam ser consideradas aceitáveis. Após isso, os resultados obtidos com esse método de compressão são comparados com os resultados utilizando o método da compressão JPEG2000.

As imagens originais e comprimidas utilizando os métodos compressão fractal e compressão JPEG2000, foram analisadas por 3 médicos. Foi pedido a cada um deles que analisasse cada uma das imagens comprimidas com diferentes taxas de compressão, classificando cada uma delas como ótima, boa, aceitável ou inaceitável. As imagens comprimidas foram nomeadas aleatoriamente para não interferir na análise e não foram mostradas a eles na ordem de compressão.

A princípio foi pedido aos médicos além de comparar as imagens comprimidas com as originais, que analisassem se na imagem original tinha sido encontrada alguma anormalidade, e se nas imagens comprimidas essa anormalidade podia ser também identificada. Segundo os médicos, anormalidades só podem ser apontadas com segurança se além das incidências básicas, forem também colocadas juntas durante a análise as incidências complementares. Outro comentário dos médicos foi que as imagens mamográficas devem ser lidas em monitores com tamanho e resolução específicos, onde eles possam alterar o contraste e o brilho para melhorar a identificação de possíveis anormalidades.

Cada uma das 16 imagens originais que possuem 8 bpp (bits por pixel), foram comprimidas com 5 diferentes taxas de compressão. A Tabela 6.1 mostra o valor médio em bpp de cada uma das 5 taxas de compressão utilizando a compressão fractal e a JPEG2000. Nesta tabela, o termo “1ª taxa de compressão” refere as imagens comprimidas com a menor taxa de compressão e o termo “5ª taxa de compressão” refere as imagens comprimidas com a maior taxa de compressão.

Tabela 6.1 – Valor em bpp das taxas de compressão.

Taxas de Compressão	Compressão Fractal (bpp)	Compressão JPEG2000 (bpp)
1ª taxa compressão	2	0,47
2ª taxa compressão	1,19	0,46
3ª taxa compressão	0,74	0,29
4ª taxa compressão	0,44	0,19
5ª taxa compressão	0,32	0,14

Pode-se observar na Tabela 6.1 que no método JPEG2000 consegue-se taxas de bits menores do que no método fractal.

Considerando os quatro casos, já que para cada imagem original, existem 5 imagens comprimidas, e para cada caso existem 16 imagens originais, os médicos compararam 80 imagens comprimidas com as originais para cada um dos dois métodos de compressão. Foi realizado um resumo das comparações entre as imagens comprimidas com diferentes taxas de compressão e as imagens originais analisadas pelos médicos. A Tabela 6.2 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método fractal, analisadas pelo médico 1, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.2 – Classificações do médico 1 (compressão fractal).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	-	-	16	-	16
2	-	6	10	-	16
1,19	-	15	1	-	16
0,74	4	12	-	-	16
0,44	14	2	-	-	16
0,32	16	-	-	-	16
TOTAL DE IMAGENS COMPRIMIDAS	34	35	11	-	80

A Tabela 6.3 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método JPEG2000, analisadas pelo médico 1, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.3 – Classificações do médico 1 (compressão JPEG2000).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	-	-	11	5	16
0,47	-	4	8	4	16
0,46	-	1	14	1	16
0,29	-	3	13	-	16
0,19	-	2	13	1	16
0,14	-	3	12	1	16
TOTAL DE IMAGENS COMPRIMIDAS	-	13	60	7	80

O médico 1 comentou que as imagens originais do banco de imagens não possuem o tamanho ideal para análise. Desse modo, ele classificou a maioria das imagens originais com qualidade “boa”. Na compressão fractal, ele classificou todas as imagens originais com qualidade “boa”. Na compressão JPEG2000, mesmo as imagens originais sendo as mesmas, 5 delas ele classificou com qualidade “ótima”. Considerando que as imagens comprimidas são comparadas com a imagem original, a qualidade delas pode ser classificada no máximo igual a classificação dada para a imagem original.

Pode-se observar na Tabela 6.2 (compressão fractal) que a partir da taxa média de bits de 0,74 bpp, o médico classificou algumas imagens comprimidas como “inaceitável”. Na taxa média de 0,32 bpp todas as imagens comprimidas foram consideradas “inaceitáveis”. Dentre as 34 imagens consideradas “inaceitáveis”, a média da taxa de bits foi de 0,41 bpp e o valor médio da PSNR foi de 33,67 dB. Das 35 imagens consideradas “aceitáveis”, a média taxa de bits foi de 1,16 bpp e o valor médio da PSNR foi de 38,61 dB. Das 11 imagens classificadas com qualidade “boa”, a taxa média de bits foi de 1,88 bpp e o valor médio da PSNR foi de 44,5 dB.

Ao analisar a Tabela 6.3 (compressão JPEG2000) percebe-se que ocorreu uma dificuldade maior para classificar as imagens e apareceram imagens com taxa de bits menor, e consequentemente qualidade pior, sendo mais bem classificadas do que imagens com taxa maior de bits e consequentemente qualidade melhor. Dentre as 13 imagens consideradas “aceitáveis”, a taxa média de bits foi de 0,29 bpp e o valor médio da PSNR foi de 54,92 dB. Das 60 imagens consideradas “boas”, a taxa média de bits foi de 0,30 bpp e o valor médio da PSNR foi de 56,89 dB. Das 7 imagens classificadas com qualidade “ótima”, a taxa média de bits foi de 0,38 bpp e o valor médio da PSNR foi de 69,77 dB.

A Tabela 6.4 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método fractal, analisadas pelo médico 2, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.4 – Classificações do médico 2 (compressão fractal).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	-	11	5	-	16
2	2	9	5	-	16
1,19	7	8	1	-	16
0,74	12	4	-	-	16
0,44	16	-	-	-	16
0,32	16	-	-	-	16
TOTAL DE IMAGENS COMPRIMIDAS	53	21	6	-	80

A Tabela 6.5 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método JPEG2000, analisadas pelo médico 2, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.5 – Classificações do médico 2 (compressão JPEG2000).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	2	14	-	-	16
0,47	1	15	-	-	16
0,46	1	15	-	-	16
0,29	6	10	-	-	16
0,19	7	9	-	-	16
0,14	9	7	-	-	16
TOTAL DE IMAGENS COMPRIMIDAS	24	56	-	-	80

Assim como o médico 1, o médico 2 comentou que o tamanho das imagens não é apropriado para a análise de possíveis anormalidades. Observando as Tabela 6.4 e 6.5, o médico 2 classificou a maioria das imagens originais como “aceitáveis”. Na classificação dessas imagens originais ocorreram algumas divergências entre a compressão fractal e a compressão JPEG2000, apesar delas serem as mesmas. Na compressão fractal, das 16 imagens originais, 5 foram classificadas com qualidade “boa” e na compressão JPEG2000, 2 imagens originais foram classificadas como “inaceitáveis”. Como as imagens comprimidas são comparadas com a imagem original, a qualidade delas pode ser classificada no máximo igual a classificação feita para a imagem original.

Na Tabela 6.4 (compressão fractal), como a maioria das imagens originais foram classificadas como “aceitáveis”, grande parte das imagens comprimidas foram classificadas como “inaceitáveis”. A partir da taxa média de bits de 0,44 bpp todas as imagens comprimidas foram classificadas como “inaceitáveis”. Dentre as 53 imagens consideradas “inaceitáveis”, a taxa média de bits foi de 0,63 bpp e o valor médio da PSNR foi de 34,98 dB. Das 21 imagens consideradas “aceitáveis”, a taxa média de bits foi de 1,44 bpp e o valor médio da PSNR foi de 41,74 dB. Das 6 imagens classificadas com qualidade “boa”, a taxa média de bits foi de 1,97 bpp e o valor médio da PSNR foi de 42,58 dB.

Pode-se observar na Tabela 6.5 (compressão JPEG2000) que todas as taxas de compressão tiveram imagens comprimidas classificadas como “inaceitáveis”. A partir da taxa de bits de 0,29 bpp a quantidade de imagens comprimidas consideradas “inaceitáveis” foi

maior. Dentre as 24 imagens consideradas “inaceitáveis”, a taxa média de bits foi de 0,22 bpp e o valor médio da PSNR foi de 45,96 dB. Das 56 imagens classificadas com qualidade “aceitável”, a taxa média de bits foi de 0,35 bpp e o valor da PSNR foi de 62,73 dB.

Assim como ocorreu com o médico 1, o médico 2 classificou imagens com taxa de bits menor, e consequentemente qualidade pior, melhor do que imagens com taxa maior de bits e consequentemente qualidade melhor, confirmando uma maior dificuldade na classificação das imagens comprimidas pelo método JPEG2000.

A Tabela 6.6 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método fractal, analisadas pelo médico 3, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.6 – Classificações do médico 3 (compressão fractal).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	-	-	16	-	16
2	1	11	4	-	16
1,19	3	12	1	-	16
0,74	5	11	-	-	16
0,44	13	3	-	-	16
0,32	16	-	-	-	16
TOTAL DE IMAGENS COMPRIMIDAS	38	37	5	-	80

A Tabela 6.7 mostra o resumo das classificações das imagens originais (8 bpp) e das imagens comprimidas com diferentes taxas de compressão, utilizando o método JPEG2000, analisadas pelo médico 3, mostrando a quantidade de imagens que obtiveram cada uma das classificações.

Tabela 6.7 – Classificações do médico 3 (compressão JPEG2000).

Taxa de bits (bpp)	Inaceitável	Aceitável	Boa	Ótima	TOTAL
8 (imagens originais)	-	-	16	-	16
0,47	7	7	2	-	16
0,46	7	5	4	-	16
0,29	8	8	-	-	16
0,19	7	8	1	-	16
0,14	9	4	3	-	16
TOTAL DE IMAGENS COMPRIMIDAS	38	32	10	-	80

Na Tabela 6.6 (compressão fractal) a maior parte das imagens foi classificada como “inaceitáveis” ou “aceitáveis”. A partir taxa média de 0,44 bpp a quantidade de imagens comprimidas consideradas “inaceitáveis” foi bem maior e na taxa média de 0,32 bpp todas as imagens comprimidas foram consideradas “inaceitáveis”. Dentre as 38 imagens consideradas “inaceitáveis”, a taxa média de bits foi de 0,49 bpp e o valor médio da PSNR foi de 34,64 dB. Das 37 imagens consideradas “aceitáveis”, a taxa média de bits foi de 1,27 bpp e o valor médio da PSNR foi de 38,82 dB. Das 5 imagens classificadas com qualidade “boa”, a taxa média de bits foi de 1,97 bpp e o valor médio da PSNR foi de 46,56 dB.

Pode-se verificar na Tabela 6.7 (compressão JPEG2000) que a maior parte das imagens também foi classificada como “inaceitáveis” ou “aceitáveis”. Dentre as 38 imagens consideradas “inaceitáveis”, a taxa média de bits foi de 0,30 bpp e o valor médio da PSNR foi de 56,84 dB. Das 32 imagens consideradas “aceitáveis”, a taxa média de bits foi de 0,31 bpp e o valor médio da PSNR foi de 56,9 dB. Das 10 imagens classificadas com qualidade “boa”, a taxa média de bits foi de 0,32 bpp e o valor médio da PSNR foi de 67,21 dB.

Com a finalidade de melhorar a análise dos resultados obtidos a Tabela 6.8 mostra o resumo das classificações realizadas por cada médico para as 80 imagens comprimidas utilizando o método fractal, mostrando a quantidade de imagens classificadas em cada nível de classificação.

Tabela 6.8 – Resumo da análise dos 3 médicos quando as imagens foram comprimidas utilizando a compressão fractal.

	Médico 1	Médico 2	Médico 3	TOTAL
Inaceitável	34	53	38	125
Aceitável	35	21	37	93
Boa	11	6	5	22
Ótima	-	-	-	-
TOTAL	80	80	80	240

Pode-se observar na Tabela 6.8 que do total de imagens analisadas, os médicos classificaram, em média, 47,92% das imagens comprimidas com diferentes taxas de compressão como “aceitáveis” ou “boas” e 52,08% das imagens comprimidas como “inaceitáveis”.

Com o objetivo de melhorar a análise dos resultados obtidos nos testes realizados, a Tabela 6.9 mostra o resumo das classificações realizadas por cada médico para as 80 imagens comprimidas utilizando o método JPEG2000, mostrando a quantidade de imagens classificadas para cada nível de classificação.

Tabela 6.9 – Resumo da análise dos 3 médicos quando as imagens foram comprimidas utilizando a compressão JPEG2000.

	Médico 1	Médico 2	Médico 3	TOTAL
Inaceitável	-	24	38	62
Aceitável	13	56	32	101
Boa	60	-	10	70
Ótima	7	-	-	7
TOTAL	80	80	80	240

Pode-se verificar na Tabela 6.9 que do total de imagens analisadas, os médicos classificaram, em média, 71,25% das imagens comprimidas com diferentes taxas de compressão como “aceitáveis” ou “boas” e 25,83% das imagens comprimidas como “inaceitáveis”.

Observando as Tabela 6.8 e Tabela 6.9, verifica-se que a quantidade de imagens consideradas “inaceitáveis” foi maior utilizando o método fractal (52,08%) do que ao utilizar o método JPEG2000 (25,83%).

O objetivo deste trabalho é combinar os critérios objetivos comumente utilizados na avaliação da qualidade das imagens comprimidas com os critérios visuais utilizados pelos médicos especialistas. Assim, combinou-se os valores das taxas de bits com a classificação dos médicos para definir o limite de compressão, ou seja, até que taxa de bits as imagens comprimidas podem ser consideradas aceitáveis.

A Tabela 6.10 mostra o resumo das classificações de cada um dos médicos especialistas, quando as imagens foram comprimidas utilizando o método fractal, indicando a média da taxa de bits para cada nível de classificação.

Tabela 6.10 – Média da taxa de bits do método fractal.

	Médico 1 (bpp)	Médico 2 (bpp)	Médico 3 (bpp)
Inaceitável	0,41	0,63	0,49
Aceitável	1,16	1,44	1,27
Boa	1,88	1,97	1,97
Ótima	-	-	-

Pode-se observar na Tabela 6.10 que imagens com taxa de bits menores que 0,63 bpp são consideradas “inaceitáveis”.

A Tabela 6.11 mostra a média do valor de PSNR para cada nível de classificação para as imagens comprimidas pelo método fractal.

Tabela 6.11 – Média do valor de PSNR do método fractal.

	Médico 1 (dB)	Médico 2 (dB)	Médico 3 (dB)
Inaceitável	33,67	38,82	34,64
Aceitável	38,61	41,74	38,82
Boa	44,50	42,58	46,56
Ótima	-	-	-

Analisando as Tabela 6.10 e Tabela 6.11 e tomando o menor valor entre a classificação dos três médicos para imagens consideradas “aceitáveis” ou com classificação superior, considera-se que são “aceitáveis” as imagens comprimidas pelo método Fractal com taxa de bits de no mínimo 1,16 bpp e um valor de PSNR de no mínimo 38,61 dB.

A Tabela 6.12 mostra o resumo das classificações de cada um dos médicos especialistas, quando as imagens foram comprimidas utilizando o método JPEG2000, indicando a média da taxa de bits para cada nível de classificação.

Tabela 6.12 – Média da taxa de bits do método JPEG2000.

	Médico 1 (bpp)	Médico 2 (bpp)	Médico 3 (bpp)
Inaceitável	-	0,22	0,30
Aceitável	0,29	0,35	0,31
Boa	0,30	-	0,32
Ótima	0,38	-	-

Pode-se observar na Tabela 6.12 que imagens com taxa de bits menores que 0,30 bpp são consideradas “inaceitáveis”.

A Tabela 6.13 mostra a média do valor de PSNR para cada nível de classificação para as imagens comprimidas pelo método JPEG2000.

Tabela 6.13 – Média do valor de PSNR do método JPEG2000.

	Médico 1	Médico 2	Médico 3
Inaceitável	-	45,96	56,84
Aceitável	54,92	62,73	56,90
Boa	56,89	-	67,21
Ótima	69,77	-	-

Analisando as Tabela 6.12 e Tabela 6.13 e tomando o menor valor entre a classificação dos três médicos para imagens consideradas “aceitáveis” ou com classificação superior, considera-se que são “aceitáveis” as imagens comprimidas pelo método JPEG2000 com taxa de bits de no mínimo 0,29 bpp e um valor de PSNR de no mínimo 54,92 dB.

Com os resultados mostrados nas Tabela 6.10 a Tabela 6.13, pode-se combinar os critérios objetivo e subjetivo na classificação de imagens comprimidas utilizando o método fractal e definir o limite de taxa de bits para que as imagens sejam consideradas “aceitáveis” pelos médicos especialistas. A Tabela 6.14 mostra os limites da taxa de bits e o valor da PSNR para imagens mamográficas comprimidas utilizando os métodos fractal e JPEG2000.

Tabela 6.14 – Limite da taxa de bits e valor da PSNR para imagens mamográficas comprimidas utilizando os métodos fractal e JPEG2000.

Medidas de desempenho	Fractal	JPEG2000
Taxa de bits (bpp)	1,16	0,29
PSNR (dB)	38,61	54,92

Pode-se verificar na Tabela 6.14 que para as imagens comprimidas utilizando o método fractal, o limite da taxa de bits para que as imagens sejam consideradas “aceitáveis” é de 1,16 bpp com uma PSNR de 38,61 dB. Quando utilizou-se o método JPEG2000 o limite da taxa de bits para que as imagens sejam consideradas “aceitáveis” é de 0,29 bpp com uma PSNR de 54,92 dB. Assim, a compressão utilizando o método JPEG2000 resulta em imagens com qualidade melhor e taxas de bits menores do que utilizando o método fractal.

6.4 Considerações Finais do Capítulo

Este capítulo mostrou a importância da mamografia no rastreamento do câncer de mama e as principais vistas da mama que são obtidas com este exame. Foram utilizadas algumas imagens mamográficas e as mesmas foram comprimidas pelos métodos fractal e JPEG2000. As imagens originais e comprimidas foram analisadas por especialistas e foi feita uma combinação entre os critérios objetivo e subjetivo utilizando os resultados obtidos nessas análises.

Com isso, estabelece-se um limite para a taxa de bits dessas imagens comprimidas para que elas sejam consideradas “aceitáveis” pelos especialistas. Para o método fractal o limite da taxa de bits para que as imagens sejam consideradas “aceitáveis” é de 1,16 bpp com uma PSNR de 38,61 dB e para o método JPEG2000 o limite da taxa de bits para que as imagens sejam consideradas “aceitáveis” é de 0,29 bpp com uma PSNR de 54,92 dB.

O próximo capítulo apresenta as conclusões e contribuições deste trabalho e os trabalhos futuros que poderão ser desenvolvidos a partir desta dissertação.

CAPÍTULO 7

CONCLUSÕES, CONTRIBUIÇÕES E SUGESTÕES PARA TRABALHOS FUTUROS

7.1 Introdução

Este capítulo apresenta as conclusões e contribuições deste trabalho, e as sugestões para trabalhos futuros que poderão ser desenvolvidos a partir desta dissertação.

7.2 Conclusões

A utilização de imagens em diagnósticos médicos é bastante ampla. O volume de imagens e a quantidade de informação gerada pelas imagens médicas acabam sendo um problema em relação a armazenamento ou transmissão dessas informações. A utilização de técnicas de compressão de dados pode ser muito eficiente nesses casos. Porém, nas imagens médicas, deve-se ter um cuidado em relação a perda de informações importantes. Por isso é importante a combinação das medidas objetivas, obtidas pelos métodos de análise de desempenho (taxa de bits e PSNR), com as medidas subjetivas, obtidas pela análise de médicos especialistas.

Neste trabalho foram implementados em Matlab algoritmos da força bruta e da partição quadtree. Utilizando o método da força bruta, foram realizados testes com a imagem da "Lena" 64 x 64 pixels, variando os parâmetros de entrada e os resultados foram mostrados

nas Tabela 4.1 a Tabela 4.3. Os resultados obtidos confirmaram o que foi visto na bibliografia, quanto maior é a sobreposição na etapa de construção do *Domain Pool* e quanto menor é o tamanho dos *range blocks*, maior é o PSNR da imagem reconstruída.

Ao utilizar o método da partição quadtree, foram realizados testes com a imagem da "Lena" 64 x 64 pixels, variando os parâmetros de entrada e os resultados foram mostrados nas Tabela 4.4 aTabela 4.8. Os resultados mostraram que quanto menor é o limiar de erro e quanto maior é o nível de partição, que são parâmetros que estão diretamente ligados ao tamanho dos blocos, maior é a PSNR da imagem reconstruída. Os resultados desses testes confirmaram o que foi visto na bibliografia.

Com a finalidade de comparação, as imagens mamográficas escolhidas para os testes de compressão foram comprimidas utilizando os métodos fractal e JPEG2000. Foram realizados testes com a imagem "Lena" 256 x 256 pixels, utilizado o software *Jasper*, desenvolvido por *Michael Adams*. Ao comparar os valores da PSNR do método fractal e do método JPEG2000, verificou-se que o método JPEG2000 consegue atingir maiores taxas de compressão mantendo uma boa qualidade da imagem comprimida.

Algumas imagens mamográficas originais foram retiradas de um banco digital de imagens, *Digital Database for Screening Mammography*, disponibilizado para pesquisa. Foram realizados testes de compressão de imagens utilizando os métodos fractal e JPEG2000.

Para o método de compressão fractal, foi desenvolvido um algoritmo em Matlab baseado no método da partição quadtree desenvolvido por *Fischer*, utilizando como dados de entrada: sobreposição de 25% na etapa de construção do *Domain Pool*, nível máximo de partição = 6, quantidade de bits para representar o contraste = 5, quantidade de bits para representar o brilho = 7 e máximo valor para contraste = 1,2. Cada uma das imagens originais foi comprimida com cinco taxas de compressão diferentes. As taxas foram obtidas alterando-se apenas um dos possíveis parâmetros que é o limiar de erro RMS. Foram utilizados cinco valores de limiar de erro RMS: 10, 20, 30, 50 e 70, para obtenção das cinco diferentes taxas de compressão. Quanto menor o limiar de erro utilizado, maior é a taxa de bits da imagem comprimida e melhor é a qualidade da imagem reconstruída e consequentemente, quanto maior o limiar de erro utilizado, menor é a taxa de bits da imagem comprimida e pior a qualidade da imagem reconstruída.

As mesmas imagens originais foram comprimidas pelo método JPEG2000. Para isso, foi utilizado o *software Jasper*, desenvolvido por *Michael Adams*. As imagens originais foram comprimidas também com cinco diferentes taxas de compressão que são parâmetros de entrada do *Jasper*. Assim como no método de Compressão Fractal, quanto menor é a taxa de compressão, maior é a taxa de bits da imagem comprimida e melhor é a qualidade da imagem reconstruída e quanto maior a taxa de compressão, menor é a taxa de bits da imagem comprimida e pior é a qualidade da imagem reconstruída.

Todas as imagens comprimidas foram mostradas a três médicos especialistas, e foi realizada uma combinação entre as medidas objetivas, que é o valor de PSNR, e os critérios subjetivos, que são as análises visuais dos médicos especialistas. Na análise visual, os médicos indicaram se cada imagem era considerada ótima, boa, aceitável ou inaceitável. Dessa maneira definiu-se até que taxa de compressão as imagens comprimidas são consideradas aceitáveis. A Tabela 6.8 e a Tabela 6.9 mostraram um resumo das classificações dos médicos para as imagens comprimidas pelo método de Compressão Fractal e pelo método de Compressão JPEG2000, respectivamente. Verificou-se que 47,92 % das imagens comprimidas pelo método fractal foram consideradas aceitáveis ou boas, já no método JPEG2000 um maior número de imagens foi considerada aceitável (74,17 %), como já era esperado. A Tabela 6.10 e a Tabela 6.11 mostraram o resumo das classificações feitas por cada médico, indicando a média da taxa de bits (bpp) e a média do valor da PSNR para cada nível de classificação, respectivamente, para o método fractal. A Tabela 6.12 e a Tabela 6.13 mostraram o resumo das classificações feitas por cada médico, indicando a média da taxa de bits (bpp) e a média do valor da PSNR para cada nível de classificação, respectivamente, para o método JPEG2000.

Pela análise das Tabelas 6.10 a 6.13 foi realizada a combinação entre os critérios objetivo e subjetivo na classificação de imagens comprimidas. Com essa combinação definiu-se o limite da taxa de bits para as imagens serem consideradas aceitáveis pelos médicos especialistas. A Tabela 6.14 mostrou a comparação desses limites entre os métodos fractal e JPEG2000. Para o método fractal, o limite de taxa de bits foi de 1,16 bpp com uma PSNR de 38,61. Para o método JPEG2000, o limite de bits foi de 0,29 bpp com uma PSNR de 54,92.

Os resultados obtidos com a compressão das imagens mamográficas pelos métodos fractal e pelo método JPEG2000 foram bastante diferentes em termos de taxa de bits. A

compressão JPEG2000 conseguiu comprimir imagens com taxas de compressão maiores mantendo a qualidade da imagem reconstruída, enquanto que as taxas de compressão não foram tão altas e a qualidade da imagem reconstruída não foi tão boa para a compressão fractal.

Considerando os resultados obtidos dos testes realizados com as imagens mamográficas, observou-se que a compressão fractal apesar de não apresentar o mesmo desempenho da compressão JPEG2000, consegue comprimir as imagens mamográficas a taxas relativamente boas, mantendo a qualidade das imagens que foram consideradas aceitáveis pelos médicos especialistas.

A técnica JPEG2000 é bastante popular e muito utilizada. A técnica fractal vem sendo conhecida aos poucos e cada vez está sendo mais estudada, por ser um método bastante simples que utiliza as similaridades da própria imagem para realizar a compressão.

7.3 Contribuições deste Trabalho

Neste trabalho foi desenvolvido um algoritmo baseado nos estudos de *Fischer*. Os resultados obtidos pelos testes realizados com a finalidade de estudar os efeitos dos parâmetros de entrada da compressão fractal, pelos métodos da força bruta e da partição quadtree, foram condizentes com a bibliografia.

A partir desses resultados, este trabalho propôs uma maneira de reduzir a quantidade de comparações entre cada *range block* e as 8 transformações da cada *domain block*. Cada bloco é classificado em uma das 3 superclasses existentes de acordo com ordenação de média dos sub-blocos e os *range block* só são comparados com os *domain blocks* que estão na mesma superclasse. Para incluir o bloco em um das superclasses, o bloco deve ser submetido a uma das 8 transformações. Conhecendo essa transformação, ao invés do *range block* ser comparado com as 8 transformações do *domain block*, ele só é comparado com o *domain block* transformado que faz com que eles estejam na mesma superclasse.

Foram selecionadas algumas imagens mamográficas que foram comprimidas pelo método fractal utilizando o algoritmo desenvolvido neste trabalho. As imagens comprimidas foram analisadas por médicos especialistas que definiram quais imagens poderiam ser

consideradas aceitáveis. A partir dessa análise estabeleceu-se um limite de taxa de bits das imagens mamográficas comprimidas. Este estudo gerou um artigo na Revista Telecomunicações publicada pelo Inatel (Instituto Nacional de Telecomunicações) [29].

7.4 Propostas para Trabalhos Futuros

Devido a dificuldade de encontrar médicos especialistas dispostos a analisar as imagens comprimidas e a quantidade de imagens mamográficas utilizadas nesta dissertação ser pequena, como trabalho futuro pode-se utilizar um maior número de médicos especialistas e uma maior quantidade de imagens mamográficas para melhorar a consistência dos resultados obtidos nesta dissertação.

Outro trabalho que pode ser desenvolvido é segmentar as imagens mamográficas para selecionar as áreas consideradas anormais pelos especialistas. E após a segmentação aplicar métodos de compressão com uma menor taxa de compressão nessas áreas e nas outras áreas métodos de compressão com taxas mais altas.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing*. 2nd ed., Prentice Hall, Upper Saddle River, New Jersey, 2002.
- [2] RIBAS, J. P. I. F. *Um modelo de Compressão de Imagens Digitais Baseado em Codificação Fractal e Quantização Vetorial*. Universidade Federal de Uberlândia, Uberlândia, MG. 2008. Tese de Doutorado.
- [3] FISCHER, Y. *Fractal Image Compression: Theory and Application*. Ed. Springer-Verlag, New York, USA. 1995. 341 p.
- [4] SAUPE, D.; HAMZAOU, R.; HARTENSTEIN, H. *Fractal Image Compression An Introductory Overview*. Revised version of chapter 2 from “Fractal Models for Image Synthesis, Encoding and Analysis”, SIGGRAPH’96 Course Notes XX, New Orleans, Aug. 1996.
- [5] BARNESLEY, M. F. *Fractals Everywhere*. Academic Press, San Diego, USA. 1993. 394 p.
- [6] MANDELBROT, B. B. *The Fractal Geometry of Nature*. W. H. Freeman, Oxford, 1983.
- [7] LIMA, E. L. *Espaços Métricos*. Instituto de Matemática Pura e Aplicada, CNPq, Rio de Janeiro, 1977.
- [8] BARNESLEY, M. F. *Fractal Image Compression*. Notices of the AMS, vol. 43, no. 06, Jun. 1996.
- [9] CRUZ, A. L. M.; SILVA, F. S.; CHAVEZ, R. F. L.; IANO, Y. *Codificação Fractal de Imagens*. Revista Telecomunicações. vol. 11, no. 01, Mai. 2008.
- [10] MELKEBEEK, D. V.; BULTHEEL, A. *On the relation between Iterated Function Systems and Partitioned Iterated Function Systems*. Report TW240, Department of Computer Science, K. U. Leuven, Mar. 1996.

- [11] JUNIOR, R. R.; GUAZZELLI, M. A. *Avaliação Computacional da Compressão Fractal de Imagens Utilizando a Técnica Partitioned Iterated Function System*. 7th Brazilian Conference on Dynamics, Control and Applications. Presidente Prudente, SP, Brasil. Mai. 2008.

- [12] GHAZEL, M. *Adaptative Fractal and Wavelet Image Denoising*. Ph.D. thesis. Waterloo, Ontario, Canada, 2004.

- [13] WOHLBERG, B.; JAGER, G. *A Review of the Fractal Image Coding Literature*. IEEE Transactions on Image Processing, vol. 8, no.12, pp. 1716-1729, Dec. 1999.

- [14] JACQUIN, A. E. *Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations*. IEEE Transactions on Image Processing, vol. 01, no. 01, Jan. 1992.

- [15] JACQUIN, A. E. *Fractal Image Coding: A Review*. IEEE, vol. 81, no. 10, Out. 1993.

- [16] TONG, C. S.; PI, M. *Fast Fractal Image Encoding Based on Adaptative Search*. IEEE Transaction on Image Processing, vol. 10, no. 09, Set. 2001.

- [17] FISCHER, Y. *Fractal Image Compression*. San Diego, California. 1992. SIGGRAPH'92, Course Notes.

- [18] WU, Y. G.; HUANG, M. Z.; WEN, Y. L. *Fractal Image Compression with Variance and Mean*. International Conference on Multimedia and Expo - volume 1 (ICME '03), pp. 353-356, 2003.

- [19] SKODRAS, A. N.; CHRISTOPOULOS, C. A.; EBRAHIMI, T. *JPEG2000: The Upcoming Still Image Compression Standard*. Pattern Recognition Letters, vol. 22, pp. 1337-1345, Oct. 2001.

- [20] MISITI, M.; MISITI, Y; OPPENHEIM, G.; POGGI, J. M. *Wavelet Toolbox User's Guide 4*. The MathWorks, Inc. Natick, MA, USA. 2009.

- [21] CHRYSAFIS C.; ORTEGA, A. *Line-Based, Reduced Memory, Wavelet Image Compression*. IEEE Transaction on Image Processing, vol. 9, no. 3, Mar. 2000.

- [22] SILVA, A. L. M. C. S. S. *Procedimentos para Método Híbrido de Compressão de Imagens Digitais Utilizando Transformadas Wavelet e Codificação Fractal*. Universidade Estadual de Campinas - UNICAMP, Campinas, SP. Mai. 2005. Tese de Doutorado.
- [23] DELFINO, S. R.; NUNES, F. L. S. *Punção de Mama Virtual: Um Módulo para Geração de Casos para Estudo*. VII Workshop de Informática Médica - WIM2007, pp. 146-155. 2007.
- [24] SILVA, J. M.; FLORES, E. L. *Compressão de Imagens Médicas Usando Fractais*. VI Workshop de Informática Médica – WIM2006. 62 – 72 p. 2006.
- [25] BRASIL. *Falando sobre Mamografia. Viva Mulher – Programa Nacional de Controle do Câncer do Colo do Útero e de Mama*. Ministério da Saúde. Instituto Nacional de Câncer. Coordenação de Prevenção e Vigilância (Conprev). Rio de Janeiro: MS/INCA, 2000. 81 p. ISBN: 85-7318-063-3
- [26] BRASIL. *Mamografia: da prática ao controle*. Ministério da Saúde. Instituto Nacional de Câncer. Rio de Janeiro: INCA, 2007. 109 p.
- [27] University of South Florida. *Digital Database for Screening Mammography*. Disponível em: <http://marathon.csee.usf.edu/Mammography/DDSM/thumbnails/normals/normal_01/overview.html> Acesso em: 26 de jun, 2009.
- [28] ADAMS, M. *The JasPer Project Home Page*. Disponível em: <<http://www.ece.uvic.ca/~mdadams/jasper/>> Acesso em: 30 de jul, 2009.
- [29] VERGÜTZ, S.; VEIGA, A. C. P. *Associação entre os Critérios Objetivo e Subjetivo na Classificação de Imagens Mamográficas Sujeitas a Compressão Fractal*. Revista Telecomunicações, vol. 13, no. 01, Mai. 2011.

APÊNDICE

Como apêndice, tem-se o CD a seguir contendo as imagens das Figura 6.6 a Figura 6.37.