



COPPE/UFRJ

ALGORITMOS PARA SUPER-RESOLUÇÃO DE IMAGENS BASEADOS NAS
FILTRAGENS DE WIENER E ADAPTATIVA USANDO A TRANSFORMADA
WAVELET

Erica Sayuri Takemura

Dissertação de Mestrado apresentada ao
Programa de Pós-graduação em Engenharia
Elétrica, COPPE, da Universidade Federal
do Rio de Janeiro, como parte dos requisitos
necessários à obtenção do título de Mestre
em Engenharia Elétrica.

Orientadora: Mariane Rembold Petraglia

Rio de Janeiro
Outubro de 2010

ALGORITMOS PARA SUPER-RESOLUÇÃO DE IMAGENS BASEADOS NAS
FILTRAGENS DE WIENER E ADAPTATIVA USANDO A TRANSFORMADA
WAVELET

Erica Sayuri Takemura

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE
ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE
JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA
ELÉTRICA.

Examinada por:

Prof. Mariane Rembold Petraglia, Ph.D.

Prof. José Gabriel Rodriguez Carneiro Gomes, Ph.D.

Prof. Lisandro Lovisolo, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
OUTUBRO DE 2010

Takemura, Erica Sayuri

Algoritmos para Super-Resolução de Imagens Baseados nas Filtragens de Wiener e Adaptativa Usando a Transformada Wavelet/Erica Sayuri Takemura. – Rio de Janeiro: UFRJ/COPPE, 2010.

XIII, 69 p.: il.; 29, 7cm.

Orientadora: Mariane Rembold Petraglia

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2010.

Referências Bibliográficas: p. 67 – 69.

1. Super-resolução de imagens. 2. Transformada Wavelet. 3. Filtro de Wiener. 4. Filtragem Adaptativa.
I. Petraglia, Mariane Rembold. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica.
III. Título.

À minha família.

Agradecimentos

À minha família, que incondicionalmente, sempre me deu o apoio que precisava.

À minha orientadora Prof. Mariane Rembold Petraglia, que com sua experiência e amplo conhecimento na área, teve muita paciência e compreensão durante todo o projeto.

Em especial, agradeço ao Paulo, Odete, Felipe e Roberta do Nascimento, por me ajudarem no tempo que foi necessário para terminar a tese de mestrado.

À secretaria do PEE, em especial, à Daniele Cristina, meus sinceros agradecimentos, pela sua colaboração em todos os momentos do curso de mestrado.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

ALGORITMOS PARA SUPER-RESOLUÇÃO DE IMAGENS BASEADOS NAS
FILTRAGENS DE WIENER E ADAPTATIVA USANDO A TRANSFORMADA
WAVELET

Erica Sayuri Takemura

Outubro/2010

Orientadora: Mariane Rembold Petraglia

Programa: Engenharia Elétrica

A super-resolução sempre teve uma atenção substancial na área de processamento de imagens, proporcionando maior resolução a partir de imagens de baixa resolução. Interessantes técnicas têm sido desenvolvidas e resultados práticos têm sido obtidos. No entanto, em várias pesquisas, bons resultados são frequentemente demonstrados em simulações sintetizadas, que limitam o uso da técnica na prática.

Esta tese apresentará um estudo de algoritmos de super-resolução, baseados em filtragem de Wiener e filtragem adaptativa, que são possuem baixa complexidade computacional em relação aos outros métodos de otimização. A seguir, aplicaremos estas técnicas à sequência de imagens de baixa resolução decompostas por coeficientes da wavelet de Haar. Resultados experimentais observados corroboram os resultados previstos analiticamente.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

SUPER-RESOLUTION ALGORITHMS BASED ON WIENER AND ADAPTIVE
FILTERING USING WAVELET TRANSFORM

Erica Sayuri Takemura

October/2010

Advisor: Mariane Rembold Petraglia

Department: Electrical Engineering

The super-resolution has always attracted a substantial attention in the field of image processing providing higher resolution from low resolution sequence image. Interesting techniques have been developed and practical results have been obtained. However, in several studies, good results are often summarily demonstrated in simulations that limit the use of the techniques in practice.

This thesis will present a study on super-resolution algorithms, based on Wiener filtering and adaptive filtering that have low computational complexity compared to other optimization methods. Then, such techniques will be applied to sequences of low-resolution images decomposed by Haar wavelet coefficients. The experimental results corroborate the results of analytical predictions.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xiii
1 Introdução	1
1.1 Revisão Bibliográfica	1
1.2 Organização da tese	4
2 Métodos de Super-Resolução por Filtragem de Wiener e Adaptativa	5
2.1 Filtragem de Wiener	5
2.1.1 Modelo Estatístico Global	9
2.1.2 Modelo estatístico de variáveis espaciais	11
2.2 Filtragem adaptativa	12
2.2.1 "Falso" algoritmo RLS	15
2.2.2 O algoritmo LMS	15
3 Métodos de Super-Resolução utilizando a Transformada Wavelet	17
3.1 Transformada Wavelet	19
3.2 Método de Wiener com a transformada wavelet	21
3.3 Algoritmo adaptativo com a transformada wavelet	22
3.4 Redução de ruído (<i>denoising</i>) com wavelets	23
4 Resultados	25
4.1 Descrição dos Experimentos	26
4.1.1 Filtragem de Wiener	26
4.1.2 Filtragem Adaptativa	27
4.2 Resultados dos Experimentos	28
5 Conclusões e Trabalhos Futuros	65
Referências Bibliográficas	67

Lista de Figuras

1.1	Método da Interpolação Não-Uniforme	2
2.1	Modelo de observação simulando movimento através de deslocamentos.	6
2.2	Modelo de observação alternativo.	7
2.3	Proposta do algoritmo usando o método Wiener.	7
2.4	Matriz de HR ilustrando pixels de LR.	8
2.5	Função autocorrelação com $\sigma^2 = 1$ e $\rho = 0.75$	10
3.1	Deslocamentos das imagens para os experimentos.	17
3.2	Geração da imagens de baixa resolução	18
3.3	Métodos de super-resolução usando a transformada wavelet	18
3.4	Wavelets de Haar: (a) Função escala $\phi_{0,0}$; (b) Wavelet mãe $\psi_{0,0}$; (c) Função escala $\phi_{1,2}$; (d) Wavelet mãe $\psi_{1,2}$	19
3.5	(a) Transformada wavelet bidimensional padrão; (b) Transformada aplicada na sub-banda LL.	21
3.6	Definição de limiares nos detalhes horizontal, diagonal e vertical.	23
4.1	Função $H(x, y) = (1/9)rect(x/3, y/3)$	27
4.2	Imagem de satélite: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.	29
4.3	Imagem de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	30
4.4	Imagem de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	31
4.5	Imagens de satélite: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem de adaptativa e Wavelet.	32

4.6	Imagens de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	33
4.7	Imagens de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	34
4.8	Imagens Super-resolução de satélite sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.	35
4.9	Imagens Super-resolução de satélite com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	36
4.10	Imagens Super-resolução de satélite com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	37
4.11	Imagens de texto: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.	41
4.12	Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	42
4.13	Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	43
4.14	Imagens de texto: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet.	44
4.15	Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	45

4.16	Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	46
4.17	Imagens Super-resolução de texto sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.	47
4.18	Imagens Super-resolução de texto com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	48
4.19	Imagens Super-resolução de texto com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	49
4.20	Imagens de face: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.	53
4.21	Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	54
4.22	Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e <i>denoising</i>	55
4.23	Imagens de face: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet.	56
4.24	Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	57
4.25	Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e <i>denoising</i>	58

4.26	Imagens Super-resolução de face sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.	59
4.27	Imagens Super-resolução de face com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	60
4.28	Imagens Super-resolução de face com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e <i>denoising</i> ; (f) Filtragem adaptativa, Wavelet e <i>denoising</i>	61

Lista de Tabelas

4.1	Erro médio quadrático das imagens de super-resolução de satélite . . .	38
4.2	Grau de Similaridade das imagens de super-resolução de satélite . . .	38
4.3	Erro médio quadrático das imagens de super-resolução de texto . . .	50
4.4	Grau de Similaridade das imagens de super-resolução de texto . . .	50
4.5	Erro médio quadrático das imagens de super-resolução de face . . .	62
4.6	Grau de Similaridade das imagens de super-resolução de face . . .	62

Capítulo 1

Introdução

Super-resolução é uma técnica que utiliza múltiplas imagens de baixa resolução, denominadas de *frames*, de uma mesma cena para se obter uma imagem em uma resolução mais alta, através de deslocamentos dos *frames* e de filtragens para redução de ruído.

A qualidade de uma imagem é afetada pela distância, posição do objeto e da câmera, condições ambientais e resolução do sensor de imagens. Na maioria das aplicações de Processamento de Imagens é desejável que o sistema de aquisição apresente a maior resolução possível e introduza uma quantidade mínima de distorções devido às imperfeições óticas e aos ruídos inerentes ao sistema. Entretanto, a densidade dos sensores empregados em sistemas de aquisição, tais como câmeras, filmadoras e escaneadores digitais é limitada e, assim como a qualidade do sistema ótico implementado, está diretamente associada ao custo do sistema. Uma alternativa para aumentar a resolução e a qualidade das imagens adquiridas, mantendo o custo do sistema de aquisição, é a utilização de técnicas de processamento digital das imagens adquiridas visando ao aumento de resolução.

Técnicas de super-resolução a partir de sequências de imagens de baixa resolução são as mais eficientes e muito úteis e importantes em várias aplicações, como na astronomia, imagens de satélite, análise de imagens médicas e para recursos militares.

Esta tese propõe algoritmos que aumentam a resolução das imagens a partir de imagens deslocadas de baixa resolução, baseados em filtros de Wiener e filtragem adaptativa no domínio da transformada wavelet.

1.1 Revisão Bibliográfica

As técnicas de super-resolução (SR) combinam uma ou mais imagens de baixa resolução (LR - *Low Resolution*) de uma mesma cena, a fim de produzir uma ou várias imagens de alta resolução (HR - *High Resolution*). As imagens LR

representam a mesma área, mas possuem diferenças entre elas, que podem ser diferentes datas de aquisição, diferentes projeções, deslocamentos de pixel ou subpixel. Na terminologia de super-resolução, o termo quadro (ou *frame*) é utilizado como referência às imagens LR.

Papoulis [1] propôs um algoritmo que projeta todos os pixels alinhados na mesma grade, realiza uma filtragem no domínio da frequência, que pode ser do tipo passa-faixa ou passa-baixa, e avalia o resultado para determinar uma possível convergência.

Tsai e Huang [2] realizaram um estudo baseado na translação de movimentos usando transformada rápida de cosseno e transformada discreta de Fourier. Esse método resolvia o problema do registro e da restauração, mas não considerava a degradação do sinal e o ruído de medição. A seguir, serão descritos os métodos de super-resolução mais utilizados: interpolação não-uniforme, métodos que utilizam a transformada wavelet e filtragem adaptativa. As técnicas de interpolação [3] (sendo as mais empregadas o vizinho mais próximo, a bilinear e a bicúbica) diferem das de SR porque nas primeiras somente uma imagem de LR é utilizada para gerar uma imagem de resolução maior.

O método de interpolação não-uniforme [4] é o mais intuitivo para a obtenção de imagens de SR. As imagens LR são mapeadas em um plano comum e então interpoladas para a grade da imagem HR, como apresentado na Figura 1.1.

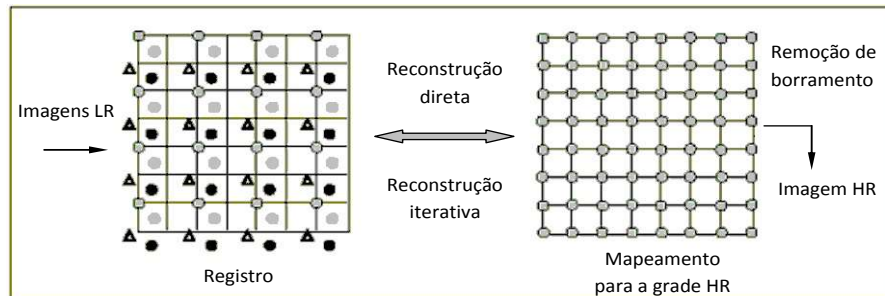


Figura 1.1: Método da Interpolação Não-Uniforme

A interpolação não-uniforme tem uma carga computacional relativamente baixa que torna possível sua aplicação em tempo real. Contudo, nesses métodos, os modelos de degradação são limitados para casos em que as características de ruído e da degradação são iguais para todas as imagens LR. Estes métodos tampouco garantem uma reconstrução total, pois a etapa de reconstrução desconhece os erros que ocorrem na etapa de interpolação.

Os métodos baseados na interpolação não-uniforme apresentam resultados um pouco melhores que a interpolação bilinear, pois na primeira, geralmente é realizada uma remoção de borrimento [5] e a interpolação bilinear é obtida através de médias dos pixels vizinhos mais próximos. Esse processo gera uma suavização nas bordas ou perda de nitidez.

Ur e Gross [6] desenvolveram um método de interpolação não-uniforme que utiliza um conjunto de imagens com deslocamentos espaciais entre elas. Eles utilizam o teorema da amostragem em multicanais generalizado de Papoulis [1] e Brown [7]. A interpolação é seguida de um processo de remoção do efeito "borrado", e os deslocamentos entre as imagens LR são assumidos como conhecidos.

O conceito de análise em multiresolução foi utilizado primeiro por Mallat [8]. Os diferentes usos da transformada wavelet (TW) no processamento de imagens se devem à sua capacidade de representação e reconstrução de funções em multiresolução. A TW é capaz de representar uma imagem em diferentes resoluções [9], permitindo tratar partes decompostas da imagem de modos diferentes.

Nguyen e Milanfar [10] desenvolveram um método híbrido que utiliza wavelet e interpolação não-uniforme, por eles denominado *wavelet super-resolution*, cuja aplicação é destinada à interpolação e restauração de imagens.

Ahuja e Bose [11] apresentaram o método inicialmente proposto por Papoulis e Brown [1], que foi melhorado por Ur e Gross [6], através de deslocamentos espaciais; no entanto, o algoritmo foi desenvolvido para ser aplicado nas imagens decompostas pela transformada wavelet.

A filtragem adaptativa foi aplicada à restauração de imagens por Elad e Feuer [12], e possui uma estrutura alterável ou ajustável, de tal forma que seu comportamento ou desempenho melhore, de acordo com algum critério desejado. O algoritmo empregado é realizado pela minimização do erro médio quadrático (LMS) ou pelo método recursivo dos mínimos quadrados (RLS). Este conceito oferece uma estimação progressiva de uma sequência de imagens em uma resolução mais alta.

O método baseado na filtragem de Wiener, proposto por Hardie [13], produz uma melhora na resolução de uma imagem a partir de sequência de imagens de baixa resolução (LR) usando estatísticas de segunda ordem dos sinais. O algoritmo estima o deslocamento das imagens para posicionar cada valor de pixel LR numa matriz comum, que será referência para os frames. Os valores dos pixels da imagem HR são obtidos através da soma ponderada dos pixels LR. Usando um modelo estatístico, os pesos para cada pixel HR são projetados para minimizar o erro médio quadrático, e dependem da posição relativa entre as imagens deslocadas de baixa resolução e a correlação entre as imagens de baixa resolução e a imagem estimada de alta resolução. Desta maneira, os pesos são adaptados para se adequarem às mudanças de distribuição dos pixels LR.

Os dois últimos métodos terão seus conceitos detalhados no Capítulo 2.

1.2 Organização da tese

No Capítulo 2 são apresentados os algoritmos baseados em filtragem de Wiener e filtragem adaptativa, onde são descritos os principais conceitos dos mesmos. No Capítulo 3 apresentamos os métodos de super-resolução propostos que utilizam a transformada wavelet, sendo mostradas as alterações feitas nos algoritmos originais. No Capítulo 4 são descritos os procedimentos para a realização das implementações e resultados dos experimentos com os algoritmos apresentados nos Capítulos 2 e 3. No Capítulo 5 são apresentadas as conclusões e direções para pesquisas futuras.

Capítulo 2

Métodos de Super-Resolução por Filtragem de Wiener e Adaptativa

Neste capítulo descreveremos os dois algoritmos de super-resolução estudados neste trabalho, os quais apresentam baixa complexidade comparados aos métodos de otimização. O método baseado em filtragem de Wiener foi proposto por Hardie [13], enquanto que o método que utiliza a filtragem adaptativa foi apresentado por Elad e Feuer [12]. As próximas seções apresentam os detalhes destes algoritmos obtidos de [13] e [12].

2.1 Filtragem de Wiener

O método de super-resolução baseado em filtragem de Wiener combina interpolação não-uniforme e restauração através de simples operações de somas ponderadas dos pixels de imagens deslocadas de baixa resolução, sendo a determinação dos coeficientes de ponderação detalhada neste capítulo.

O diagrama em blocos ilustrado na figura 2.1 é um modelo de observação que mostra como a imagem de alta resolução ideal, definida como \mathbf{D} , é obtida a partir de imagens de baixa resolução deslocadas, $\mathbf{f}(k)$, através desse processo inverso. A princípio, a imagem HR passa por uma transformação geométrica para reproduzir deslocamentos que ocorrem durante a aquisição de N imagens, ou seja:

$$\mathbf{D}_k = T_{\theta_k} \mathbf{D} \quad (2.1)$$

para $k = 1, 2, \dots, N$, onde θ_k são parâmetros de deslocamentos associados ao k -ésimo quadro e \mathbf{D} é uma matriz retangular que representa a imagem HR. Embora seja necessária exatidão na grandeza de subpixel, esta resolução é difícil de se obter quando uma grande quantidade de parâmetros de deslocamento são estimados de um número limitado de pixels. Apesar do foco principal ser o deslocamento linear,

o algoritmo descrito nesta seção é também adequado para movimentos rotatórios, pois considera que a função de espalhamento pontual (*Point Spread Function*-PSF) é simétrica e circular (como será descrito abaixo), podendo ser estendido para outros tipos de movimentos.

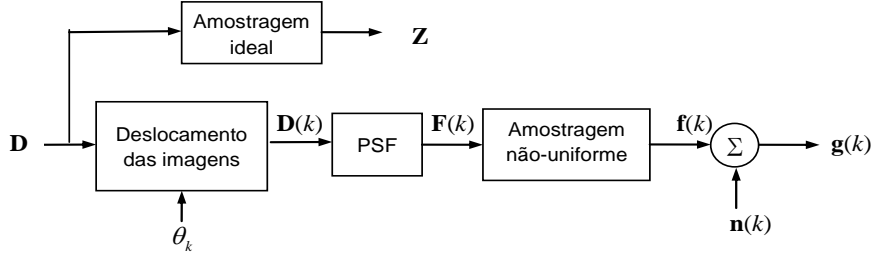


Figura 2.1: Modelo de observação simulando movimento através de deslocamentos.

Depois da transformação geométrica, a imagem é convoluída com a PSF, um operador linear que representa foco incorreto e contaminação da energia de um pixel por seus vizinhos. A convolução é descrita por:

$$\mathbf{F}_k = \mathbf{D}_k * \mathbf{H} \quad (2.2)$$

onde \mathbf{H} representa a PSF do sistema. O próximo passo no modelo de observação é subamostrar a imagem. Finalmente é usado um modelo de ruído aditivo para simular fontes de ruído fotométrico e eletrônico. Com isso, forma-se um vetor com os valores de pixels para cada frame k , denominados de $\mathbf{g}(k)$. Todos os valores dos pixels das imagens LR podem ser combinados num único vetor dado por $\mathbf{g} = [\mathbf{g}(1)^T, \mathbf{g}(2)^T, \dots, \mathbf{g}(N)^T]^T$.

Embora a figura 2.1 seja uma boa representação de um processo de aquisição física de imagens, um modelo mais conveniente é apresentado na figura 2.2. A transformação geométrica que representa os movimentos é removida e substituída por um processo de amostragem não-uniforme guiado por parâmetros de deslocamentos.

Os modelos das figuras 2.1 e 2.2 são equivalentes para movimentos lineares, se a função PSF for circularmente simétrica. Uma vez adquirida a função PSF e os modelos de deslocamentos estimados, os movimentos e as operações de amostragem da figura 2.1 podem ser combinados em uma simples operação de amostragem não-uniforme, como mostrado na figura 2.2. A proposta do algoritmo é usar $\mathbf{g}(k)$, onde $k = 1, 2, \dots, N$, para formar uma estimativa de uma imagem de alta resolução \mathbf{Z} , denominada por $\hat{\mathbf{Z}}$. Uma visão geral da proposta do algoritmo usando o método de Wiener é dada pelo diagrama em blocos da figura 2.3.

A função r_{ff} contém os dados estatísticos das imagens deslocadas de baixa

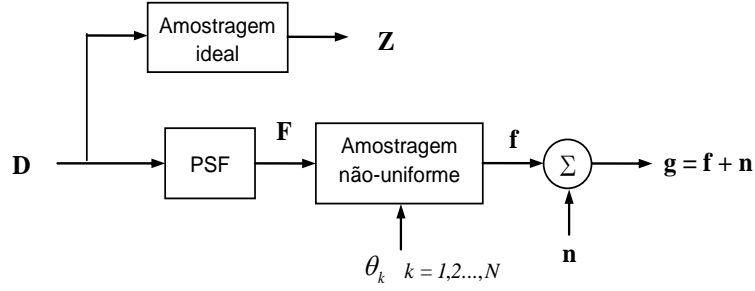


Figura 2.2: Modelo de observação alternativo.

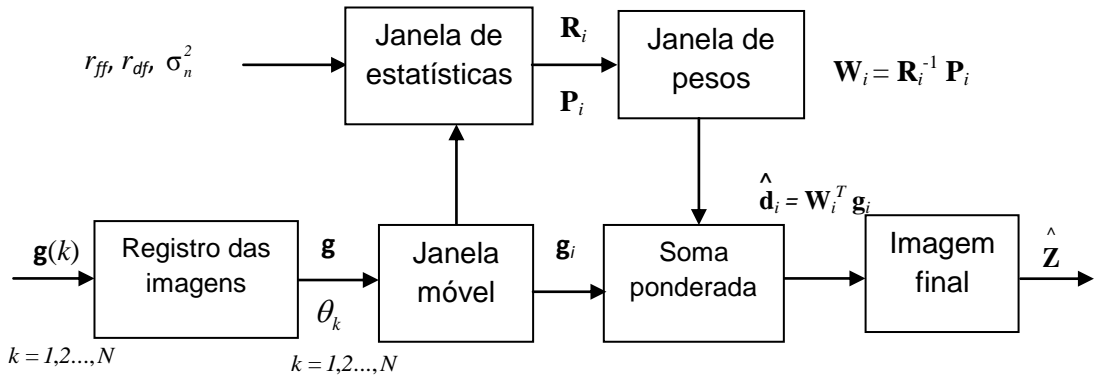


Figura 2.3: Proposta do algoritmo usando o método Wiener.

resolução, r_{df} é a função correlação cruzada entre as imagens de baixa resolução e a imagem desejada, e σ_n^2 é a variância do vetor do ruído de medição decorrelatado de média zero das imagens de baixa resolução.

O primeiro passo do algoritmo é ordenar os *frames* em uma matriz comum. A matriz de HR é alinhada na posição da média do *frame* (o que dá estabilidade no movimento para processamento de vídeo) sem adição de complexidade computacional, ou seja, os pixels de alta resolução a serem estimados ficam dispostos no centro da janela de observação, como mostra a figura 2.4. Os resultados dos parâmetros de deslocamentos são guardados em θ_k , para $k = 1, 2, \dots, N$. Com esses parâmetros é possível estabelecer as coordenadas de cada pixel das imagens de baixa resolução em uma grade comum. O próximo passo é agrupar pixels de baixa resolução através de uma janela de observação. A janela cobre w_x pixels da imagem HR na direção horizontal e w_y na direção vertical. Todos os pixels da imagem LR que estão na matriz de HR estão situados no vetor de observação $\mathbf{g}_i = [\mathbf{g}_{i,1}^T, \mathbf{g}_{i,2}^T, \dots, \mathbf{g}_{i,N}^T]$, onde i é o índice da posição da janela da matriz de HR, N é o número de imagens de baixa resolução utilizadas para formar a imagem de alta resolução e $\mathbf{g}_{i,k}$ é o vetor da k -ésima imagem LR da i -ésima janela de observação.

Considerando que a janela de observação é limitada por w_x e w_y e que são

inteiros múltiplos dos fatores de interpolação nas direções horizontal e vertical, l_x e l_y , respectivamente, o número de elementos que compõe o vetor \mathbf{g}_i é dado por:

$$K_i = \frac{Nw_xw_y}{l_xl_y} \quad (2.3)$$

onde N é o número de imagens de baixa resolução.

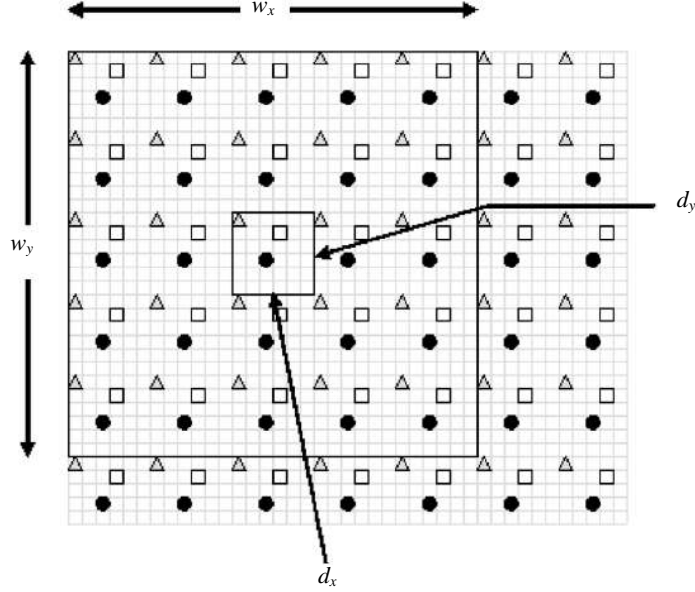


Figura 2.4: Matriz de HR ilustrando pixels de LR.

Para cada janela de observação é formada uma estimativa dos pixels de HR contidos na subjanela central de observação, como mostrado na figura 2.4. Nesse exemplo, as dimensões das subjanelas são denominadas d_x e d_y , onde $1 < d_x < w_x$ e $1 < d_y < w_y$. Note que a subjanela representada na figura 2.4 tem dimensões $d_x = d_y = 6$ e que $l_x = l_y = 6$. Nas implementações dos métodos de super-resolução, o registro das imagens assume que os deslocamentos dos *frames* são conhecidos. A estimativa dos pixels HR da subjanela é obtida usando a soma ponderada dos pixels LR da janela de observação e é expressa por:

$$\hat{\mathbf{d}}_i = \mathbf{W}_i^T \mathbf{g}_i \quad (2.4)$$

onde $\hat{\mathbf{d}}_i = [\hat{d}_{i,1}, \hat{d}_{i,2}, \dots, \hat{d}_{i,d_x d_y}]^T$ é o vetor com as estimativas dos pixels da i -ésima sub-janela da imagem desejada e \mathbf{W}_i é uma matriz de pesos de dimensão $K_i \times d_x d_y$. Cada coluna de \mathbf{W}_i contém os pesos usados para um pixel HR particular da janela de estimação $\hat{d}_{i,j}$. Portanto, $\mathbf{w}_{i,j}$ é vetor de pesos para estimar o j -ésimo pixel da imagem desejada da i -ésima janela de observação. A janela de observação se move nas direções horizontal e vertical de d_x e d_y pixels, respectivamente. Os pesos

da matriz que minimizam o erro médio quadrático entre a imagem estimada e as imagens deslocadas de baixa resolução são dados por

$$\mathbf{W}_i = \mathbf{R}_i^{-1} \mathbf{P}_i \quad (2.5)$$

onde $\mathbf{R}_i = E[\mathbf{g}_i \mathbf{g}_i^T]$ é a matriz de autocorrelação usada para o vetor de observação e $\mathbf{P}_i = E[\mathbf{g}_i \mathbf{d}_i^T]$ é a matriz de correlação cruzada entre o vetor desejado $\mathbf{d}_i = [d_{i,1}, d_{i,2}, \dots, d_{i,d_x d_y}]^T$ e o vetor de observação \mathbf{g}_i . O vetor \mathbf{d}_i é o vetor das amostras da imagem desejada correspondente às posições da janela de estimação. É usado o modelo contínuo paramétrico de função autocorrelação e correlação cruzada para a imagem desejada, ou seja, essas funções contínuas são amostradas em coordenadas determinadas pelos pixels de LR situados na matriz de HR para que as matrizes \mathbf{R}_i e \mathbf{P}_i sejam calculadas, eliminando a necessidade de treinamento de imagens e atribuindo ao sistema um número pequeno de parâmetros.

2.1.1 Modelo Estatístico Global

Para encontrar as matrizes \mathbf{R}_i e \mathbf{P}_i , considera-se a princípio que \mathbf{f}_i é uma imagem livre de ruído aleatório correspondente ao i -ésimo vetor de observação \mathbf{g}_i . O sinal observado pode então ser escrito como:

$$\mathbf{g}_i = \mathbf{f}_i + \mathbf{n}_i \quad (2.6)$$

onde \mathbf{n}_i é um vetor de ruído aleatório associado às amostras da i -ésima janela de observação. Considera-se também que o vetor ruído é branco, com média zero e variância σ_n^2 , e descorrelatado com \mathbf{f}_i . Nesse caso, é simples mostrar que a matriz de autocorrelação para o vetor de observação é dada por

$$\mathbf{R}_i = E[\mathbf{g}_i \mathbf{g}_i^T] = E[\mathbf{f}_i \mathbf{f}_i^T] + \sigma_n^2 \mathbf{I} \quad (2.7)$$

e a matriz de correlação cruzada é dada por

$$\mathbf{P}_i = E[\mathbf{g}_i \mathbf{d}_i^T] = E[\mathbf{f}_i \mathbf{d}_i^T] \quad (2.8)$$

Para determinar $E[\mathbf{f}_i \mathbf{f}_i^T]$ e $E[\mathbf{f}_i \mathbf{d}_i^T]$, considera-se uma função de autocorrelação $r_{dd}(x, y)$ de um processo estacionário no sentido amplo para a imagem desejada $d(x, y)$. A função de correlação cruzada entre $d(x, y)$ e $f(x, y)$ pode ser expressa em termos de $r_{dd}(x, y)$ na forma

$$r_{df}(x, y) = r_{dd}(x, y) * h(x, y) \quad (2.9)$$

onde $h(x, y)$ é uma função retangular que representa a PSF. A função de autocorrelação de $f(x, y)$ é dada por

$$r_{ff}(x, y) = r_{dd}(x, y) * h(x, y) * h(-x, -y) \quad (2.10)$$

Com os dados das posições dos pixels de LR, as distâncias horizontal e vertical entre as amostras podem ser facilmente encontradas.

Calculando a expressão definida na equação (2.10) e usando todas as informações de deslocamentos, obtém-se $E[\mathbf{f}_i \mathbf{f}_i^T]$, e conseqüentemente a matriz \mathbf{R}_i da equação (2.7) é determinada. O mesmo ocorre para determinar \mathbf{P}_i pela equação (2.8), a qual será obtida através da equação (2.9). A posição relativa dos pixels de baixa resolução em um grid de alta resolução é o que determina as matrizes de autocorrelação e correlação cruzada, e por isso os pesos são ajustados para cada modelo de deslocamentos.

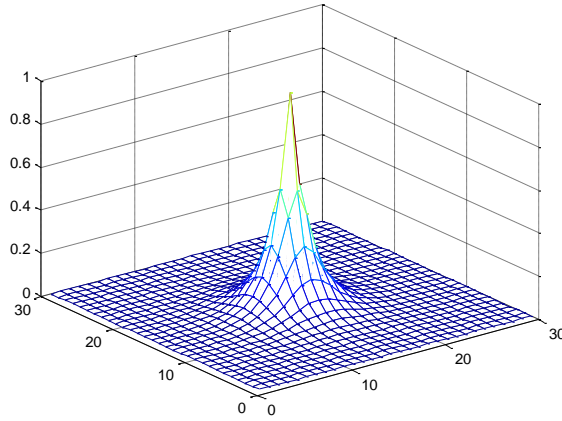


Figura 2.5: Função autocorrelação com $\sigma^2 = 1$ e $\rho = 0.75$

O modelo da função autocorrelação (figura 2.5) utilizado neste trabalho possui parâmetro circularmente simétrico, ou seja,

$$r_{dd}(x, y) = \sigma_d^2 \rho \sqrt{x^2 + y^2} \quad (2.11)$$

onde σ_d^2 é a variância da imagem desejada e ρ é um parâmetro de ajuste que controla o declínio da função autocorrelação com a distância do centro. A figura 2.5 ilustra a função autocorrelação para $\sigma_d^2 = 1$ e $\rho = 0.75$. Com esses parâmetros, é formada a função $r_{dd}(x, y)$ de (2.11), para se obter $r_{df}(x, y)$ dado por (2.9) e $r_{ff}(x, y)$ de (2.10), que serão usados para formar \mathbf{R}_i e \mathbf{P}_i . Usando essas matrizes em (2.5), geram-se os pesos para cada vetor de observação. Finalmente as estimativas dos pixels HR são formadas usando a expressão em (2.4). Este processo é repetido para cada posição da janela de observação.

2.1.2 Modelo estatístico de variáveis espaciais

O método anteriormente descrito supõe que os processos aleatórios envolvidos são estacionários no sentido amplo. Como as imagens normalmente não obedecem a essa suposição, pode ser mais vantajoso empregar um modelo de autocorrelação que é função da posição da janela de observação. Deste modo, além de adaptar-se a distribuição espacial local dos pixels LR em cada janela de observação, os pesos também podem se adaptar às intensidades das estatísticas locais dessas amostras. Portanto, o parâmetro da variância muda de acordo com a posição da janela de observação, e o novo modelo da função autocorrelação é expresso por:

$$r_{dd_i}(x, y) = \sigma_{d_i}^2 \rho \sqrt{x^2 + y^2} \quad (2.12)$$

onde $\sigma_{d_i}^2$ é a variância da i -ésima região da imagem desejada e que é responsável por produzir \mathbf{g}_i . Portanto, devemos estimar o parâmetro $\sigma_{d_i}^2$ para cada vetor de observação.

Observando a expressão em (2.10), a relação entre a variância da imagem desejada, $\sigma_{d_i}^2$, e a variância da imagem degradada por PSF correspondente, $\sigma_{f_i}^2$, é dada por:

$$\sigma_{d_i}^2 = \frac{1}{C(\rho)} \sigma_{f_i}^2 \quad (2.13)$$

onde

$$C(\rho) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \rho \sqrt{x^2 + y^2} \tilde{h}(x, y) dx dy \quad (2.14)$$

e $\tilde{h}(x, y) = h(x, y) * h(-x, -y)$. Como o ruído é assumido independente, $\sigma_{f_i}^2$ é estimado por:

$$\hat{\sigma}_{f_i}^2 = \hat{\sigma}_{g_i}^2 - \sigma_n^2 \quad (2.15)$$

onde $\hat{\sigma}_{g_i}^2$ é a variância estimada dos elementos de observação do vetor \mathbf{g}_i .

Portanto, as matrizes \mathbf{R}_i e \mathbf{P}_i são definidas, para qualquer $\sigma_{d_i}^2$, como:

$$\mathbf{R}_i = E[\mathbf{g}_i \mathbf{g}_i^T] = \sigma_{d_i}^2 E[\mathbf{f}_i \mathbf{f}_i^T] + \sigma_n^2 \mathbf{I} \quad (2.16)$$

$$\mathbf{P}_i = E[\mathbf{g}_i \mathbf{d}_i^T] = \sigma_{d_i}^2 E[\mathbf{f}_i \mathbf{d}_i^T] \quad (2.17)$$

e o filtro de Wiener é dado por:

$$\mathbf{W}_i = \left[E[\mathbf{f}_i \mathbf{f}_i^T] + \frac{\sigma_n^2}{\sigma_{d_i}^2} \mathbf{I} \right]^{-1} E[\mathbf{f}_i \mathbf{d}_i^T] \quad (2.18)$$

No Capítulo 4 serão mostrados os resultados da aplicação deste filtro para super-resolução de algumas imagens com diferentes razões sinal-ruído.

2.2 Filtragem adaptativa

O método de super-resolução por filtragem adaptativa restaura a imagem de alta-resolução a partir de uma sequência de imagens de baixa resolução, supondo que as imagens LR estejam em diferentes posições (que podem ser resultantes do movimento de uma câmera) ou então simplesmente deslocadas de subpixels. Num problema clássico de super-resolução, a imagem desejada e as imagens de baixa resolução, representadas em vetores, são relacionadas através da seguinte expressão:

$$\mathbf{g}_k = \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k \mathbf{d} + \mathbf{e}_k \quad (2.19)$$

para $1 \leq k \leq N$, onde N é o número de imagens de baixa resolução \mathbf{g}_k , cada uma com M^2 pixels, \mathbf{d} é o vetor com os L^2 pixels da imagem desejada (de alta resolução), \mathbf{C}_k é o operador de decimação de dimensão $M^2 \times L^2$, \mathbf{B}_k é uma matriz que introduz algum tipo de borramento, \mathbf{U}_k é uma matriz que representa a inserção de distorção geométrica, e \mathbf{e}_k representa ruído aleatório, diferente para cada imagem. Assume-se que as matrizes \mathbf{C}_k , \mathbf{B}_k e \mathbf{U}_k são conhecidas (ou estimadas) a priori.

A minimização do erro quadrático ponderado é comumente usada em processamento de sinais, segundo Becker [14] e Pearlman [15], por demonstrar bons resultados, e pode ser obtida através da escolha da função de ponderação \mathbf{W} de forma a minimizar o erro quadrático, segundo Rittner [16].

O erro citado é calculado através da seguinte equação:

$$\begin{aligned} \epsilon^2 &= \|\mathbf{S}\hat{\mathbf{d}}\|_{\mathbf{V}}^2 + \sum_{k=1}^N \|\mathbf{g}_k - \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k \hat{\mathbf{d}}\|_{\mathbf{W}_k}^2 \\ &= \hat{\mathbf{d}}^T \mathbf{S}^T \mathbf{V} \mathbf{S} \hat{\mathbf{d}} + \sum_{k=1}^N (\mathbf{g}_k - \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k \hat{\mathbf{d}})^T \mathbf{W}_k (\mathbf{g}_k - \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k \hat{\mathbf{d}}) \end{aligned} \quad (2.20)$$

O primeiro termo $\|\mathbf{S}\hat{\mathbf{d}}\|_{\mathbf{V}}^2$ é um mecanismo de regularização, que penaliza algebricamente a não suavidade da estimativa obtida da imagem desejada. A matriz \mathbf{S} é tipicamente um operador Laplaciano, e a matriz \mathbf{V} reflete o intuito de aplicar o filtro adaptável localmente e de não suavizar as bordas.

A matriz de pesos \mathbf{W}_k , de dimensão $M^2 \times M^2$, pode ser usada para dar pesos diferentes às imagens e pixels diferentes, dependendo da confiança dos seus valores. Esta confiança é ditada pela matriz de autocorrelação do ruído aditivo, ou seja, ruído de baixa variância se traduz em maior confiança.

Derivando ϵ^2 em relação a $\hat{\mathbf{d}}$ resulta na minimização do erro ponderado quadrático:

$$\frac{\partial \epsilon^2}{\partial \hat{\mathbf{d}}} = \mathbf{0} = -[\mathbf{S}^T \mathbf{V} \mathbf{S} + \sum_{k=1}^N \mathbf{U}_k^T \mathbf{B}_k^T \mathbf{C}_k^T \mathbf{W}_k \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k] \hat{\mathbf{d}} + \sum_{k=1}^N \mathbf{U}_k^T \mathbf{B}_k^T \mathbf{C}_k^T \mathbf{W}_k \mathbf{g}_k \quad (2.21)$$

O algoritmo da filtragem adaptativa propõe uma modificação na notação, considerando que as imagens resultam da amostragem em diferentes instantes de tempo.

Considerando que $\mathbf{d}(t)$ seja uma sequência de imagens de alta resolução ideal e $\mathbf{g}(t)$ seja de baixa resolução, esta é gerada a partir de $\mathbf{d}(t)$ através de:

$$\mathbf{g}(t) = \mathbf{CB}(t)\mathbf{d}(t) + \mathbf{e}_1(t) \quad (2.22)$$

onde \mathbf{C} é a matriz de decimação, de dimensão $M^2 \times L^2$, constante para todas as imagens obtidas, e $\mathbf{B}(t)$ é uma matriz de dimensão $L^2 \times L^2$, que representa o borrimento da t -ésima imagem de alta resolução, gerando a t -ésima imagem de baixa resolução correspondente. O efeito da degradação por decimação resulta em alteração da dimensão das matrizes: L^2 pixels para imagens de alta resolução e M^2 para imagens de baixa resolução, onde $L > M$. O vetor $\mathbf{e}_1(t)$ é o erro de medição do modelo no instante t , que pode ser considerado um ruído aleatório.

A expressão seguinte representa a relação no tempo da sequência de imagens ideais:

$$\mathbf{d}(t - k) = \mathbf{U}(t, k)\mathbf{d}(t) + \mathbf{e}_2(t, k) \quad (2.23)$$

onde $\mathbf{U}(t, k)$ é uma matriz $L^2 \times L^2$ de deslocamento geométrico aplicado em $\mathbf{d}(t)$ resultando na k -ésima imagem deslocada $\mathbf{d}(t-k)$ e $\mathbf{e}_2(t, k)$ é o erro entre a imagem medida em $t - 1$ e a imagem medida em t , sendo considerado um ruído aleatório.

Combinando as duas equações anteriores, podemos reescrever a equação (2.22), para as N imagens, como:

$$\mathbf{g}(t - k) = \mathbf{CB}(t - k)\mathbf{U}(t, k)\mathbf{d}(t) + \mathbf{e}(t, k) \quad (2.24)$$

onde $\mathbf{e}(t, k)$ é o erro total entre a imagem adquirida em $t - k$ e em t , assumindo ser aleatório, com matriz autocorrelação definida por $\mathbf{W}(t, k)^{-1}$. A motivação para utilizar esse modelo vem da tentativa de resolver o problema da restauração de uma sequência de imagens. Nesse caso, as N imagens medidas $\mathbf{g}(t-k)$, onde $0 \leq k \leq N - 1$, são geradas a partir de uma única imagem $\mathbf{d}(t)$, em uma similar relação vista anteriormente na equação (2.19). Alterando a notação do segundo termo da equação (2.20) para o atual modelo de aquisição das imagens de baixa resolução, tem-se o seguinte critério de otimização:

$$\begin{aligned} \epsilon(t)^2 = & \sum_{k=0}^{N-1} \|\mathbf{g}(t - k) - \mathbf{CB}(t - k)\mathbf{U}(t, k)\hat{\mathbf{d}}(t)\|_{\mathbf{W}(t, k)}^2 = \sum_{k=0}^{N-1} [\mathbf{g}(t - k) \\ & - \mathbf{CB}(t - k)\mathbf{U}(t, k)\hat{\mathbf{d}}(t)]^T \mathbf{W}(t, k) [\mathbf{g}(t - k) - \mathbf{CB}(t - k)\mathbf{U}(t, k)\hat{\mathbf{d}}(t)] \end{aligned} \quad (2.25)$$

que é similar ao critério apresentado em (2.20). A matriz $\mathbf{W}(t, k)$ funciona como

matriz de pesos positivos para todo t e k . A imagem ideal $\mathbf{d}(t)$, obtida minimizando $\epsilon(t)^2$, é a solução de:

$$\begin{aligned} \frac{\partial \epsilon(t)^2}{\partial \mathbf{d}(t)} = \mathbf{0} &\Rightarrow \sum_{k=0}^{N-1} \mathbf{U}(t, k)^T \mathbf{B}(t-k)^T \mathbf{C}^T \mathbf{W}(t, k) \hat{\mathbf{g}}(t-k) \\ &= \left[\sum_{k=0}^{N-1} \mathbf{U}(t, k)^T \mathbf{B}(t-k)^T \mathbf{C}^T \mathbf{W}(t, k) \mathbf{C} \mathbf{B}(t-k) \mathbf{U}(t, k) \right] \hat{\mathbf{d}}(t) \end{aligned} \quad (2.26)$$

que pode ser reescrita como:

$$\mathbf{p}(t) = \mathbf{R}(t) \hat{\mathbf{d}}(t) \quad (2.27)$$

onde

$$\mathbf{p}(t) = \sum_{k=0}^{N-1} \mathbf{U}(t, k)^T \mathbf{B}(t-k)^T \mathbf{C}^T \mathbf{W}(t, k) \mathbf{g}(t-k) \quad (2.28)$$

e

$$\mathbf{R}(t) = \sum_{k=0}^{N-1} \mathbf{U}(t, k)^T \mathbf{B}(t-k)^T \mathbf{C}^T \mathbf{W}(t, k) \mathbf{C} \mathbf{B}(t-k) \mathbf{U}(t-k) \quad (2.29)$$

Assumindo que temos um grande número de imagens em (2.28) e (2.29), ou seja, $N \rightarrow \infty$, e que a matriz de pesos $\mathbf{W}(t, \beta)$ tenha sua expressão alterada para $\lambda^\beta \hat{\mathbf{W}}$, sendo que $0 < \lambda < 1$, tal procedimento tem o objetivo de utilizar um decréscimo exponencial dos pesos de \mathbf{W} . Portanto, das expressões (2.28) e (2.29), tem-se:

$$\mathbf{p}(t) = \sum_{k=0}^{\infty} \lambda^\beta \mathbf{U}(t, \beta)^T \mathbf{B}(t-k)^T \mathbf{C}^T \mathbf{W}(t, k) \mathbf{g}(t-k) \quad (2.30)$$

$$\mathbf{R}(t) = \sum_{k=0}^{\infty} \lambda^\beta \mathbf{U}(t, \beta)^T \mathbf{B}(t-k)^T \mathbf{C}^T \hat{\mathbf{W}} \mathbf{C} \mathbf{B}(t-k) \mathbf{U}(t-k) \quad (2.31)$$

A fim de solucionar as equações (2.30) e (2.31), podemos reescrever $\mathbf{U}(t, k)$, visto na equação (2.23), da seguinte maneira:

$$\begin{aligned} \mathbf{U}(t, k) &= \mathbf{U}(t-1, k-1) \mathbf{U}(t, 1) = \dots \\ &= \mathbf{U}(t-k+1, 1) \mathbf{U}(t-k+2, 1) \mathbf{U}(t-k+3, 1) \dots \mathbf{U}(t-k, 1) \mathbf{U}(t, 1) \\ &= \prod_{j=1}^k \mathbf{U}(t-k+j, 1) \end{aligned} \quad (2.32)$$

onde $\mathbf{U}(t, 0) = \mathbf{I}$, definida como a matriz identidade. Dessa forma, as expressões (2.30) e (2.31) resultam nos seguintes filtros recursivos:

$$\mathbf{p}(t) = \lambda \mathbf{U}(t, 1)^T \mathbf{p}(t-1) + \mathbf{B}(t)^T \mathbf{C}^T \hat{\mathbf{W}} \mathbf{g}(t) \quad (2.33)$$

$$\mathbf{R}(t) = \lambda \mathbf{U}(t, 1)^T \mathbf{R}(t-1) \mathbf{U}(t, 1) + \mathbf{B}(t)^T \mathbf{C}^T \hat{\mathbf{W}} \mathbf{C} \mathbf{B}(t) \quad (2.34)$$

2.2.1 "Falso" algoritmo RLS

Para estimar a imagem super-resolução a partir da equação (2.27) em cada instante t , é necessário utilizar as equações (2.23), (2.24), (2.33) e (2.34). Essa solução é dada por:

$$\hat{\mathbf{d}}(t) = \mathbf{R}(t)^{-1}\mathbf{p}(t) \quad (2.35)$$

Contudo, a inversa de $\mathbf{R}(t)$ torna o processo impraticável pela carga computacional requerida para calculá-la em cada instante t . Como o critério é o dos mínimos quadrados (*Least-Squares*), o processo é chamado genericamente de "falso" algoritmo RLS, pois usa a análise feita em (2.32) para atualizar \mathbf{p} e \mathbf{R} . Usando o processo de otimização NSD (gradiente descendente normalizado - *normalized steepest descent*) para solucionar (2.27) iterativamente, resulta em:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu_j(t)[\mathbf{p}(t) - \mathbf{R}(t)\hat{\mathbf{d}}_j(t)] \quad (2.36)$$

sendo μ_j definido por:

$$\mu_j(t) = \frac{\mathbf{z}_j^T \mathbf{z}_j}{\mathbf{z}_j^T \mathbf{R}(t) \mathbf{z}_j} \quad (2.37)$$

onde $\mathbf{z}_j = \mathbf{p}(t) - \mathbf{R}(t)\hat{\mathbf{d}}(t)$.

A cada momento, são realizadas β iterações e no próximo instante t o vetor $\hat{\mathbf{d}}(t)$ é inicializado usando a última solução obtida do instante anterior.

2.2.2 O algoritmo LMS

No intuito de formar a imagem com super-resolução, o algoritmo LMS procura minimizar uma função custo \mathbf{J}_{MS} dada pelo erro médio quadrático entre as imagens de baixa resolução e suas respectivas estimativas, obtidas reduzindo-se a resolução da estimativa da imagem HR (após compensar pelo deslocamento entre as imagens observadas). Usando-se um algoritmo iterativo do tipo gradiente descendente, obtém-se a seguinte equação de atualização da estimativa da imagem HR:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) - \frac{1}{2}\mu\nabla\mathbf{J}_{MS}(t) \quad (2.38)$$

onde $\nabla\mathbf{J}_{MS}(t)$ é o respectivo vetor gradiente.

Sendo $\epsilon(t)$ a diferença entre a t -ésima imagem de baixa resolução e a imagem estimada de alta resolução decimada, a função custo é dada por:

$$\mathbf{J}_{MS}(t) = E[\|\epsilon(t)\|^2] \quad (2.39)$$

e pode ser reescrita como:

$$\mathbf{J}_{MS}(t) = E[\|\mathbf{g}(t) - \mathbf{B}(t)\hat{\mathbf{d}}(t)\|^2] \quad (2.40)$$

onde $\mathbf{B}(t)$ representa a decimação aplicada à imagem de alta resolução atual (t -ésimo quadro). Da equação acima, obtém-se a seguinte expressão para o vetor gradiente:

$$\begin{aligned} \nabla \mathbf{J}_{MS}(t) &= \frac{\partial \mathbf{J}_{MS}(t)}{\partial \hat{\mathbf{d}}(t)} \\ &= -2\mathbf{B}(t)[E[\mathbf{g}(t)] - \mathbf{B}(t)\hat{\mathbf{d}}(t)] \end{aligned} \quad (2.41)$$

Substituindo-se este resultado na equação (2.38), temos:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu\mathbf{B}(t)[E[\mathbf{g}(t)] - \mathbf{B}(t)\hat{\mathbf{d}}(t)] \quad (2.42)$$

Podemos aproximar o valor esperado pelo seu valor instantâneo (ou seja, considerando apenas uma observação), obtendo a seguinte equação para o algoritmo LMS:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu\mathbf{B}(t)[\mathbf{g}(t) - \mathbf{B}(t)\hat{\mathbf{d}}(t)] \quad (2.43)$$

para cada t com β iterações. Para iniciar o algoritmo, é utilizada uma das imagens de baixa resolução interpolada como $\mathbf{g}(1)$. No Capítulo 4 serão mostrados os resultados da aplicação deste algoritmo para super-resolução de algumas imagens com diferentes razões sinal-ruído.

Capítulo 3

Métodos de Super-Resolução utilizando a Transformada Wavelet

Neste trabalho, foram geradas imagens de baixa resolução ruidosas a partir de deslocamentos de sub-pixels aplicados a uma imagem original para que esta sirva de comparação ao final do experimento. Consideramos uma sequência de deslocamentos conhecidos, mostrada na figura 3.1.

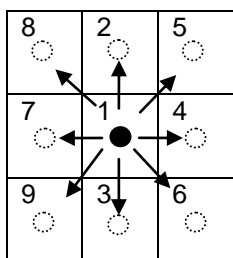


Figura 3.1: Deslocamentos das imagens para os experimentos.

A figura 3.2 descreve o procedimento proposto neste trabalho, que inicia com as N imagens deslocadas de dimensão $L \times L$, que serão reduzidas para dimensão $M \times M$, com a inserção de ruído aleatório n_k , onde $k = 1, 2, \dots, N$.

A figura 3.3 mostra a aplicação da transformada wavelet às imagens de baixa resolução, gerando para cada uma delas quatro subimagens contendo, respectivamente, uma aproximação (LL), detalhes horizontais (LH), detalhes verticais (HL) e detalhes diagonais (HH). Em seguida utilizamos um dos métodos de super-resolução descritos no Capítulo 2 a cada conjunto de imagens de baixa resolução das diferentes sub-bandas LL, LH, HL e HH. A imagem final é obtida combinando-se as estimativas das imagens SR das sub-bandas LL, LH, HL e HH. Como usaremos as propriedades de cada uma das subimagens para obter as soluções de Wiener e adaptativa, melhores resultados, principalmente para imagens ruidosas,

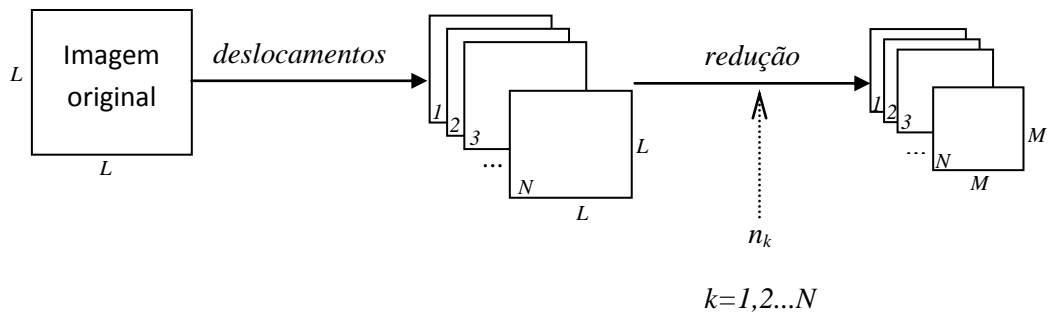


Figura 3.2: Geração da imagens de baixa resolução

são esperados e posteriormente todos os resultados serão comparados à imagem HR original.

Procurando aprimorar os métodos para imagens ruidosas, foram incluídos métodos de remoção de ruído (*denoising*) com wavelets, como passo inicial das técnicas de super-resolução.

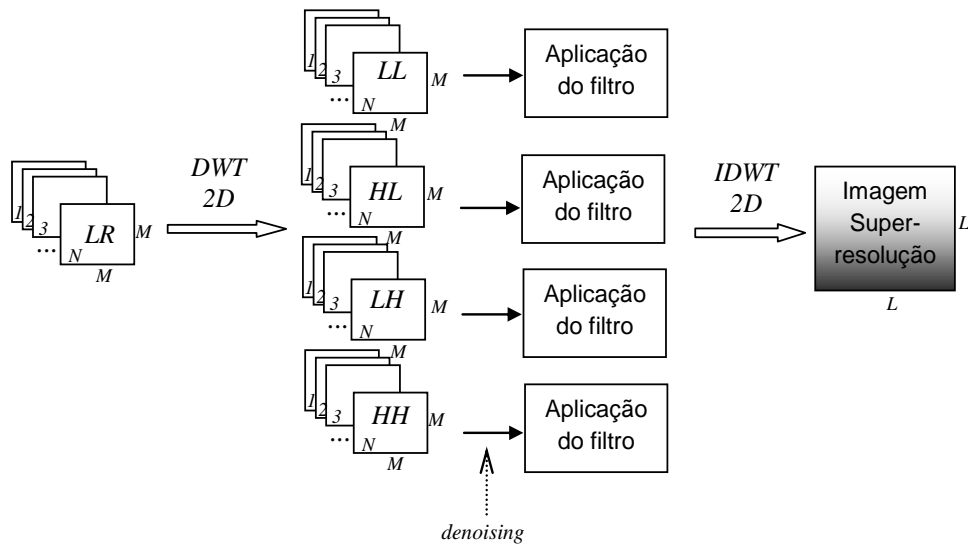


Figura 3.3: Métodos de super-resolução usando a transformada wavelet

Na próxima seção apresentaremos a teoria básica da transformada wavelet, cujos conceitos foram obtidas das referências de Mallat [8] e [17] e Nguyen [10]. Em seguida descreveremos os métodos de super-resolução propostos com a transformada wavelet e a técnica de *denoising* utilizada para remoção de ruído.

3.1 Transformada Wavelet

Wavelets são funções matemáticas que permitem dividir um sinal em diferentes escalas, e estudar cada componente com a resolução igual à sua escala. Nossa percepção do universo utiliza o conceito de escala, onde cada observação realizada é feita em uma escala adequada para a compreensão dos diferentes detalhes de que precisamos.

O procedimento de análise por wavelet consiste em adotar uma função base, denominada wavelet mãe, que serve como protótipo para todas as funções usadas no processo. As funções usadas são versões dilatadas ou comprimidas da função escala $\phi(t)$ e da wavelet mãe $\psi(t)$, podendo também ser transladadas no tempo. A geração das wavelets de Haar é ilustrada na figura 3.4.

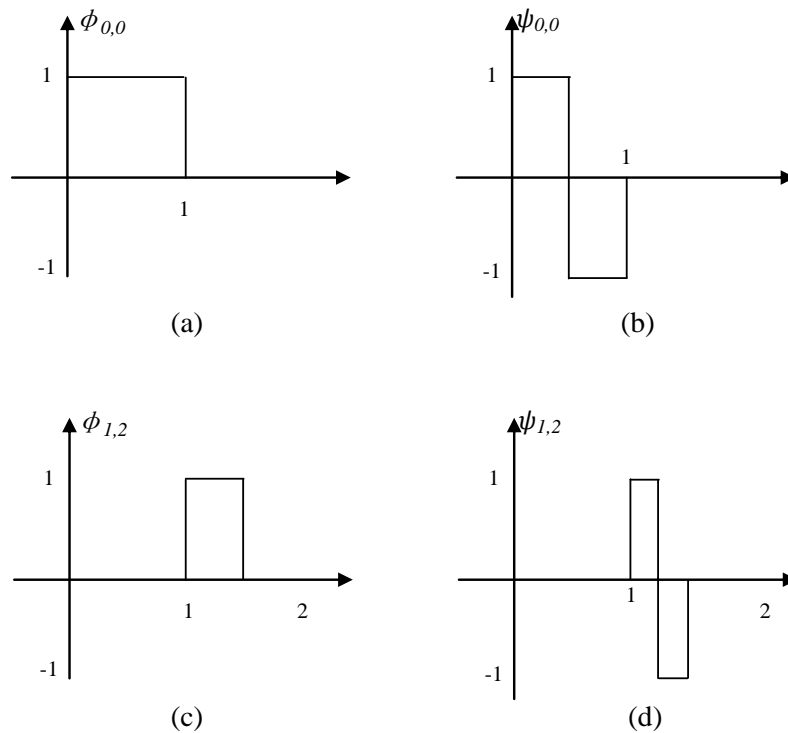


Figura 3.4: Wavelets de Haar: (a) Função escala $\phi_{0,0}$; (b) Wavelet mãe $\psi_{0,0}$; (c) Função escala $\phi_{1,2}$; (d) Wavelet mãe $\psi_{1,2}$.

Há duas versões da transformada wavelet, a contínua e a discreta. A contínua é usada principalmente na análise e caracterização de detalhes de sinais, enquanto a discreta é mais apropriada para compressão de dados e reconstrução de sinais. A transformada wavelet contínua (CWT) é a integral, ao longo do tempo, do sinal multiplicado pela função wavelet em certa escala a e deslocada de certa quantidade b . Este processo produz coeficientes $\psi_{a,b}$ de wavelet que são funções da escala e da

posição. Matematicamente, segundo Burrus [18], pode-se definir a CWT de uma função $f(t)$ como o produto interno de $f(t)$ e $\psi_{a,b}$, ou seja,

$$F(a, b) = \langle f(t), \psi_{a,b}(t) \rangle = \int_{-\infty}^{+\infty} f(t)\psi_{a,b}(t)dt \quad (3.1)$$

sendo

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right) \quad (3.2)$$

onde a wavelet mãe $\psi(t)$ possui a seguinte propriedade:

$$\int_{-\infty}^{+\infty} \psi(t)dt = 0 \quad (3.3)$$

e a constante $\frac{1}{\sqrt{a}}$ é usada para normalização da energia da função wavelet nas diferentes escalas.

A sua inversa é dada por

$$f(t) = \frac{1}{C} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{1}{|a|^2} W(a, b)\psi_{a,b}(t)da db \quad (3.4)$$

onde

$$C = \int_{-\infty}^{\infty} \frac{|\Psi(w)|^2}{|w|} dw$$

e

$$\Psi(w) = \int_{-\infty}^{\infty} e^{-j\omega t} dt$$

De acordo com a figura 3.4, quanto menor a escala a , mais a wavelet se comprime, analisando detalhes de frequências altas; quanto maior a escala a , mais a wavelet se dilata, analisando características globais, ou seja, as frequências baixas. Como na CWT a obtenção dos coeficientes de wavelet para cada escala necessita de uma grande quantidade de cálculo, a alternativa é abordar de uma forma discreta, cuja implementação computacional é direta, dispensando estágios sensíveis a erros, sem causar prejuízo na qualidade final.

A função escala $\phi(t)$ e a função wavelet mãe $\psi(t)$ podem ser geradas a partir de um par de filtros digitais passa-baixas $H_0(z)$ e passa-altas $H_1(z)$, que devem satisfazer certas condições para que seja possível representar qualquer função de energia finita como uma combinação linear destas funções comprimidas e deslocadas, conforme exposto por Vaidyanathan [19].

Neste trabalho foi utilizada a transformada Haar discreta com um único nível de decomposição, sendo armazenados separadamente uma aproximação (LL) e os detalhes horizontal (HL), vertical (LH) e diagonal (HH) da imagem, como ilustrado na figura 3.5(a). Com mais níveis de decomposição, este processo é aplicado novamente na imagem LL, como mostrado na figura 3.5(b).

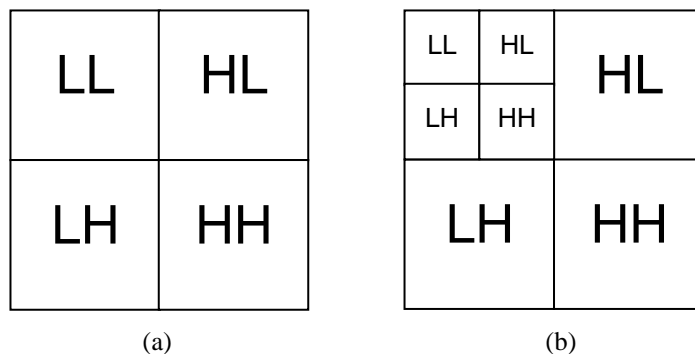


Figura 3.5: (a) Transformada wavelet bidimensional padrão; (b) Transformada aplicada na sub-banda LL.

A decomposição de cada imagem LR foi feita através da convolução bidimensional, aplicando filtros passa-baixas e passa-altas, separadamente, nas linhas e nas colunas. Foram utilizados os filtros de Haar, $H_0(z) = 1 + z^{-1}$ and $H_1(z) = 1 - z^{-1}$, no entanto, não foi feita decimação (redução da dimensão das imagens LR decompostas) para que não houvesse efeito de *aliasing*. Portanto, cada imagem LR resultou em quatro imagens de mesma dimensão, as quais foram processadas separadamente, conforme descrito nas próximas seções.

3.2 Método de Wiener com a transformada wavelet

No método proposto, a filtragem de Wiener é aplicada a cada conjunto de sub-bandas, ou seja, a banda de aproximação (LL) e as com detalhes (HL, LH e HH) das imagens LR são tratados separadamente, levando-se em conta as suas características. Portanto cada conjunto de sub-banda das imagens de baixa resolução gera uma imagem de resolução superior da sub-banda correspondente, que serão convoluídas bi-dimensionalmente com seus respectivos filtros de reconstrução e a imagem super-resolução é obtida a partir do somatório desses resultados.

Seja $\mathbf{g}_{i,j,1}$ o vetor que contém os pixels da i -ésima janela de observação da imagem da aproximação LL da j -ésima imagem LR, $\mathbf{g}_{i,j,2}$ o vetor correspondente da imagem HL, $\mathbf{g}_{i,j,3}$ o vetor correspondente da imagem LH e $\mathbf{g}_{i,j,4}$ o vetor correspondente da imagem HH. Seja

$$\mathbf{g}_{i,l} = [\mathbf{g}_{i,1,l}^T, \mathbf{g}_{i,2,l}^T, \dots, \mathbf{g}_{i,P,l}^T]^T \quad (3.5)$$

o vetor com as observações da l -ésima banda das P imagens LR utilizadas para estimar os pixels da i -ésima janela da imagem desejada HR correspondente.

Portanto, o vetor $\mathbf{d}_{i,l}$ com estes pixels é obtido a partir da equação:

$$\hat{\mathbf{d}}_{i,l} = \mathbf{W}_{i,l}^T \mathbf{g}_{i,l} \quad (3.6)$$

A matriz de pesos $\mathbf{W}_{i,l}$ é obtida usando os mesmos modelos para a função PSF e para a função autocorrelação descritos no Capítulo 2 e considerando as estatísticas locais, ou seja, usando estimativas das variâncias dos sinais em cada sub-banda considerando cada região da imagem. Este processo é realizado pela equação:

$$\mathbf{W}_{i,l} = \left[E[\mathbf{f}_i \mathbf{f}_i^T] + \frac{\sigma_n^2}{\hat{\sigma}_{d_{i,l}}^2} \mathbf{I} \right]^{-1} E[\mathbf{f}_i \mathbf{d}_i^T] \quad (3.7)$$

A utilização de diferentes razões sinal-ruído para as diferentes bandas como descrito na equação acima tem o intuito de que o algoritmo de Wiener trate as mesmas de maneira diferenciada.

Para melhorar o desempenho do algoritmo de Wiener para imagens LR muito ruidosas, foram realizados tratamentos para remoção de ruído (*denoising*) no domínio da transformada wavelet. Na Seção 3.4 é apresentada uma breve descrição da técnica utilizada e os resultados de sua aplicação em conjunto com as técnicas de super-resolução de Wiener e adaptativa são comparados no Capítulo 4.

Na formação da imagem super-resolução final, foi realizada para cada sub-banda a convolução bi-dimensional da imagem obtida da aplicação do filtro de Wiener com as respostas impulsivas dos filtros de reconstrução correspondentes da wavelet utilizada. Em seguida, foi realizada a soma das imagens resultantes das quatro sub-bandas com resolução maior.

3.3 Algoritmo adaptativo com a transformada wavelet

No algoritmo adaptativo com a transformada wavelet, a análise em sub-bandas foi aplicada às imagens LR. As imagens resultantes podem ser modeladas conforme a teoria vista no Capítulo 2, ou seja, a imagem LR na l -ésima sub-banda pode ser escrita como:

$$\mathbf{g}_{k,l} = \mathbf{C}_k \mathbf{B}_k \mathbf{U}_k \mathbf{d}_{k,l} + \mathbf{e}_{k,l} \quad (3.8)$$

onde $\mathbf{d}_{k,l}$ é o vetor com os L^2 pixels da imagem de alta resolução da l -ésima sub-banda; \mathbf{C}_k é o operador de decimação; \mathbf{B}_k é a matriz correspondente a algum tipo de borramento; \mathbf{U}_k é a matriz que representa a inserção de distorção geométrica; e $\mathbf{e}_{k,l}$ representa ruído aleatório, o qual pode ser diferente para cada imagem e sub-

banda. A equação resultante para a atualização da imagem HR estimada de cada sub-banda pelo algoritmo LMS, usando a mesma notação empregada no Capítulo 2 (correspondente à aquisição temporal das imagens LR) na qual denominamos de $\hat{\mathbf{d}}_{j,l}(t)$ a estimativa da imagem de maior resolução da sub-banda l no instante t na j -ésima iteração, é dada por:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu \sum_{l=1}^4 \mathbf{B}(t)^T \mathbf{C}^T \hat{\mathbf{W}}[\mathbf{g}_l(t) - \mathbf{C}\mathbf{B}(t)\hat{\mathbf{d}}_{j,l}(t)] \quad (3.9)$$

sendo a atualização da imagem feita através de uma parcela μ da interpolação da diferença entre a imagem LR atual da sub-banda l e a estimativa da subimagem HR $\mathbf{d}_{t,l}$ decimada. O fator μ foi definido experimentalmente, com o objetivo de encontrar o menor valor de erro médio quadrático.

Para iniciar a iteração, foi utilizada a versão interpolada da imagem de baixa resolução, no caso definida por $\hat{\mathbf{d}}_0(t)$. Neste trabalho, utilizamos a mesma função (erro médio quadrático) para iteração das imagens nas diferentes sub-bandas. No entanto, diferentes funções custos poderiam ser utilizadas, com possível melhoria no desempenho do algoritmo.

O algoritmo de remoção de ruído (*denoising*) no domínio da transformada wavelet foi também combinado com o método adaptativo para melhorar o seu desempenho para imagens fortemente contaminadas por ruído.

3.4 Redução de ruído (*denoising*) com wavelets

Um modo de reduzir os níveis de ruído presente nas imagens de baixa resolução foi combinar as técnicas de *denoising* usando a transformada wavelet com as técnicas de super-resolução descritas nas seções acima.

O método de *denoising* consiste na análise da imagem através dos histogramas dos seus detalhes horizontais, diagonais e verticais, e na escolha de limiares (*thresholds*) λ , conforme ilustrado em linhas tracejadas na figura 3.6.

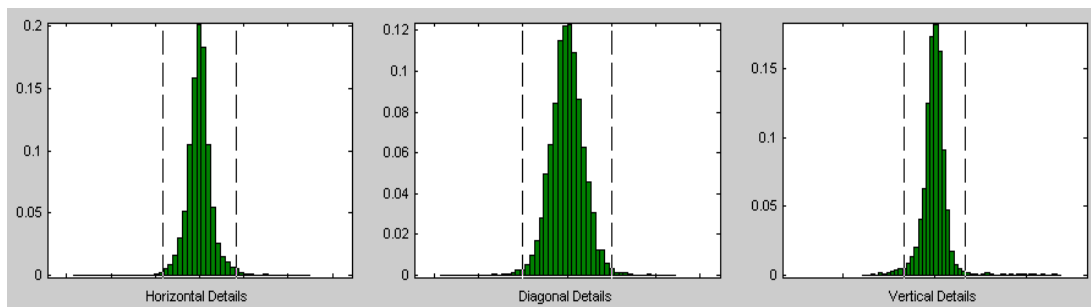


Figura 3.6: Definição de limiares nos detalhes horizontal, diagonal e vertical.

Este procedimento elimina (zera) os coeficientes com valores abaixo dos limiares escolhidos, os quais julga-se que correspondam a ruído. Um limiar muito grande pode ocasionar perda de informação importante, causando danos e distorções à imagem. No caso de limiares muito baixos, pouco ruído é extraído. Segundo Donoho [20], a saída B resultante da aplicação do limiar a uma entrada A pode ser obtida pela equação (3.10), correspondente ao algoritmo *hard threshold*, ou pela equação (3.11), correspondente ao algoritmo *soft threshold*.

$$B_{hard}(x, y) = \begin{cases} A(x, y), & |A(x, y)| > \lambda \\ 0, & |A(x, y)| \leq \lambda \end{cases} \quad (3.10)$$

$$B_{soft}(x, y) = \begin{cases} [\text{sign}[A(x, y)]] [|A(x, y)| - \lambda], & |A(x, y)| > \lambda \\ 0, & |A(x, y)| \leq \lambda \end{cases} \quad (3.11)$$

Neste trabalho, optamos por usar o algoritmo *soft threshold*, que em geral apresenta melhores resultados por evitar grande perda de informação [20]. A escolha do limiar λ pode ser determinado de diferentes maneiras. Segundo Donoho [21], o limiar pode ser calculado por:

$$\lambda = \sigma \sqrt{2 \log(M)} \quad (3.12)$$

onde σ é o desvio padrão do ruído e M é o número de pixels da imagem. O ruído pode ser estimado por diversos métodos, como por exemplo, através de banco de filtros proposto por Olsen [22] e também por operador Laplaciano de Shen [23].

No próximo capítulo apresentaremos os resultados obtidos com a utilização desta técnica nos métodos de super-resolução de Wiener e adaptativo.

Capítulo 4

Resultados

Este capítulo apresenta os experimentos realizados com os algoritmos discutidos nos Capítulos 2 e 3 e seus resultados. A descrição de como foram obtidas as imagens de baixa resolução bem como os parâmetros e inicializações utilizados em cada experimento são detalhados. As imagens resultantes dos diferentes algoritmos de super-resolução são apresentadas e comparadas entre si, para diferentes razões sinal-ruído.

Para realizar comparação dos desempenhos de cada método, foi usada a média dos erros quadráticos [24] e o programa de similaridade de imagens, denominada SSIM (*Structural Similarity*), proposto por Wang et.al [25].

A média dos erros quadráticos é obtida através da seguinte equação:

$$MSE = \frac{1}{M^2} \sum_{x=1}^M \sum_{y=1}^M (a_D(x, y) - a_{SR}(x, y))^2 \quad (4.1)$$

onde M é a altura e largura da imagem, visto que foram adotadas imagens quadradas para os experimentos; a_D é o valor do pixel da imagem original na posição (x, y) e a_{SR} é o valor do pixel da imagem super-resolução na posição (x, y) .

O método de comparação por similaridade usado foi obtido do *site* [26] disponibilizado por Wang. O algoritmo calcula medidas de similaridade através de três comparações: de luminância, contraste e estrutura, que depois serão combinadas para fornecer o resultado de similaridade.

Os resultados dos métodos de avaliação descritos acima para os diferentes algoritmos de super-resolução, considerando imagens com diferentes conteúdos e razões sinal-ruído, são apresentados em tabelas mostradas após as sequências de imagens.

4.1 Descrição dos Experimentos

Todos os experimentos realizados utilizaram sequências de nove imagens de baixa resolução para gerar a estimativa da imagem HR, com aumento de resolução por um fator igual a três em cada dimensão.

As imagens de baixa resolução foram obtidas a partir de uma única imagem original de alta resolução, deslocada e reduzida por fator três. Para realizar essa redução, foi utilizada a média aritmética de janelas 3×3 da imagem HR para obter as imagens LR. Foram realizados deslocamentos que são considerados conhecidos e sua sequência de registro foi mostrada na figura 3.1.

Para simular a degradação das imagens durante o processo de aquisição, ruídos foram inseridos após a redução das dimensões das imagens deslocadas. Experimentos foram realizados sem ruído e com ruído de medição aleatório de variâncias 0,03 e 0,05.

A seguir descrevemos os detalhes das simulações realizadas para cada algoritmo.

4.1.1 Filtragem de Wiener

Foi gerada uma matriz função autocorrelação de dimensão 30×30 (a mesma ilustrada na figura 2.5) e uma matriz de mesma dimensão, $\mathbf{H}(x, y)$, denominada função retangular e ilustrada na figura 4.1, para formar as matrizes \mathbf{R}_i e \mathbf{P}_i .

Na sequência, foi realizada a convolução de elementos dessas duas matrizes, de autocorrelação e retangular, para o resultado ser utilizado na formação da matriz \mathbf{P} , e uma segunda convolução para formar a matriz \mathbf{R} . As matrizes \mathbf{P} e \mathbf{R} são formadas a partir de valores determinados pela sequência dos deslocamentos das imagens de baixa resolução.

Para utilizarmos as estatísticas locais das janelas da imagem, foi formada uma matriz identidade multiplicada pelo valor da razão das variâncias do ruído de medição e de cada imagem de baixa resolução na janela considerada. Em seguida, esta matriz foi somada à matriz \mathbf{R} , obtendo-se a matriz de coeficientes ótimos para gerar os pixels da estimativa da i -ésima janela da imagem HR, ou seja:

$$\mathbf{W}_i = \left[\mathbf{R} + \frac{\sigma_n^2}{\sigma_{d_i}^2} \mathbf{I} \right]^{-1} \mathbf{P} \quad (4.2)$$

Deste modo, é obtida a matriz \mathbf{W}_i , de dimensão $K \times (d_x d_y)$, e como na proposta foi definida o uso de nove imagens de baixa resolução, a dimensão da janela a ser estimada $d_x \times d_y$ é de 3×3 e a dimensão da janela de observação é 9×9 , resultando em $K = 81$, da equação (2.3) do Capítulo 2. A matriz \mathbf{W}_i é então normalizada para garantir continuidade do resultado entre janelas de observação, evitando assim a introdução de efeitos quadriculados nas imagens. Para realizar esta normalização,

os elementos de cada coluna da matriz \mathbf{W}_i são somados e os resultados são inseridos numa matriz diagonal. O produto de \mathbf{W}_i pela inversa da matriz diagonal obtida fará com que cada coluna da matriz resultante tenha elementos com soma igual a 1.

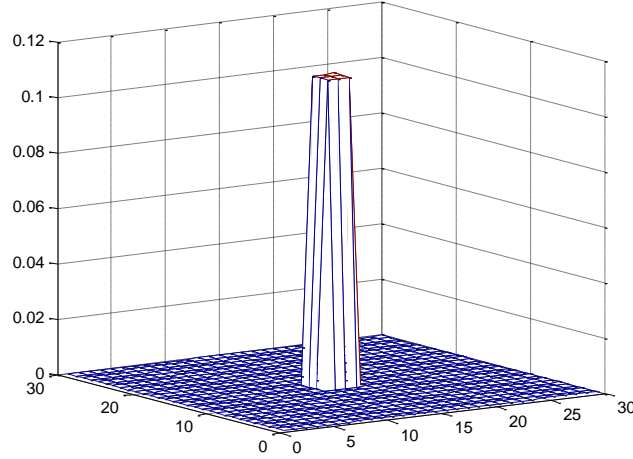


Figura 4.1: Função $H(x, y) = (1/9)rect(x/3, y/3)$

A transposta da matriz \mathbf{W} é multiplicada pelo vetor de dimensão 81×1 , formado por blocos de tamanho 3×3 das nove imagens de baixa resolução, resultando em um vetor de nove elementos que é reestruturado em uma matriz 3×3 . Esse processo é repetido, deslocando-se esses blocos pixel a pixel, definindo a imagem de super-resolução desejada.

No método de filtragem de Wiener com transformada wavelet, foi empregada a wavelet Haar (ou Daubechies 1) com um nível de decomposição e sem decimação. As quatro imagens resultantes foram tratadas de forma independente utilizando o algoritmo de Wiener descrito acima. Como as estatísticas locais são diferentes para cada imagem, obteve-se uma matriz de coeficientes distinta para ser aplicada em cada imagem resultante da decomposição. Portanto a equação (4.2) foi modificada no método proposto conforme descrito na equação (4.3) para a i -ésima janela de observação da l -ésima banda.

$$\mathbf{W}_{i,l} = \left[E[\mathbf{f}_i \mathbf{f}_i^T] + \frac{\sigma_n^2}{\hat{\sigma}_{d_{i,l}}^2} \mathbf{I} \right]^{-1} E[\mathbf{f}_i \mathbf{d}_i^T] \quad (4.3)$$

4.1.2 Filtragem Adaptativa

Nesse processo foram também utilizadas nove imagens de baixa resolução obtidas por um processo semelhante ao do algoritmo anterior, como ilustrado na figura 3.1.

Esse algoritmo é inicializado com a imagem interpolada, gerada pelo método do vizinho mais próximo, da primeira imagem de baixa resolução adquirida.

A cada iteração é adicionada uma fração μ da interpolação do resultado da diferença entre a imagem de baixa resolução correspondente e a imagem atual reduzida por média aritmética. Este processo é repetido $N_{it} = 30$ vezes antes de se utilizar a próxima imagem de baixa resolução na equação de atualização, dada por:

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu \mathbf{B}(t)^T \mathbf{C}^T \hat{\mathbf{W}}[\mathbf{g}(t) - \mathbf{CB}(t)\hat{\mathbf{d}}_j(t)] \quad (4.4)$$

Na aplicação do algoritmo de filtragem adaptativa foi usado $\mu = 0,2$; no entanto, nos experimentos com a transformada wavelet foi necessário reduzir o valor de μ para 0,08. A transformada Haar com um nível de decomposição também foi usada nestes experimentos, gerando quatro imagens para cada imagem LR. E a equação (4.4) que corresponde ao método proposto com filtragem adaptativa é descrita pela equação (4.5), onde $\hat{\mathbf{d}}_{j,l}(t)$ contém os pixels correspondentes a l -ésima sub-banda de $\hat{\mathbf{d}}_j(t)$.

$$\hat{\mathbf{d}}_{j+1}(t) = \hat{\mathbf{d}}_j(t) + \mu \sum_{l=1}^4 \mathbf{B}(t)^T \mathbf{C}^T \hat{\mathbf{W}}[\mathbf{g}_l(t) - \mathbf{CB}(t)\hat{\mathbf{d}}_{j,l}(t)] \quad (4.5)$$

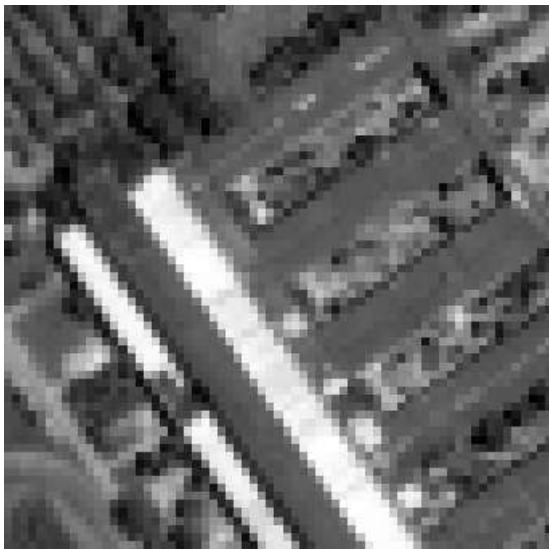
4.2 Resultados dos Experimentos

Os resultados dos experimentos com os algoritmos de super-resolução serão mostrados a seguir, comparando-se as imagens originais, de baixa resolução, interpoladas pelo método bilinear e obtidas pelos diferentes métodos descritos acima. Foram adicionados ruídos de medição com diferentes variâncias às imagens testadas e os resultados dos diferentes métodos foram comparados. Os testes foram realizados com imagens de satélite (Ilha do Fundão), de um texto (Filtros Adaptativos) e de um rosto (Charlize) com o objetivo de analisar o desempenho de cada método com imagens de diferentes conteúdos.

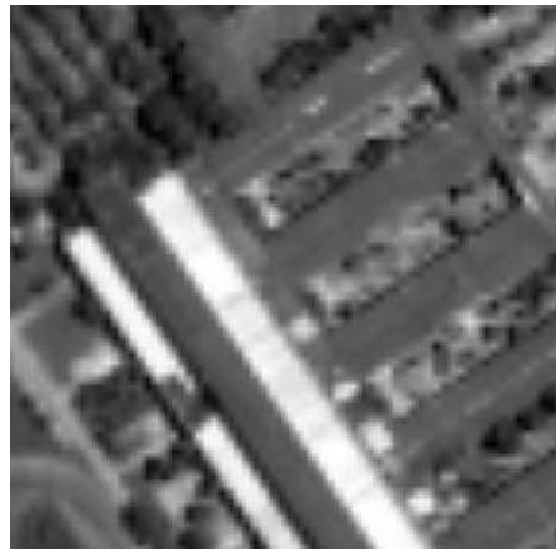
As figuras 4.2 a 4.10 contém as imagens de satélite obtidas pelos algoritmos descritos neste trabalho, enquanto que as tabelas 4.1 e 4.2 comparam quantitativamente esses resultados através do erro médio quadrático e do SSIM. Os valores 0, 0,03 e 0,05 são as variâncias do ruído de medição utilizados, sendo que para o nível 0 (sem ruído), não foi implementado o método de *denoising*.



(a)



(b)



(c)



(d)



(e)

Figura 4.2: Imagem de satélite: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.

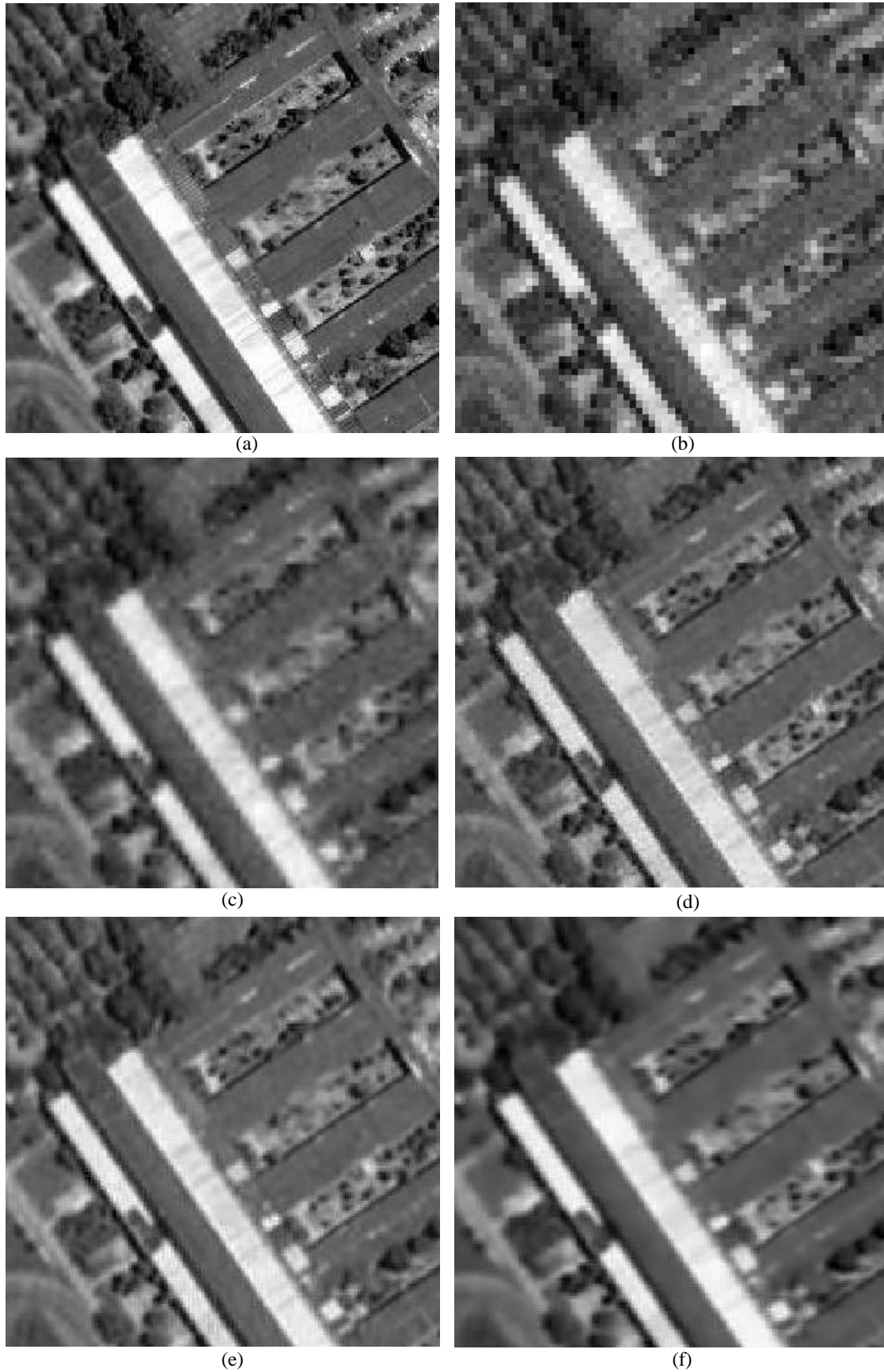


Figura 4.3: Imagem de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.

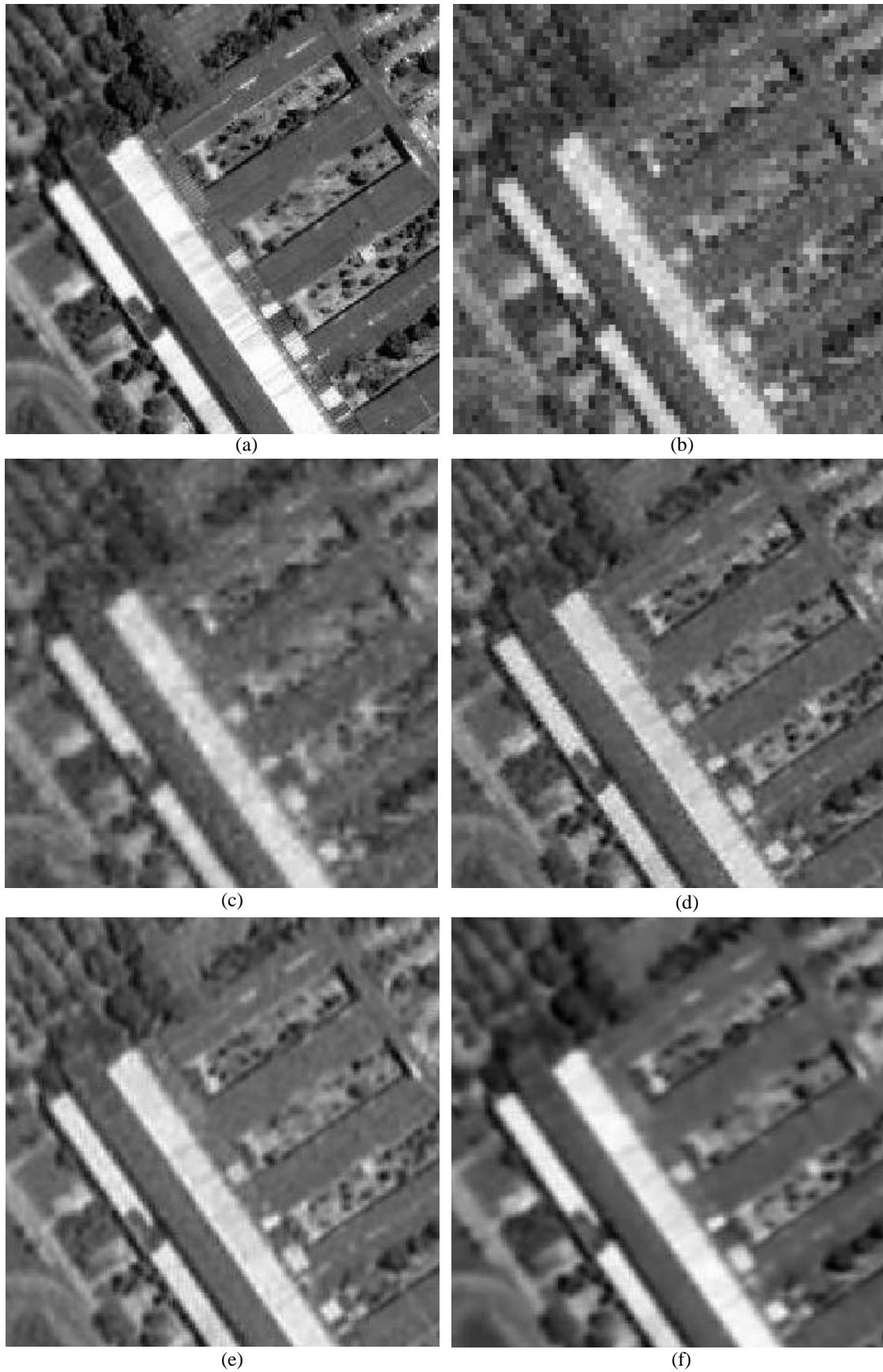
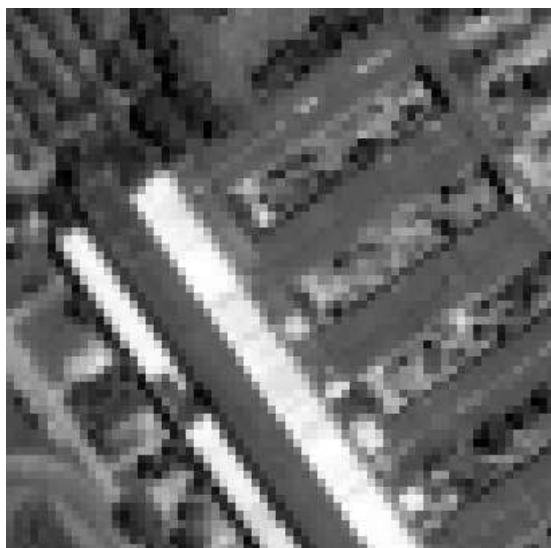


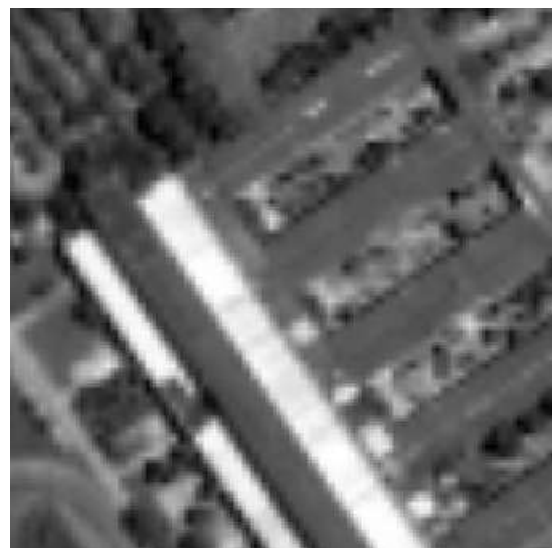
Figura 4.4: Imagem de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.



(a)



(b)



(c)



(d)



(e)

Figura 4.5: Imagens de satélite: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem de adaptativa e Wavelet.

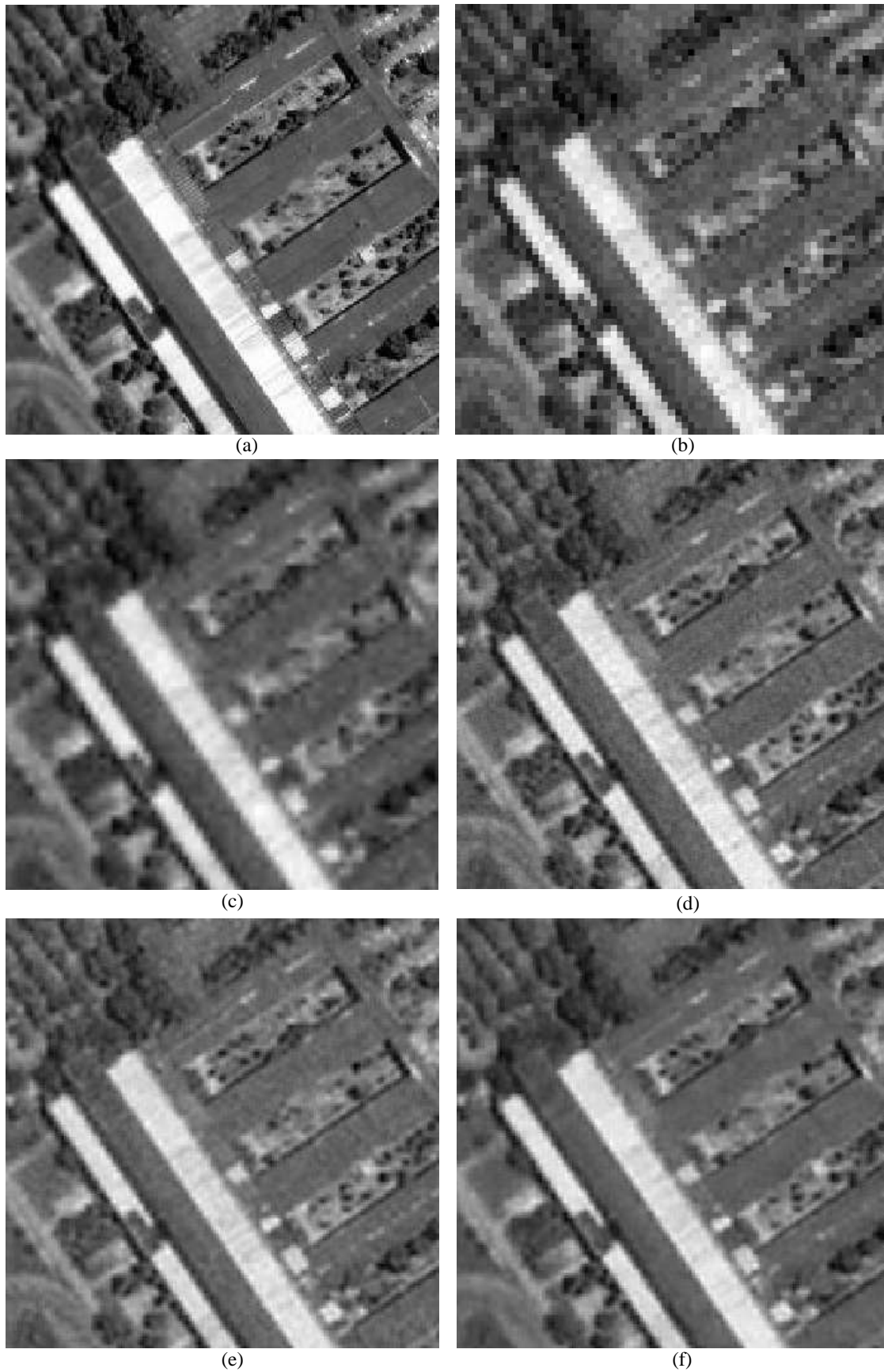


Figura 4.6: Imagens de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.

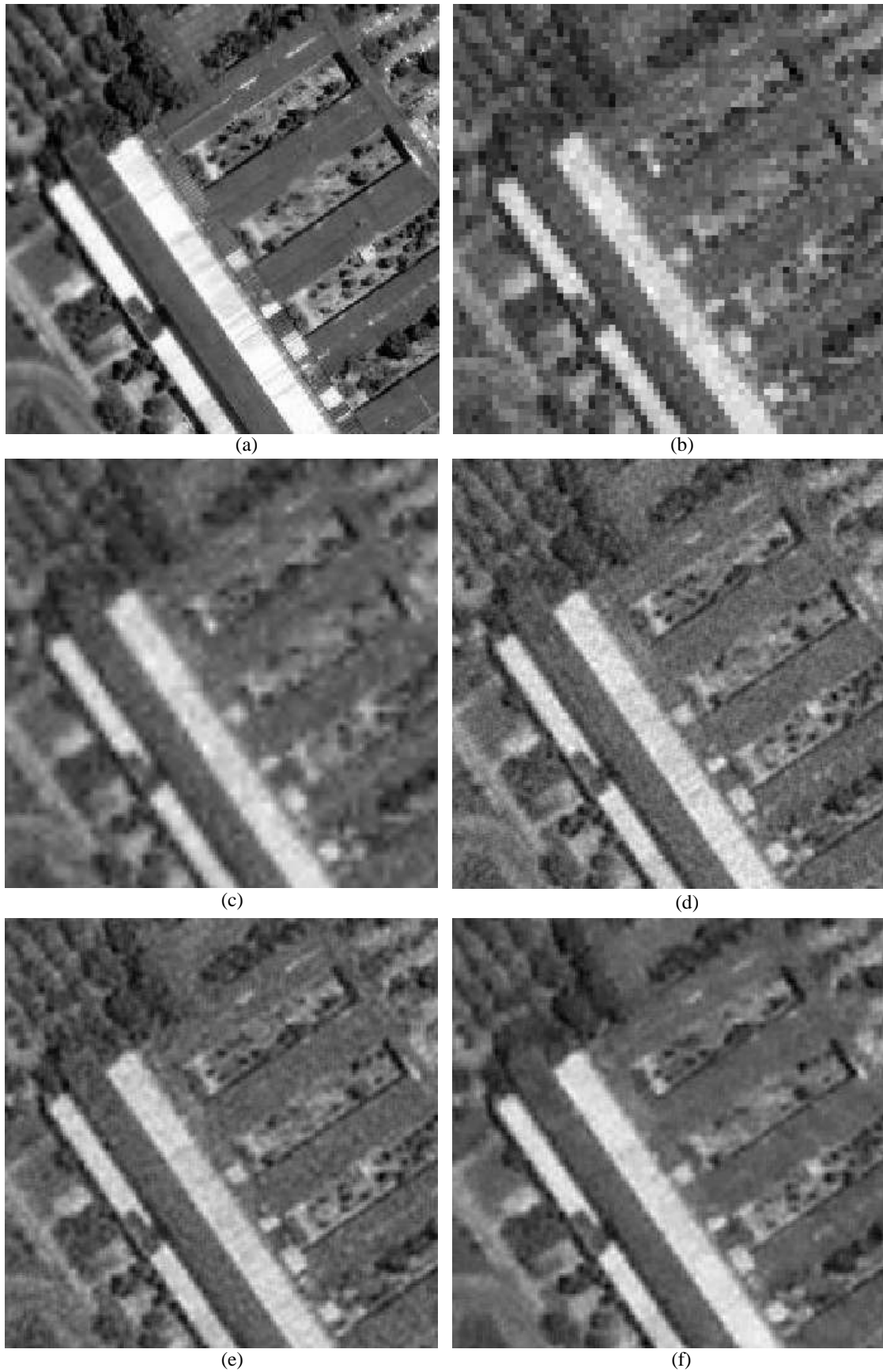


Figura 4.7: Imagens de satélite: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.

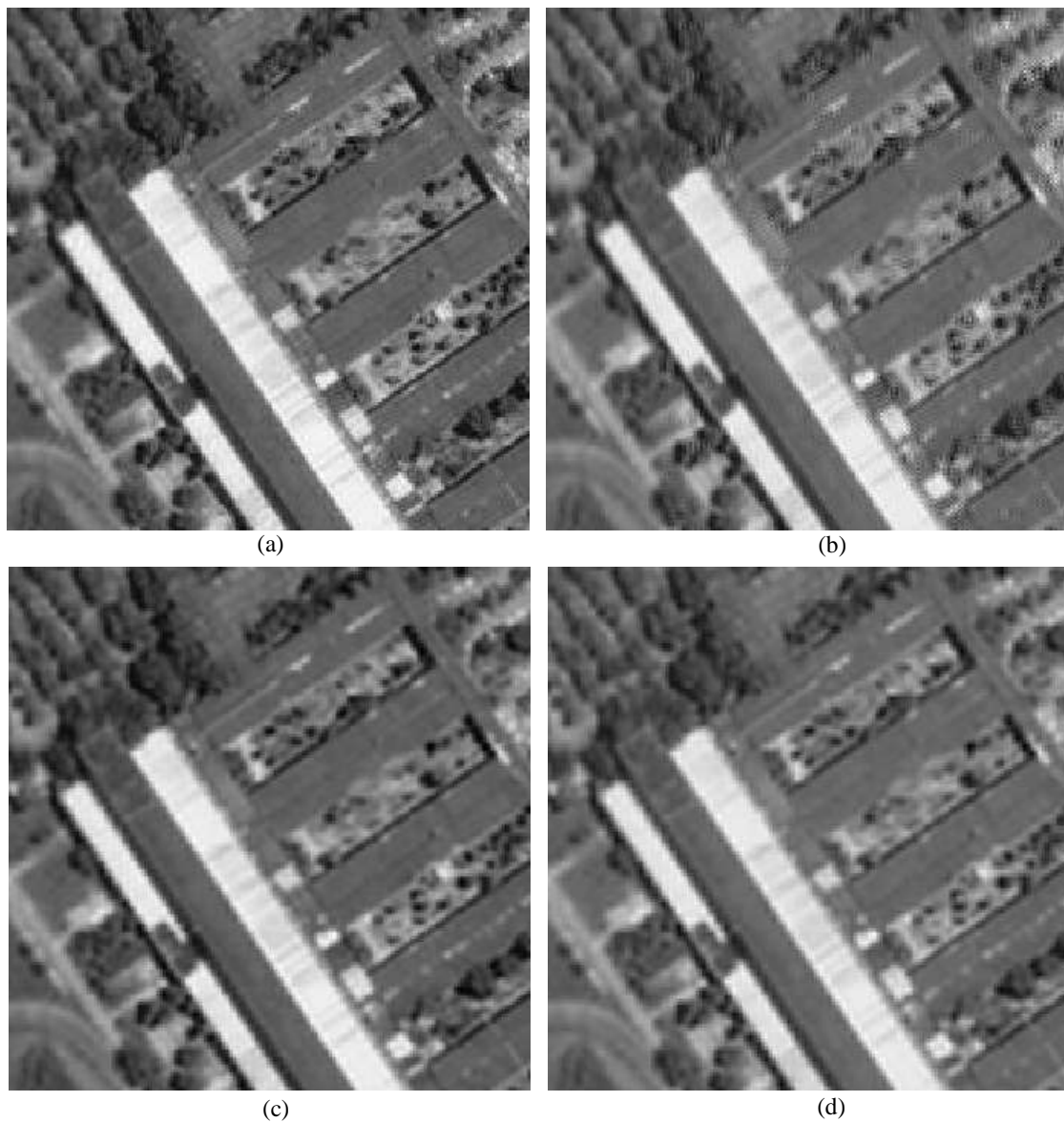


Figura 4.8: Imagens Super-resolução de satélite sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.

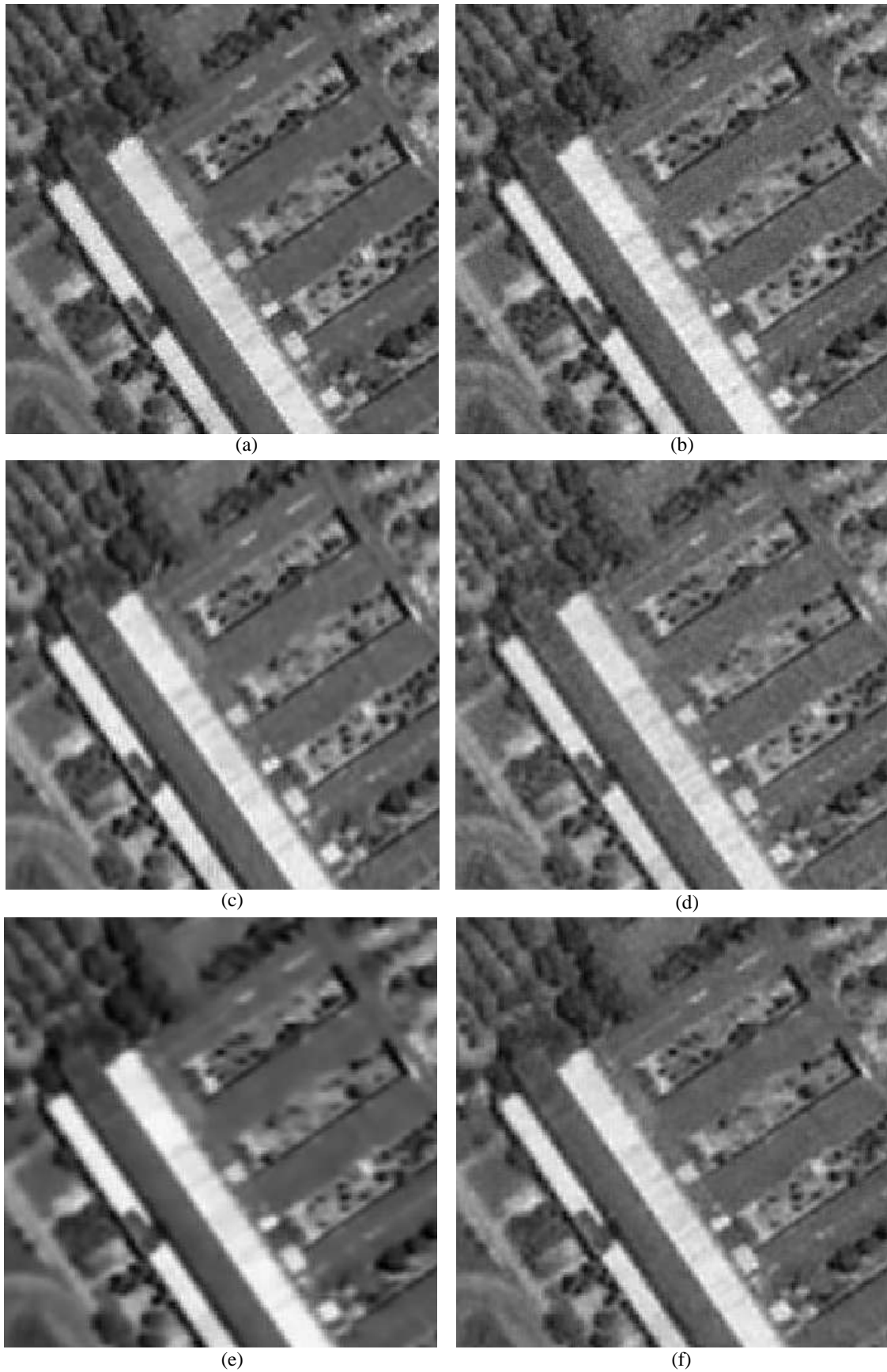


Figura 4.9: Imagens Super-resolução de satélite com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.

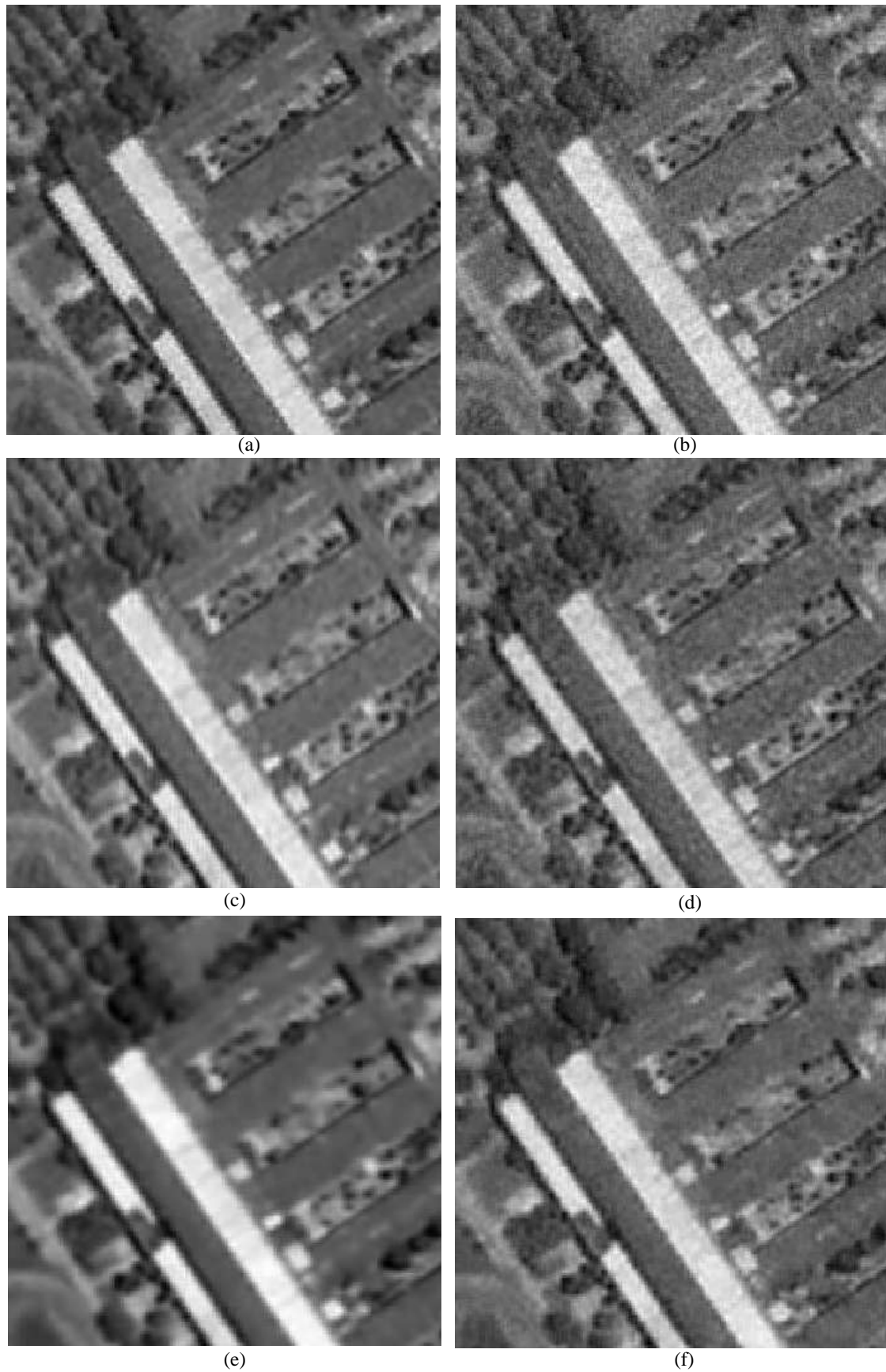


Figura 4.10: Imagens Super-resolução de satélite com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.

Tabela 4.1: Erro médio quadrático das imagens de super-resolução de satélite

Imagem de satélite Método de Filtragem	Variância do ruído		
	0	0,03	0,05
Wiener	0,001827	0,003584	0,004447
Wiener - DWT	0,005545	0,003831	0,004682
Wiener - DWT e <i>denoising</i>	-	0,003375	0,003761
Adaptativa	0,002258	0,003847	0,006157
Adaptativa - DWT	0,004907	0,005162	0,006533
Adaptativa - DWT e <i>denoising</i>	-	0,004130	0,004964
Interpolação bilinear	0,004053	0,005966	0,007088

Tabela 4.2: Grau de Similaridade das imagens de super-resolução de satélite

Imagem de satélite Método de Filtragem	Variância do ruído		
	0	0,03	0,05
Wiener	0,933963	0,815139	0,770873
Wiener - DWT	0,794334	0,792830	0,757264
Wiener - DWT e <i>denoising</i>	-	0,746594	0,719146
Adaptativa	0,882578	0,783632	0,686955
Adaptativa - DWT	0,831166	0,767191	0,692564
Adaptativa - DWT e <i>denoising</i>	-	0,761850	0,701098
Interpolação bilinear	0,722545	0,654608	0,604839

A figura 4.2 contém os resultados obtidos com a imagem original de satélite sem adição de ruído, usando os métodos baseados em filtragem de Wiener. Pode-se observar desta figura os bons resultados alcançados com a filtragem de Wiener em relação ao processo de interpolação. O uso da transformada wavelet neste caso não foi vantajoso, sendo a imagem resultante um pouco menos nítida do que a obtida sem o uso da transformada, resultado confirmado avaliação quantitativa das tabelas 4.1 e 4.2.

As figuras 4.3 e 4.4 mostram os resultados da filtragem de Wiener com a mesma imagem de satélite, mas com ruído de medição de variância $\sigma^2 = 0,03$ e $\sigma^2 = 0,05$, respectivamente. Também é mostrada neste caso a imagem super-resolução obtida utilizando filtragem de Wiener em conjunto com a transformada wavelet e o método de *denoising*. Com este nível de ruído, a filtragem de Wiener continuou superior à interpolação, porém manteve boa parte do ruído aditivo. O uso deste método em conjunto com a transformada wavelet amenizou o ruído visualmente, porém o erro médio quadrático aumentou e teve piora no teste de similaridade; já com a utilização do *denoising* foi obtida uma imagem com ruído e erro reduzidos, mas resultou em imagens ligeiramente desfocadas e com similaridade ainda mais baixa.

Experimentos semelhantes foram realizados com a filtragem adaptativa. A figura

4.5 contém as imagens resultantes para as simulações sem ruído aditivo. Novamente observa-se o resultado superior alcançado com os métodos de filtragem adaptativa em relação ao processo de interpolação. O uso da transformada wavelet, neste caso, não apresentou vantagens porque obteve erro médio quadrático maior que a filtragem adaptativa e interpolação bilinear, contudo no teste de similaridade foi bem superior à interpolação.

A figura 4.6 mostra os resultados do método de filtragem adaptativa para a imagem de satélite com ruído de medição de variância $\sigma^2 = 0,03$. As imagens super-resolução obtidas são pela filtragem adaptativa, filtragem adaptativa em conjunto com a wavelet e utilizando *denoising*. Com este nível de ruído, a filtragem adaptativa foi superior à interpolação, porém manteve o ruído aditivo das imagens LR. E com o uso da transformada wavelet, o ruído continuou ainda muito presente, tendo seu erro quadrático médio um pouco maior e similaridade mais baixa do que sem o uso da wavelet. Já com a utilização do *denoising* foi obtida uma imagem com ruído e erro quadrático médio reduzidos, mas sem melhora na similaridade.

A figura 4.7 mostra os resultados dos métodos de filtragem adaptativa com a imagem de satélite e ruído de medição de variância $\sigma^2 = 0,05$. Com o nível de ruído maior, a filtragem adaptativa se mostrou superior ao método de interpolação, aparentemente, porém manteve o ruído e o aspecto "borrado". No entanto, no resultado deste algoritmo em conjunto com a transformada wavelet, o ruído continuou ainda muito visível, com erro maior, mas com grau de similaridade um pouco melhor. Já com a utilização do *denoising* foi obtida uma imagem com erro médio reduzido e com similaridade superior às demais.

Para facilitar a comparação dos resultados, a figura 4.8 mostra todas as imagens super-resolução obtidos a partir das imagens LR sem adição de ruído. A figura 4.8(a) é a imagem super-resolução obtida pela filtragem de Wiener, a figura 4.8(b) é a imagem super-resolução obtida pela filtragem de Wiener em conjunto com a wavelet, a figura 4.8(c) é a imagem super-resolução obtida pela filtragem adaptativa, a figura 4.8(d) é a imagem super-resolução obtida pela filtragem adaptativa em conjunto com a wavelet. Nota-se que nessa implementação sem ruído, a aplicação de wavelet tornou os resultados piores, tanto visualmente quanto quantitativamente. A filtragem de Wiener foi o método que obteve melhor resultado e com o uso da wavelet, a filtragem adaptativa obteve resultado um pouco melhor.

As figuras 4.9 mostram as imagens super-resolução com ruído de variância $\sigma^2 = 0,03$. Com o uso da transformada wavelet, nos dois casos, as imagens resultantes tiveram erro e similaridade inferiores, e com o uso do *denoising*, o erro foi reduzido mas com similaridades inferiores às demais. Apesar de não ter obtido uma similaridade muito alta, a filtragem adaptativa usando wavelet e *denoising*, foi o que apresentou melhor resultado por análise subjetiva.

As figuras 4.10 mostram as imagens super-resolução com ruído de variância $\sigma^2 = 0,05$. Com o uso da transformada wavelet, nos dois casos, as imagens SR tiveram erro e similaridade inferiores, e com o uso do *denoising*, somente com a filtragem adaptativa obteve-se similaridade superior à filtragem original correspondente. Apesar de não ter obtido uma similaridade muito alta, a filtragem adaptativa usando wavelet e *denoising* foi novamente o método que apresentou melhor resultado por análise subjetiva.

Para a imagem de satélite, os melhores resultados de acordo com o erro médio quadrático e a avaliação do SSIM foram obtidos usando a filtragem de Wiener.

Comparando a análise subjetiva e os valores das avaliações para as imagens super-resolução de satélite, podemos concluir que as avaliações quantitativas foram condizentes em relação às imagens obtidas com pequenas contradições. Portanto, um erro menor ou um SSIM maior pode não se traduzir em um melhor resultado considerando a análise subjetiva.

As figuras 4.11 a 4.19 contém as imagens de texto obtidas pelos algoritmos descritos neste trabalho, enquanto que as tabelas 4.3 e 4.4 comparam quantitativamente esses resultados através do erro médio quadrático e do SSIM. Os valores 0, 0,03 e 0,05 são as variâncias do ruído de medição utilizados, sendo que para o nível 0 (sem ruído), não foi implementado o método de *denoising*.

Filtros
Adaptative
Filtros
Adaptative

(a)

Filtros Filtros
Adaptative Adaptative
Filtros Filtros
Adaptative Adaptative

(b)

(c)

Filtros Filtros
Adaptative Adaptative
Filtros Filtros
Adaptative Adaptative

(d)

(e)

Figura 4.11: Imagens de texto: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.



Figura 4.12: Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.

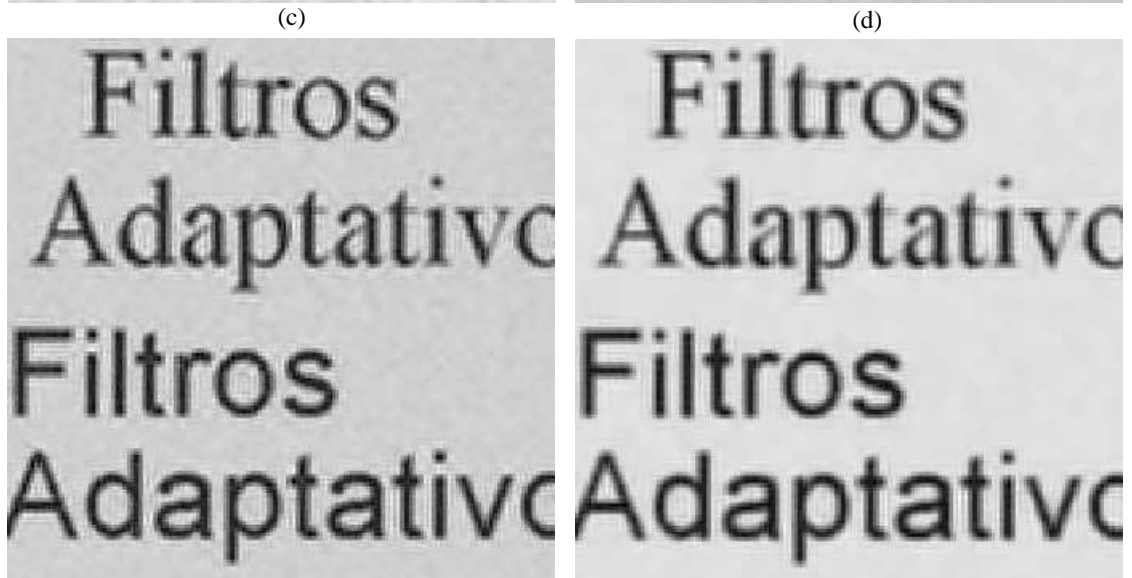
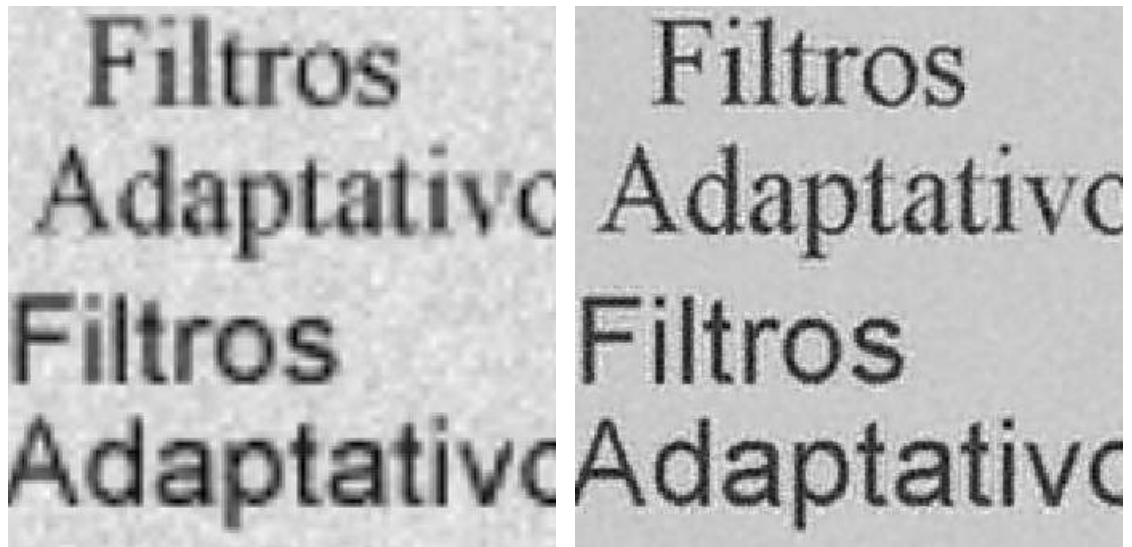
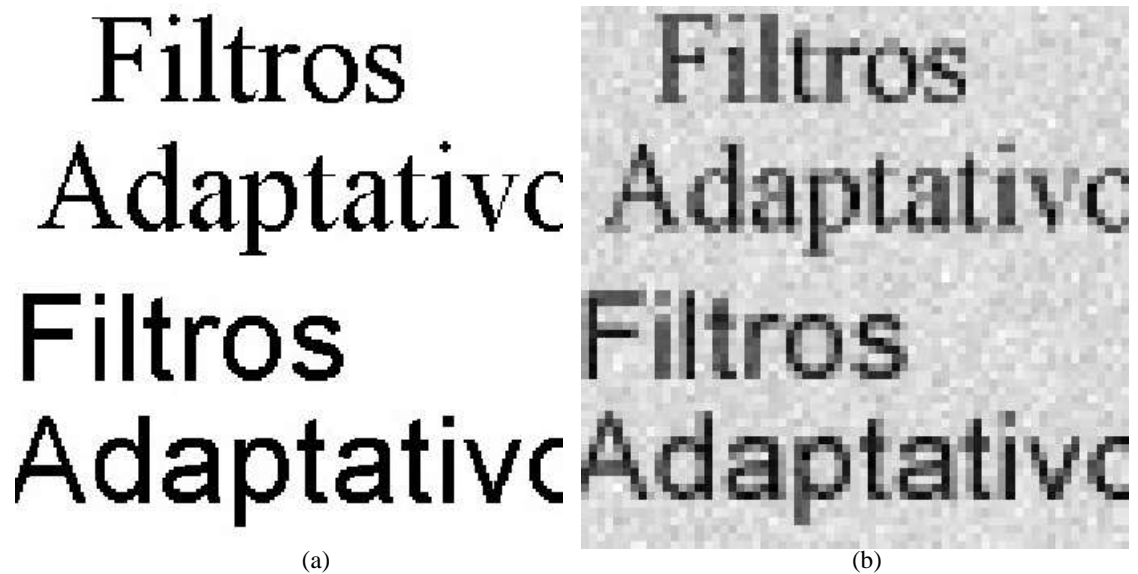


Figura 4.13: Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.

Filtros
Adaptative
Filtros
Adaptative

(a)

Filtros Filtros
Adaptative Adaptative
Filtros Filtros
Adaptative Adaptative

(b)

(c)

Filtros Filtros
Adaptative Adaptative
Filtros Filtros
Adaptative Adaptative

(d)

(e)

Figura 4.14: Imagens de texto: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet.

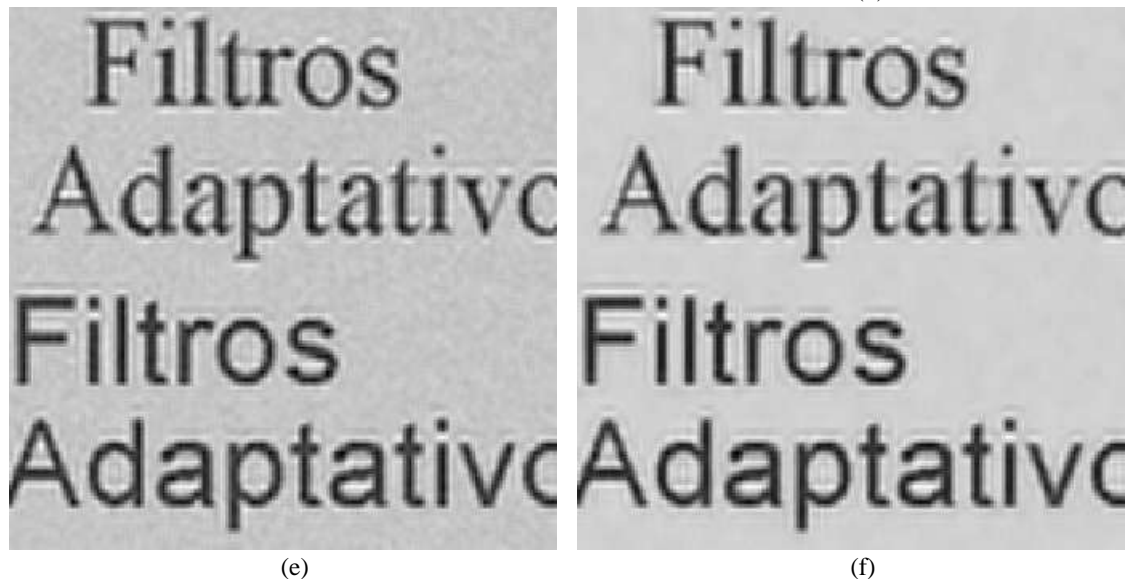
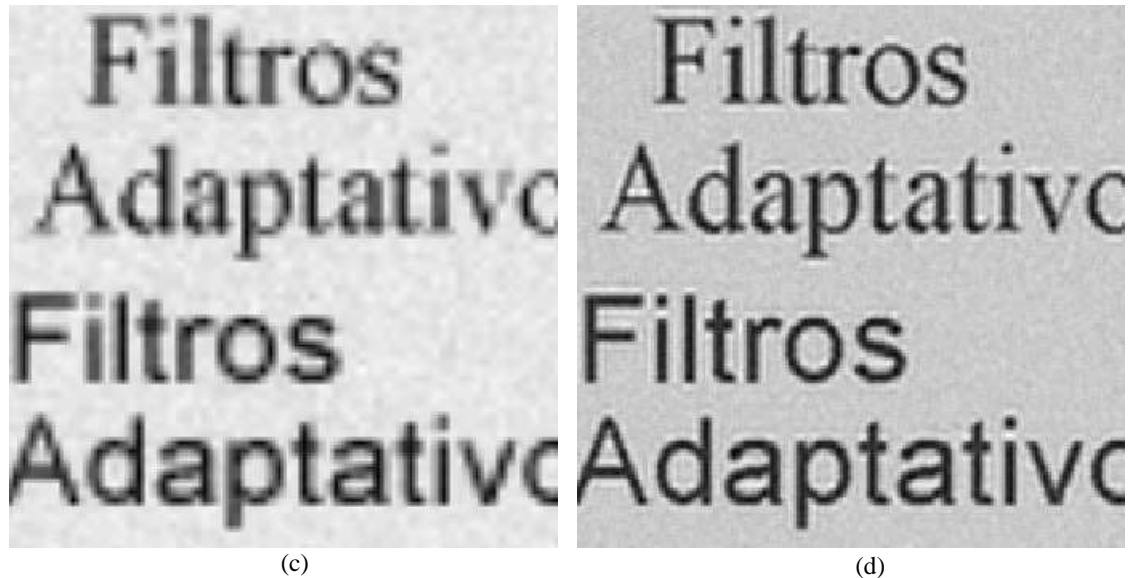
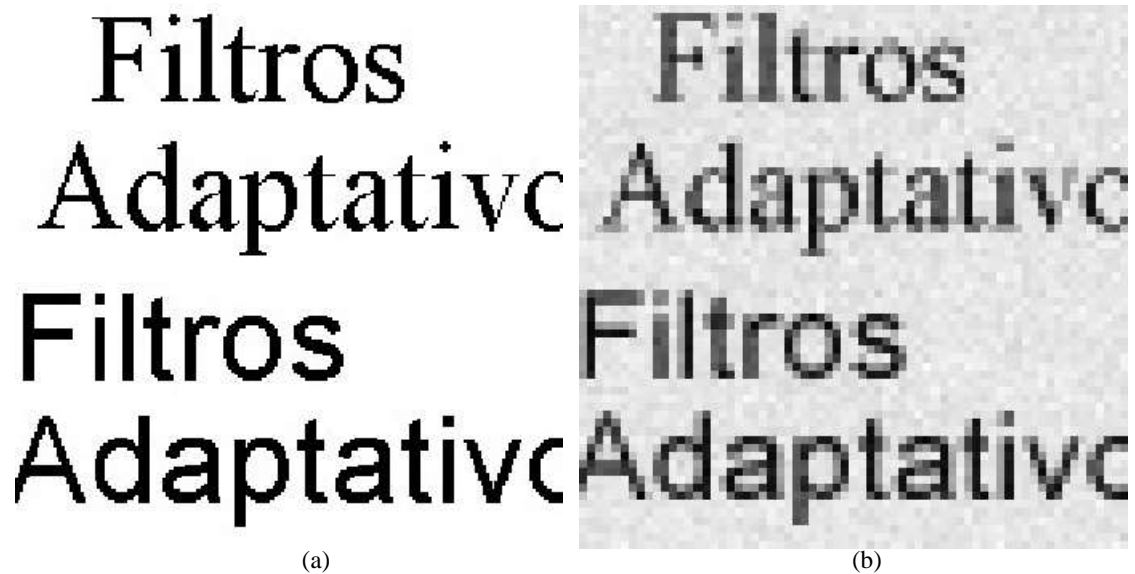


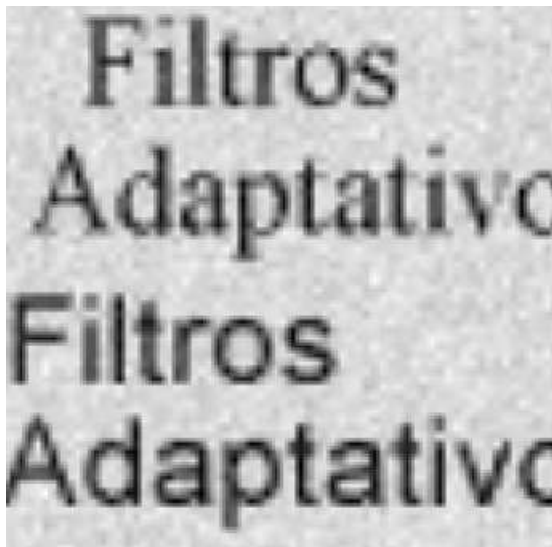
Figura 4.15: Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.

Filtros
Adaptativo
Filtros
Adaptativo

(a)



(b)



(c)



(d)



(e)



(f)

Figura 4.16: Imagens de texto: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.



(a)

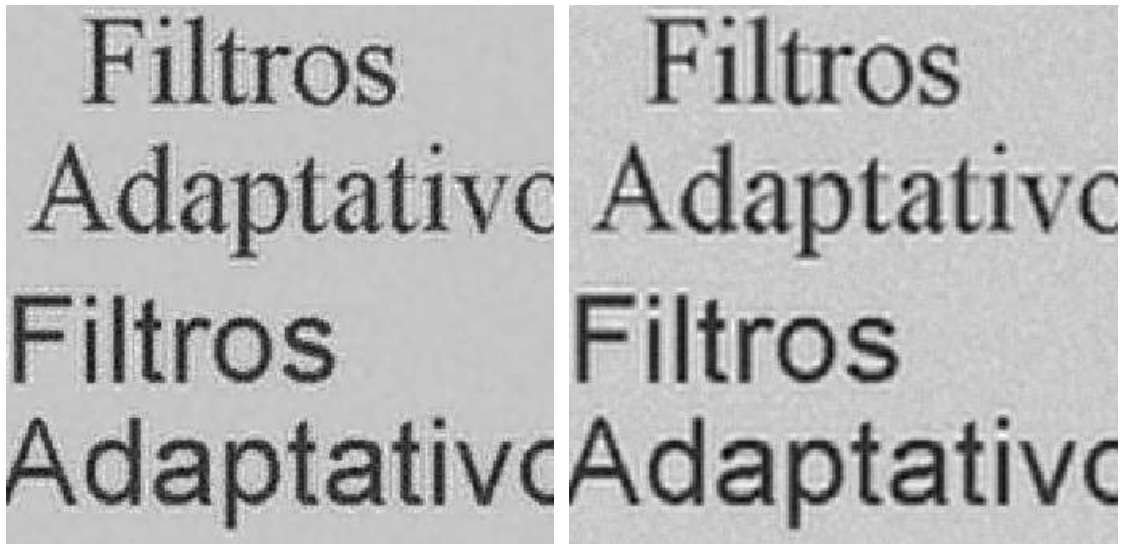
(b)



(c)

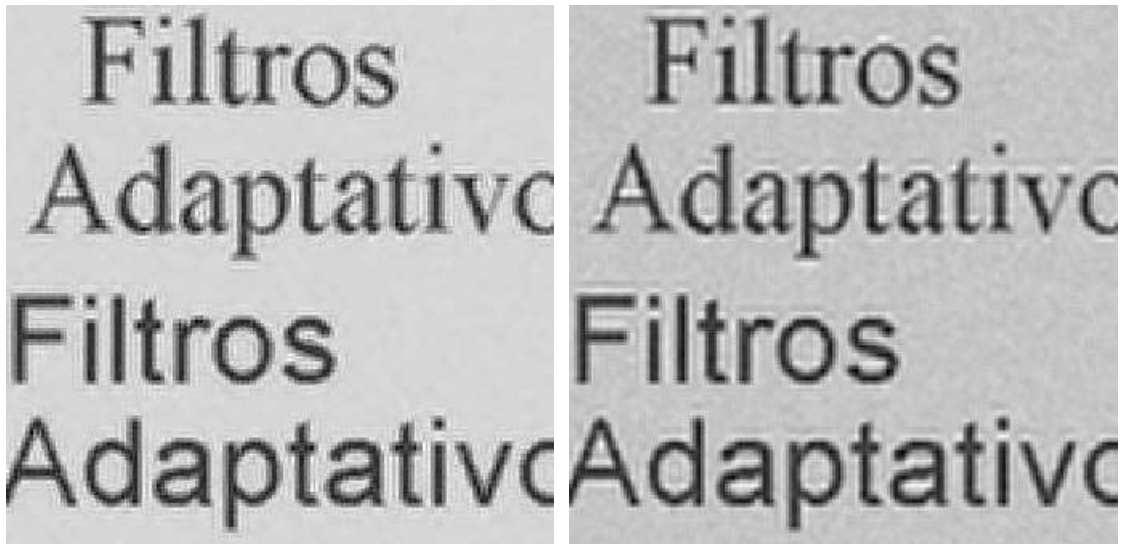
(d)

Figura 4.17: Imagens Super-resolução de texto sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.



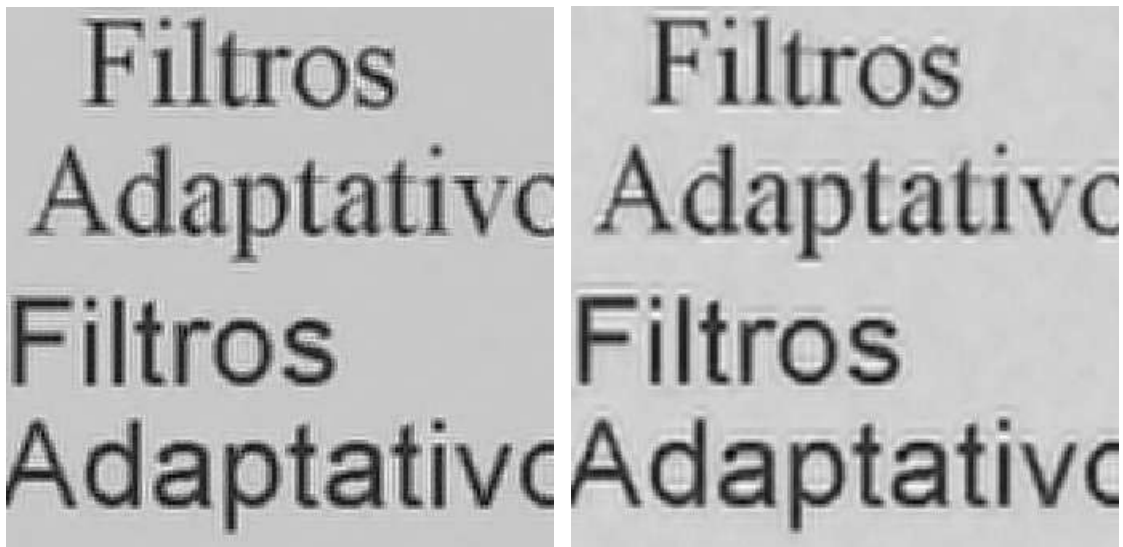
(a)

(b)



(c)

(d)



(e)

(f)

Figura 4.18: Imagens Super-resolução de texto com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.



Figura 4.19: Imagens Super-resolução de texto com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.

Tabela 4.3: Erro médio quadrático das imagens de super-resolução de texto

Imagem de texto	Variância do ruído		
	0	0,03	0,05
Método de Filtragem			
Wiener	0,054630	0,056863	0,056024
Wiener - DWT	0,100667	0,059920	0,056924
Wiener - DWT e <i>denoising</i>	-	0,066840	0,043610
Adaptativa	0,050152	0,051624	0,055508
Adaptativa - DWT	0,060065	0,062211	0,065787
Adaptativa - DWT e <i>denoising</i>	-	0,053524	0,058925
Interpolação bilinear	0,034570	0,043913	0,054436

Tabela 4.4: Grau de Similaridade das imagens de super-resolução de texto

Imagem de texto	Variância do ruído		
	0	0,03	0,05
Método de Filtragem			
Wiener	0,797376	0,779911	0,753465
Wiener - DWT	0,616558	0,707188	0,713446
Wiener - DWT e <i>denoising</i>	-	0,727300	0,756146
Adaptativa	0,798168	0,699679	0,601485
Adaptativa - DWT	0,747728	0,684480	0,609373
Adaptativa - DWT e <i>denoising</i>	-	0,747966	0,703974
Interpolação bilinear	0,753116	0,660938	0,582813

A figura 4.11 contém os resultados obtidos com a imagem original de texto sem adição de ruído, usando os métodos baseados em filtragem de Wiener. Apesar da filtragem em Wiener ter obtido erro médio quadrático maior em relação à interpolação, o índice de similaridade foi superior. Nesse caso, não houve melhora nos resultados ao utilizar a transformada wavelet, mesmo com os efeitos de borda atenuados.¹

A figura 4.12 mostra os resultados da filtragem de Wiener com a mesma imagem de texto, mas com ruído de medição de variância $\sigma^2 = 0,03$. Também é mostrada neste caso a imagem super-resolução obtida utilizando filtragem de Wiener em conjunto com a transformada wavelet e o método de *denoising*. Com este nível de ruído, a filtragem de Wiener obteve erro bem maior que a interpolação, mas com similaridade maior. O uso deste método em conjunto com a transformada wavelet aumentou o erro e piorou a similaridade; já com a utilização do *denoising*, o erro aumentou mais, porém a similaridade melhorou em relação à anterior, sendo este considerado o melhor resultado entre os citados, por análise subjetiva.

¹A variação do nível de cinza no fundo do texto observada nesta figura se deve à normalização utilizada pela função de plotagem usada (comando `imshow(imagem,[])` do Matlab).

A figura 4.13 contém os resultados da filtragem de Wiener na imagem de texto com ruído de medição de variância $\sigma^2 = 0,05$. Para este nível de ruído, a filtragem de Wiener obteve erro um pouco maior que a interpolação, mas com similaridade superior. Usando a transformada wavelet, não houve vantagem, mas nota-se uma redução no ruído, e usando o *denoising* o resultado foi melhor que os demais.

Experimentos semelhantes foram realizados com o método de filtragem adaptativa. A figura 4.14 contém as imagens resultantes para as simulações sem ruído aditivo. A filtragem adaptativa apresentou similaridade superior, mas com erro médio maior em relação à interpolação. O uso da transformada wavelet resultou num aumento do erro e numa similaridade mais baixa em relação à interpolação. Visualmente, a filtragem adaptativa obteve o melhor resultado.

As figuras 4.15 e 4.16 mostram os resultados do método de filtragem adaptativa para a imagem de texto com ruído de medição de variância $\sigma^2 = 0,03$ e $\sigma^2 = 0,05$, respectivamente. As imagens super-resolução obtidas são pela filtragem adaptativa, filtragem adaptativa em conjunto com a wavelet e utilizando *denoising*. Com este nível de ruído, apesar de apresentar erro muito maior que a interpolação, a filtragem adaptativa obteve similaridade superior. Usando a transformada wavelet, não houve melhora, mas com o uso do *denoising*, a similaridade foi superior às demais e a imagem resultante foi considerada a melhor pela análise subjetiva.

Para facilitar a comparação dos resultados, a figura 4.17 mostra todas as imagens super-resolução obtidos a partir das imagens LR sem adição de ruído. A figura 4.17(a) é a imagem super-resolução obtida pela filtragem de Wiener, a figura 4.17(b) é a imagem super-resolução obtida pela filtragem de Wiener em conjunto com a wavelet, a figura 4.17(c) é a imagem super-resolução obtida pela filtragem adaptativa, a figura 4.17(d) é a imagem super-resolução obtida pela filtragem adaptativa em conjunto com a wavelet. Nota-se que nessa implementação sem ruído, a aplicação de wavelet proporcionou uma redução de similaridade, e aumento do erro médio quadrático. Pela análise subjetiva, a filtragem adaptativa obteve melhor resultado, contudo, pela avaliação do SSIM, a melhor imagem super-resolução obtida foi através da filtragem baseada em Wiener.

A figura 4.18 mostra as imagens super-resolução com ruído de variância $\sigma^2 = 0,03$. Comparando a sequência de imagens, nota-se claramente a redução de similaridade com o uso de wavelet, mas obteve-se melhora com a inclusão *denoising*. Pela análise subjetiva, o melhor resultado alcançado foi com a filtragem baseada em Wiener usando transformada wavelet e *denoising*. No entanto, pela avaliação do SSIM, a melhor super-resolução alcançada foi através da filtragem baseada em Wiener.

A figura 4.19 mostra todas as imagens super-resolução com ruído de variância $\sigma^2 = 0,05$. Avaliando-se subjetivamente todos os efeitos descritos acima, concluímos

que o método da filtragem baseada em Wiener em conjunto com a transformada wavelet e *denoising* foi o que apresentou o melhor resultado para esse nível de ruído, em concordância com a avaliação por SSIM.

As tabelas 4.3 e 4.4 mostram o desempenho ruim dos algoritmos de super-resolução para imagens tipo texto embora, visualmente, os resultados sejam superiores ao da interpolação bilinear. Esse fato foi consequência da distorção introduzida pela filtragem e da inserção do ruído principalmente em regiões sem variância de intensidade.

A avaliação por erro médio quadrático para imagem tipo texto, foi influenciada pela distorção causada principalmente pela wavelet. Já na avaliação por SSIM houve também algumas contradições, no entanto esta avaliação foi a mais coerente.

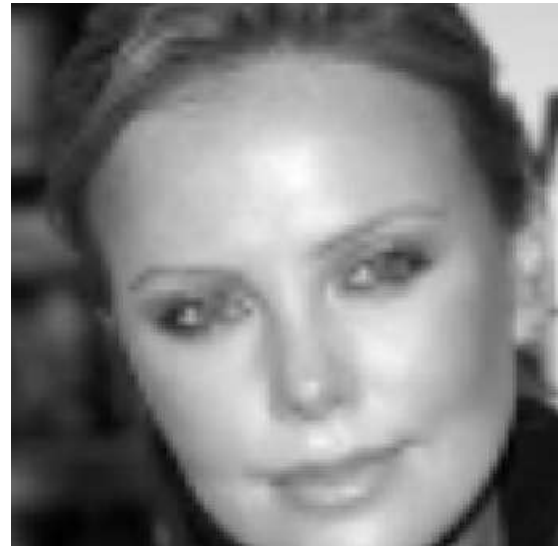
As figuras 4.20 a 4.28 contém as imagens de face obtidas pelos algoritmos descritos neste trabalho, enquanto que as tabelas 4.5 e 4.6 comparam quantitativamente esses resultados através do erro médio quadrático e do SSIM. Os valores 0, 0,03 e 0,05 são as variâncias do ruído de medição utilizados, sendo que para o nível 0 (sem ruído), não foi implementado o método de *denoising*.



(a)



(b)



(c)



(d)



(e)

Figura 4.20: Imagens de face: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet.

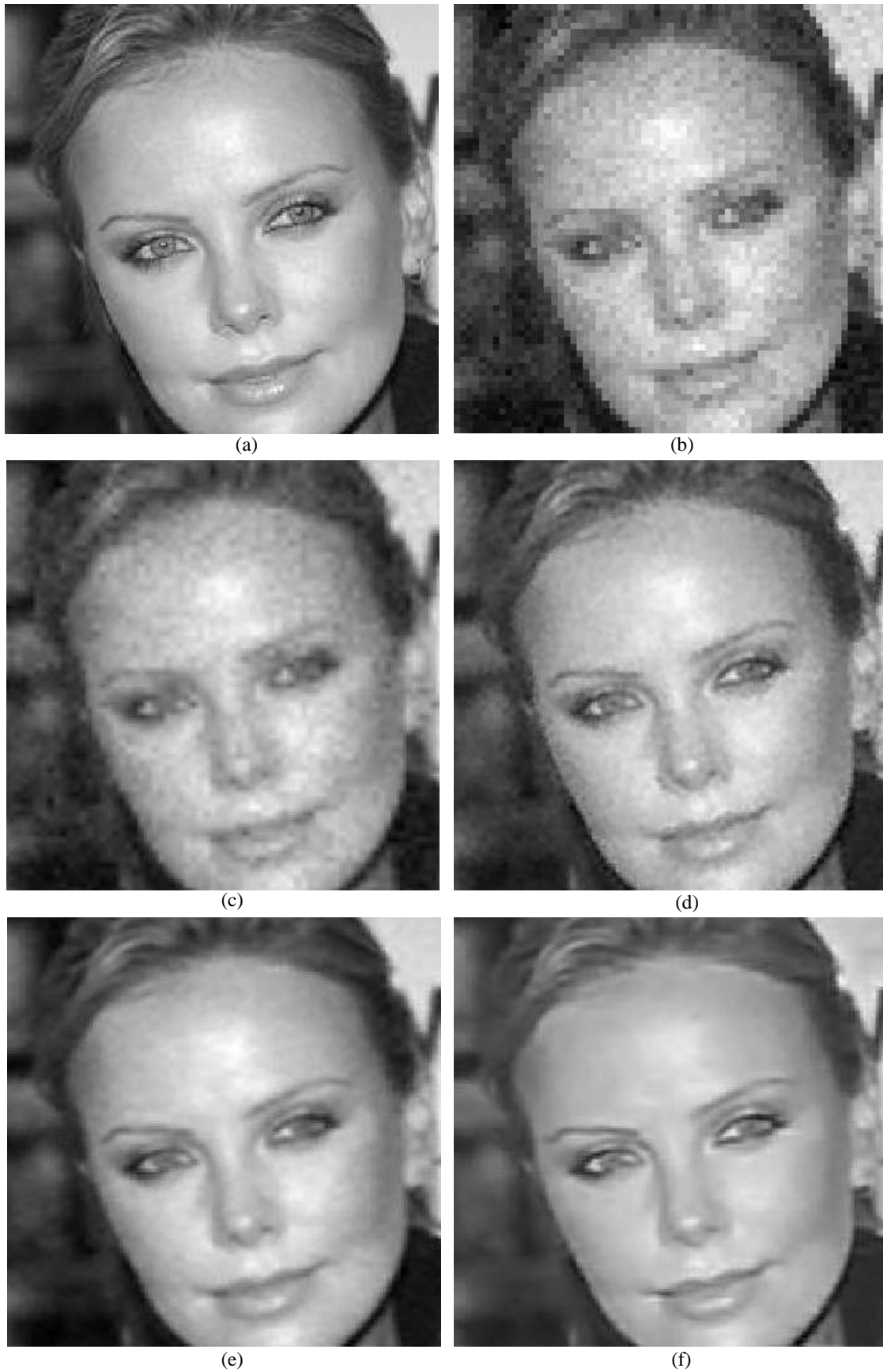


Figura 4.21: Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.

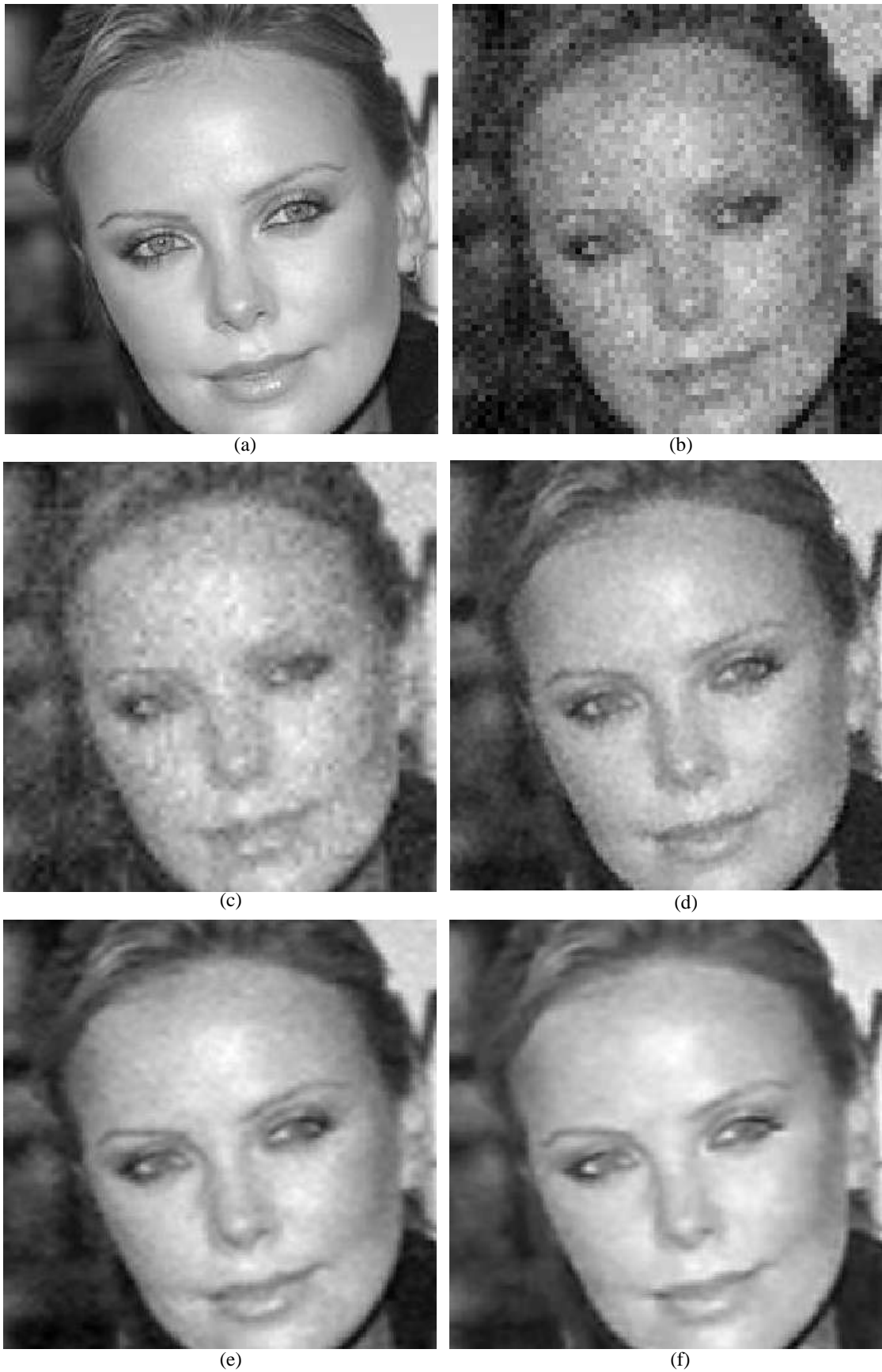


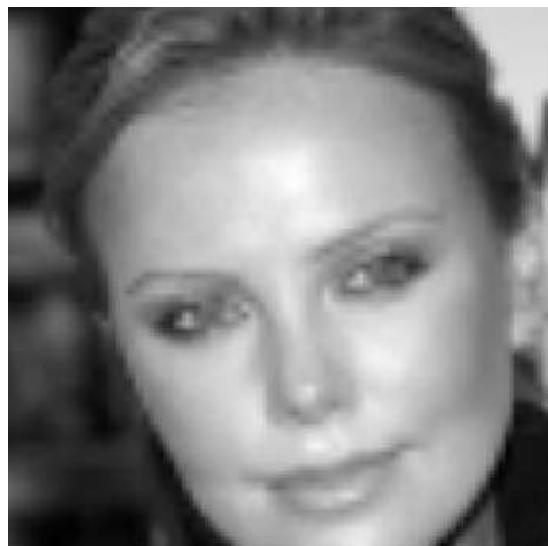
Figura 4.22: Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem de Wiener; (e) SR com filtragem de Wiener e Wavelet; (f) SR com Filtragem de Wiener, Wavelet e *denoising*.



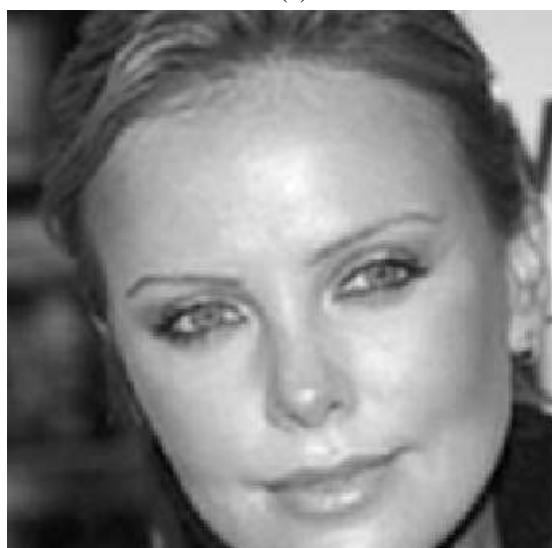
(a)



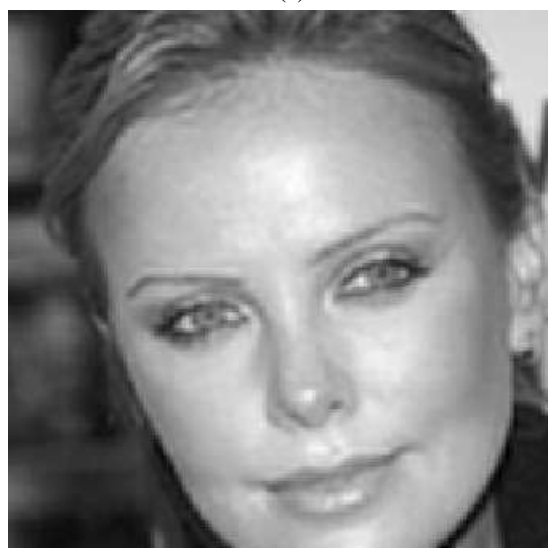
(b)



(c)



(d)



(e)

Figura 4.23: Imagens de face: (a) HR; (b) LR sem ruído; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet.

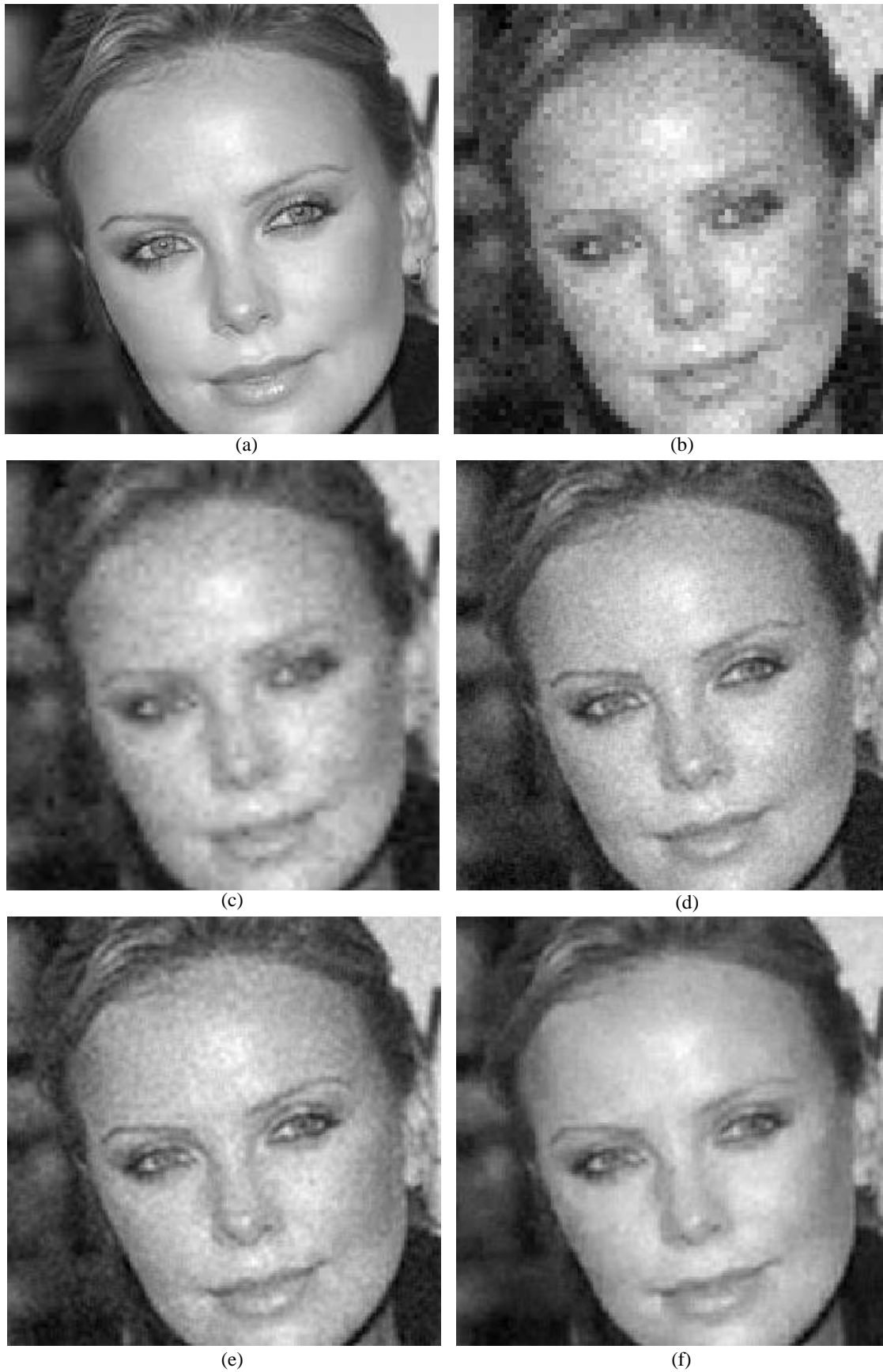


Figura 4.24: Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,03$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.

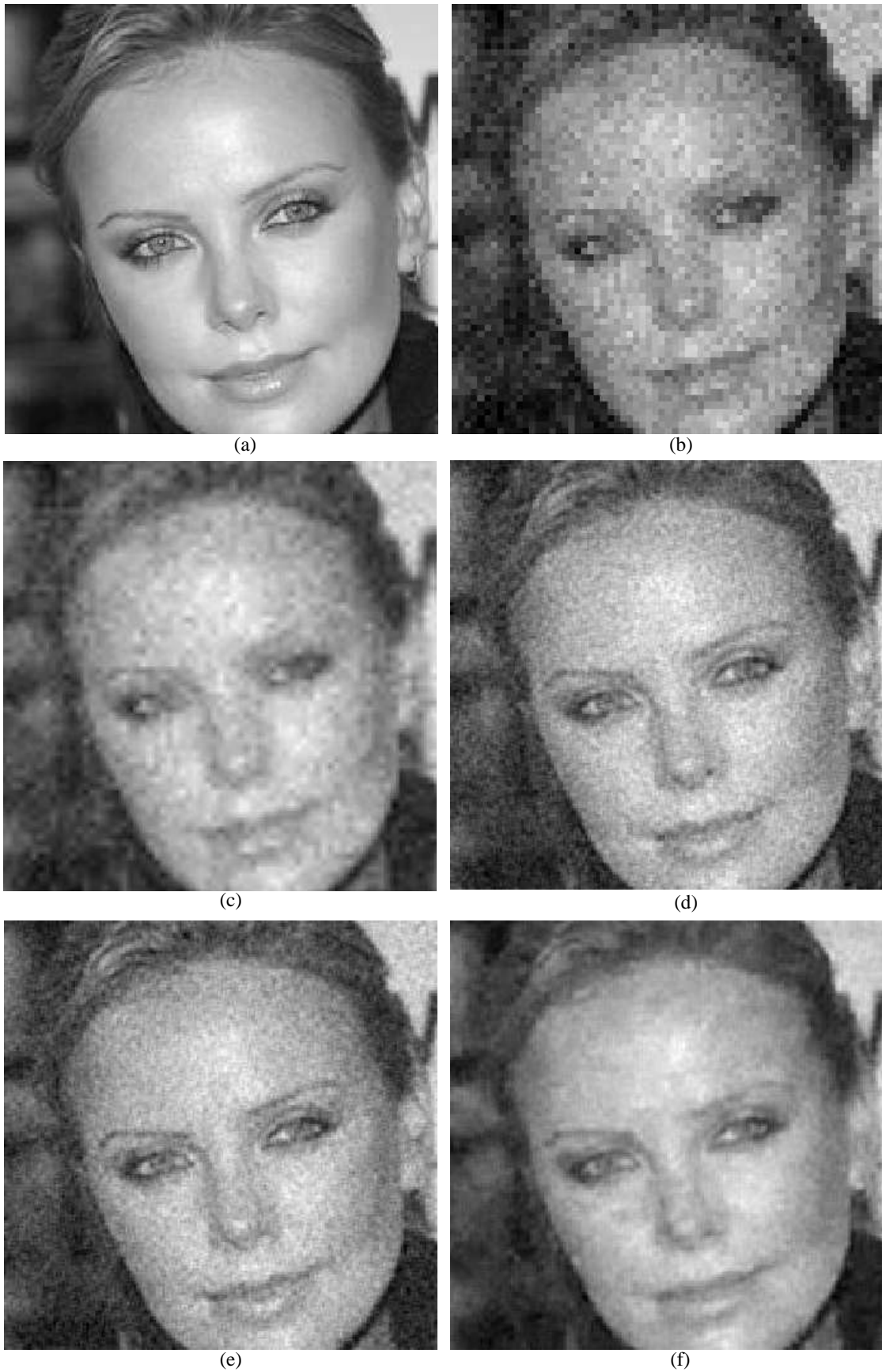


Figura 4.25: Imagens de face: (a) HR; (b) LR com ruído $\sigma^2 = 0,05$; (c) Interpolada; (d) SR com filtragem adaptativa; (e) SR com filtragem adaptativa e Wavelet; (f) SR com Filtragem adaptativa, Wavelet e *denoising*.



Figura 4.26: Imagens Super-resolução de face sem ruído: (a) Filtragem de Wiener; (b) Filtragem de Wiener e Wavelet; (c) Filtragem adaptativa; (d) Filtragem adaptativa e Wavelet.

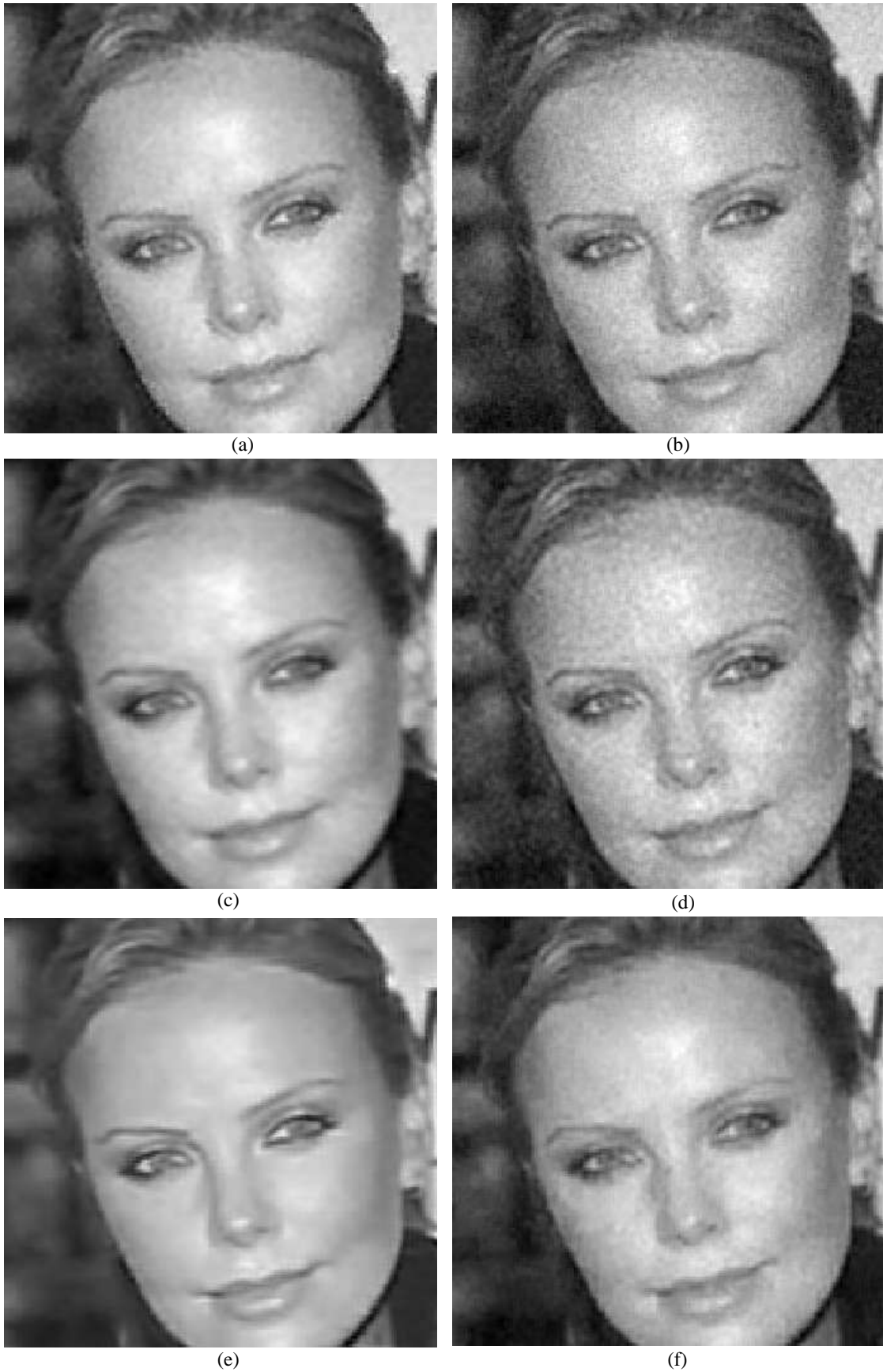


Figura 4.27: Imagens Super-resolução de face com ruído $\sigma^2 = 0,03$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.

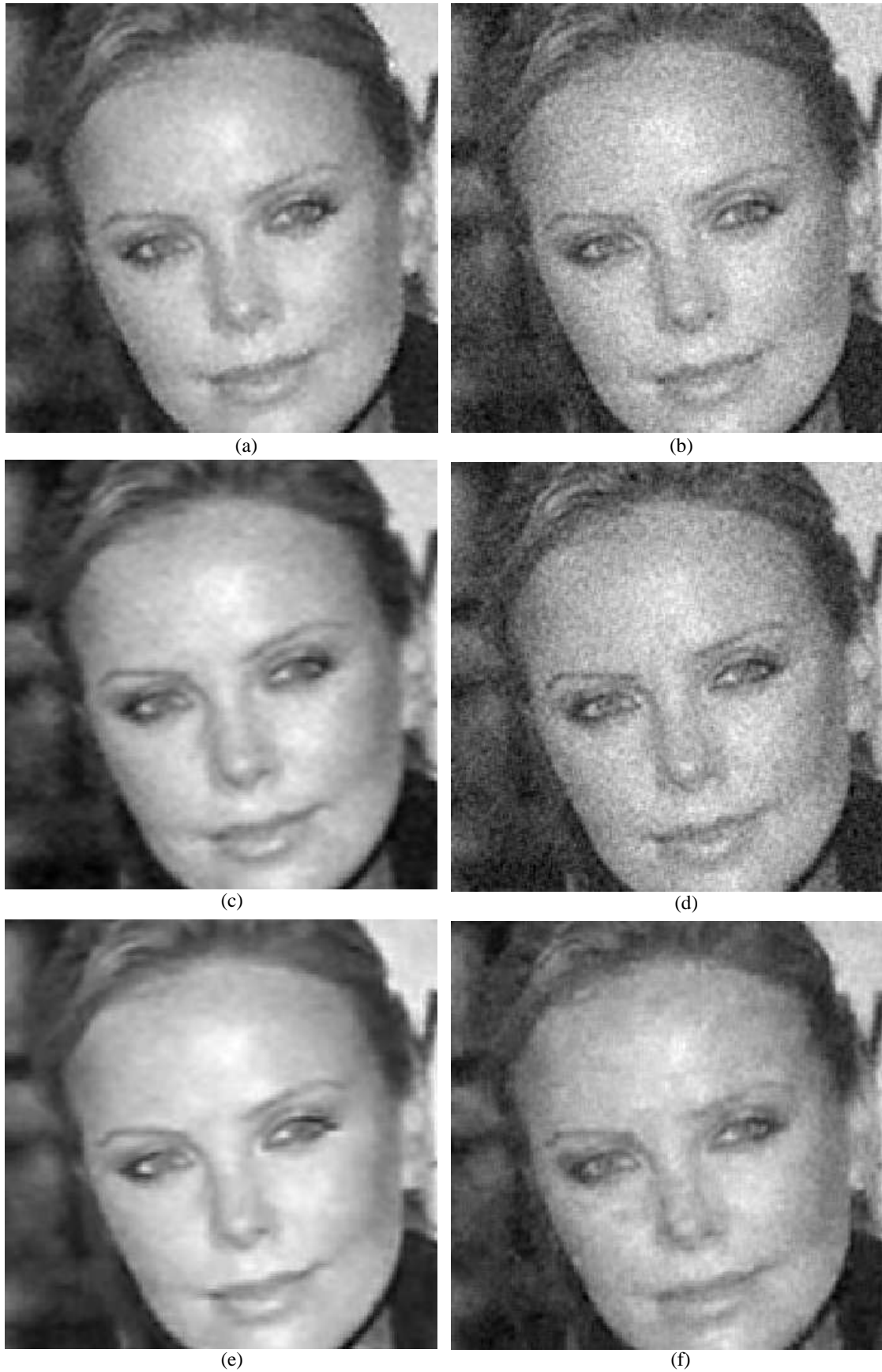


Figura 4.28: Imagens Super-resolução de face com ruído $\sigma^2 = 0,05$: (a) Filtragem de Wiener; (b) Filtragem adaptativa; (c) Filtragem de Wiener e Wavelet; (d) Filtragem adaptativa e Wavelet; (e) Filtragem de Wiener, Wavelet e *denoising*; (f) Filtragem adaptativa, Wavelet e *denoising*.

Tabela 4.5: Erro médio quadrático das imagens de super-resolução de face

Imagem de face	Variância do ruído		
	0	0,03	0,05
Método de Filtragem			
Wiener	0,000172	0,002692	0,001118
Wiener - DWT	0,000386	0,000838	0,000997
Wiener - DWT e <i>denoising</i>	-	0,002134	0,001779
Adaptativa	0,000335	0,001324	0,003370
Adaptativa - DWT	0,000420	0,001365	0,002717
Adaptativa - DWT e <i>denoising</i>	-	0,001044	0,002184
Interpolação bilinear	0,001308	0,001615	0,003233

Tabela 4.6: Grau de Similaridade das imagens de super-resolução de face

Imagem de face	Variância do ruído		
	0	0,03	0,05
Método de Filtragem			
Wiener	0,982375	0,862741	0,794211
Wiener - DWT	0,956471	0,871627	0,833878
Wiener - DWT e <i>denoising</i>	-	0,872563	0,848120
Adaptativa	0,959725	0,721363	0,566410
Adaptativa - DWT	0,945645	0,752167	0,606599
Adaptativa - DWT e <i>denoising</i>	-	0,823139	0,724614
Interpolação bilinear	0,885772	0,757302	0,632307

A figura 4.20 contém os resultados obtidos com a imagem original de uma face (Charlize) sem adição de ruído, usando os métodos baseados em filtragem de Wiener. Pode-se observar desta figura os bons resultados alcançados com a filtragem de Wiener em relação ao processo de interpolação. O uso da transformada wavelet neste caso apresentou resultado semelhante ao obtido sem o uso da transformada, perdendo um pouco a nitidez, sem melhora na similaridade e no erro médio quadrático.

As figuras 4.21 e 4.22 mostram os resultados da filtragem de Wiener com a mesma imagem de face, com ruído de medição de variância $\sigma^2 = 0,03$ e $\sigma^2 = 0,05$. Também é mostrada neste caso a imagem super-resolução obtida utilizando filtragem de Wiener em conjunto com a transformada wavelet e o método de *denoising*. Com este nível de ruído, a filtragem de Wiener continuou superior à interpolação, porém manteve boa parte do ruído aditivo. O uso da filtragem de Wiener em conjunto com a transformada wavelet tivemos uma considerável redução do erro e ruído, e houve aumento da similaridade; já com a utilização do *denoising* foi obtida uma similaridade melhor, mesmo com o aumento do erro em relação a anterior. Este último método teve o resultado melhor na análise subjetiva.

Experimentos semelhantes foram realizados com a filtragem adaptativa. A figura

4.23 contém as imagens resultantes para as simulações sem ruído aditivo. Novamente observa-se o resultado superior alcançado com os métodos de super-resolução, agora usando filtragem adaptativa, em relação ao processo de interpolação. A utilização da transformada wavelet produziu resultado semelhante ao obtido sem a transformada, porém sem melhora na similaridade e no erro médio.

As figuras 4.24 e 4.25 mostram os resultados do método de filtragem adaptativa para a imagem da face com ruído de medição de variância $\sigma^2 = 0,03$ e $\sigma^2 = 0,05$, respectivamente. As imagens super-resolução obtidas são pela filtragem adaptativa, filtragem adaptativa em conjunto com a transformada wavelet e utilizando *denoising*. Com este nível de ruído, a filtragem adaptativa continuou superior à interpolação, porém resultou numa imagem bastante ruidosa. Com o uso da transformada wavelet houve aumento da similaridade, mas o ruído continuou ainda muito presente. Já com a utilização do *denoising*, o ruído foi reduzido, sem muita perda de informação, resultando em uma similaridade foi bem mais elevada.

Para facilitar a comparação dos resultados, a figura 4.26 mostra todas as imagens super-resolução obtidos a partir das imagens LR da face sem adição de ruído. A figura 4.26(a) é a imagem super-resolução obtida pela filtragem de Wiener, a figura 4.26(b) é a imagem super-resolução obtida pela filtragem de Wiener em conjunto com a wavelet, a figura 4.26(c) é a imagem super-resolução obtida pela filtragem adaptativa, a figura 4.26(d) é a imagem super-resolução obtida pela filtragem adaptativa em conjunto com a wavelet. Nota-se que nessa implementação sem ruído, a aplicação de wavelet proporcionou maior definição, porém inseriu pequenos efeitos indesejáveis, decorrente da amplificação das altas frequências. O método de Wiener apresentou resultados superiores aos da filtragem adaptativa.

A figura 4.27 mostra as imagens super-resolução obtidas das imagens LR da face com ruído de variância $\sigma^2 = 0,03$. Comparando a sequência de imagens, nota-se claramente a redução de ruído com o uso da transformada wavelet, e mais ainda com a inclusão do *denoising*. O método da filtragem de Wiener produz imagens com ruído mais suavizado do que a filtragem adaptativa. Portanto, o método de filtragem de Wiener empregando a transformada wavelet e *denoising* foi o que obteve melhor resultado.

A figura 4.28 mostra todas as imagens super-resolução com ruído de variância $\sigma^2 = 0,05$. Comparando a sequência de imagens, nota-se ligeira redução de ruído com o uso da transformada wavelet tanto com o método de Wiener quanto com o da filtragem adaptativa. Com o aumento do ruído, a vantagem do uso do *denoising* fica mais evidente. Novamente, o melhor resultado foi obtido com o método de filtragem de Wiener empregando a transformada wavelet e *denoising*.

Os experimentos com a imagem da face tiveram os melhores resultados nas avaliações quantitativas, sendo estas mais condizentes com a análise subjetiva em

comparação com outros tipos de imagens. Este resultado pode ser justificado pelo fato de haver predomínio de frequências baixas na composição de imagens de faces.

Capítulo 5

Conclusões e Trabalhos Futuros

Neste trabalho, foram estudadas duas técnicas de super-resolução de imagens: a primeira baseada na teoria de filtragem de Wiener e a segunda utilizando algoritmos de filtragem adaptativa. Estas técnicas apresentam baixa complexidade computacional quando comparadas a outras técnicas que empregam sofisticados métodos de otimização.

Em seguida, estas técnicas foram estendidas para o domínio da transformada wavelet, onde as imagens decompostas foram tratadas separadamente e os resultados foram combinados para formar a imagem super-resolução final. O método de *denoising* usando a transformada wavelet foi combinado com os métodos de super-resolução, com o objetivo de reduzir ruído e aumentar a nitidez das imagens resultantes.

As técnicas investigadas foram implementadas e seus resultados foram comparados para imagens com diferentes conteúdos (satélite, texto e face), e razões sinal-ruído. Sem a presença de ruído, o emprego da transformada wavelet não apresentou alteração significativa nos resultados das técnicas de filtragem de Wiener e adaptativa. Com ruído adicionado às imagens de baixa resolução, o uso da wavelet mostrou-se eficaz para redução do ruído. Com o emprego do método de *denoising* com a transformada wavelet, os resultados foram bastante satisfatórios, gerando imagens menos ruidosas.

Nota-se que em alguns casos, as técnicas de comparação por SSIM [25] e erro médio quadrático [24] não estão em concordância pelo fato de usarem métodos diferentes de avaliação.

Em termos de trabalhos futuros, existem muitas linhas que podem ser seguidas. Seria interessante fazer mais experimentos, com aumento da resolução da imagem por diferentes fatores e uso de um número reduzido de imagens, para investigar as limitações dos algoritmos implementados. Nos métodos que empregam a transformada wavelet, utilizamos apenas a transformada de Haar com um nível de decomposição. O uso de funções wavelets mais seletivas ocasionou um efeito

indesejado de *ringing*, o qual poderia ser analisado, e uma solução para redução deste problema poderia ser investigada. O efeito da utilização de mais níveis de decomposição não foi estudado neste trabalho e poderia ser vantajoso em alguns casos. Uma análise da complexidade computacional dos métodos com diferentes parâmetros, tais como número de imagens LR e fator de aumento de resolução, poderia ser realizada. Outras formas de encontrar o limiar de remoção de ruído no método de *denoising* para cada tipo de imagem poderiam ser investigados.

Referências Bibliográficas

- [1] PAPOULIS, A. “Generalized Sampling Expansion”, *IEEE Transactions on Circuits Systems*, v. 24, n. 11, pp. 652–654, 1977.
- [2] TSAI, R. Y., HUANG, T. S. “Multiframe Image Restoration and Registration”, *Advances in Computer Vision and Image Processing*, pp. 317–339, 1984.
- [3] GONZALEZ, R. C., WOODS, R. E. *Digital Image Processing*. Prentice Hall, 2002.
- [4] CLARK, J. J., PALMER, M. R., LAURENCE, P. D. “A Transformation Method fo the Reconstruction of functions Nonuniformly Spaced Samples”, *IEEE Transactions Acoustic, Speech, Signal Processing*, v. 33, pp. 1151–1165, 1985.
- [5] PARK, S., PARK, M., KANG, M. “Super-resolution Image Reconstruction: a Technical Overview”, *IEEE Signal Processing Magazine*, v. 20, n. 3, pp. 21–36, maio 2003.
- [6] UR, H., GROSS, D. “Improved Resolution from Subpixel shifted Pictures”, *Graphical Models and Image Processing*, v. 54, n. 2, pp. 181–186, 1992.
- [7] BROWN, J. L. J. “Multi-Channel Sampling of Low Pass Signals”, *IEEE Transactions on Circuits Systems*, v. 28, pp. 101–106, fev. 1981.
- [8] MALLAT, S. “A Theory of Super-resolution Signal Decomposition: The Wavelet Representation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 11, n. 7, pp. 674–693, 1989.
- [9] WATT, A., POLICARPO, F. *The Computer Image*. Addison-Wesley, 1998.
- [10] NGUYEN, N., MILANFAR, P. “A Wavelet based Interpolation Restoration method for Superresolution”, *Circuits, Systems, and Signal Processing*, v. 19, pp. 321–338, ago. 2000.

- [11] AHUJA, N. A., BOSE, N. K. “Multidimensional Generalized Sampling Theorem for Wavelet Based Image Super-Resolution”, *2006 IEEE International Conference on Image Processing*, pp. 1589–1592, 2006.
- [12] ELAD, M., FEUER, A. “Super-resolution Restoration of an Image Sequence: Adaptive Filtering Approach”, *IEEE Transactions Image Processing*, v. 8, n. 3, pp. 387–395, mar 1999.
- [13] HARDIE, R. “A fast Image Super-resolution Algorithm using an Adaptive Wiener Filter”, *IEEE Transactions Image Processing*, v. 16, n. 12, 2007.
- [14] BECKER, A. “A Review on Image Distortion Measures”, *Fraunhofer - ITWM*, march 2000.
- [15] PEARLMAN, W. A. “A Visual Model and a new Distortion Measure in the Context of Image Processing”, *1978 Journal Optical Society of America*, pp. 374–386, 1978.
- [16] RITTNER, L. *"Identificação e Transformação de Valores Aberrantes como Medida de Confiabilidade do Método das Diferenças para Estimativa de Fluxo Óptico em Sequências de Imagens"*. Tese de Mestrado, Universidade Estadual de Campinas, 2004.
- [17] S.MALLAT. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.
- [18] BURRUS, C. S., GOPINATH, R. A., GUO, H. *Introduction to Wavelets and Wavelets Transforms: A Primer*. Prentice Hall, 1997.
- [19] VAIDYANATHAN, P. P. *Multirate Systems and Filter Banks*. Prentice Hall, 1993.
- [20] DONOHO, D. L. “De-noising by Soft Thresholding”, *IEEE Transactions on Information Theory*, v. 41, n. 3, pp. 613–627, may 1995.
- [21] DONOHO, D. L., JOHNSTONE, I. M. “Ideal Spatial Adaptation via Wavelet Shrinkage”, *Biometrika*, v. 81, pp. 425–455, 1994.
- [22] OLSEN, S. I. “Estimation of Noise in Images: An Evaluation”, .
- [23] TAI, S. C., YANG, S. M. “A Fast Method for Image Noise Estimation Using Laplacian Operator and Adaptive Edge Detection”, .
- [24] ESKICIOGLU, A. M., FISHER, P. S. “Image Quality Measures and their Performance”, *IEEE Transactions on Communications*, v. 43, n. 12, pp. 2959–2965, december 1995.

- [25] WANG, Z., SHEIKH, H. R., SIMONCELLI, E. P. “Image Quality Assessment: From Error Visibility to Structural Similarity”, *IEEE Transactions on Image Processing*, v. 13, n. 4, pp. 600–612, april 2004.
- [26] WANG, Z., BOVIK, A. C., SHEIKH, H. R., et al. “The SSIM Index for Image Quality Assessment”, . Created Feb. 2003; Last updated Nov. 2010, available at <http://www.ece.uwaterloo.ca/z70wang/research/ssim/>.