

UNIVERSIDADE ESTADUAL PAULISTA

“Júlio de Mesquita Filho”

Pós-Graduação em Ciência da Computação

Rafael Gonçalves Pires

Restauração de Imagens Utilizando Projeções em  
Conjuntos Convexos e Algoritmos Evolucionistas

UNESP

2014

Rafael Gonçalves Pires

Restauração de Imagens Utilizando Projeções em  
Conjuntos Convexos e Algoritmos Evolucionistas

Orientador: Prof. Dr. João Paulo Papa

Dissertação de Mestrado elaborada junto ao  
Programa de Pós-Graduação em Ciência da  
Computação – Área de Concentração em  
Computação Científica, como parte dos requisitos  
para a obtenção do título de Mestre em Ciência da  
Computação

UNESP

2014

Pires, Rafael Gonçalves.

Restauração de imagens utilizando projeções em conjuntos convexos e algoritmos evolucionistas / Rafael Gonçalves Pires. -- São José do Rio Preto, 2014

59 f. : il., fórmulas, gráfs., tabs

Orientador: João Paulo Papa

Dissertação (mestrado) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Computação. 2. Processamento de imagens - Técnicas digitais. 3. Reconstrução de imagens. 4. Conjuntos convexos. 5. Otimização matemática. I. Papa, João Paulo. II. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 518.72:76

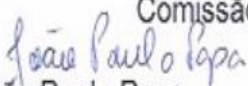
Ficha catalográfica elaborada pela Biblioteca do IBILCE  
UNESP - Câmpus de São José do Rio Preto


Rafael Gonçalves Pires

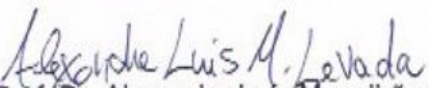
Restauração de Imagens Utilizando Projeções em Conjuntos  
Convexos e Algoritmos Evolucionistas

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, Área de Concentração – Computação Científica, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de São José do Rio Preto.

Comissão Examinadora

  
Prof. Dr. João Paulo Papa  
UNESP – Bauru  
Orientador

  
Prof. Dr. Antonio Carlos Sementille  
UNESP – Bauru

  
Prof. Dr. Alexandre Luis Magalhães Levada  
UFSCar – São Carlos

São José do Rio Preto  
24 de Fevereiro de 2014

Dedico este trabalho

Dedico este trabalho a Deus, meus Avós, Família, Namorada e a todos Pesquisadores.

# Agradecimentos

A Deus pelo dom da vida, e ao meu anjo da guarda meu zeloso guardador.

Agradeço a minha família pelo apoio e Amor incondicional.

Ao meu Orientador João Paulo Papa, pela oportunidade, paciência, amizade, humildade e compreensão.

A todos os Professores responsáveis pela minha formação.

Ao Professor Alexandre Luís Magalhães Levada, pelos ensinamentos, humildade e boa vontade.

Aos meus companheiros de pesquisa, pela ajuda, conversas e idéias em especial a Luís Augusto Martins.

A grande alegria da minha vida, Ariane, meu amor, que sempre esteve ao meu lado em todos os momentos.

A todos os meus Avós Aparecida, Manoel, Tereza, Mauro e Familiares.

Muito Obrigado a todos!

## **Oração a Nossa Senhora**

Ó Maria, Mãe do belo amor, do temor, do conhecimento e da santa esperança, por cuja piedosa intercessão muitos, embora rudes no entendimento, fazem progressos admiráveis na ciência e na piedade, escolho-Vos como protetora e patrona dos meus estudos. Humildemente Vos imploro, pelas entranhas de vossa maternal piedade, e principalmente por aquela Sabedoria Eterna que Se dignou assumir em Vós nossa carne e exaltar-Vos, na luz celestial, acima de todos os Santos, obtende-me, por vossa intercessão, a graça do Espírito Santo, de maneira que eu possa penetrar com o entendimento, reter na memória, exprimir pela vida e por palavras, ensinar aos outros todas as coisas que trazem honra a Vós e a vosso Filho, como também proveito para minha felicidade eterna e dos demais homens. Amém.

Composta por São Tomás de Aquino para ser rezada antes dos estudos.

## Resumo

O processo de restauração de imagens tem como objetivo melhorar as imagens corrompidas por ruídos e borramentos. Técnicas iterativas podem controlar melhor o algoritmo de restauração a fim de restaurar detalhes em regiões borradas sem aumentar o ruído. Técnicas baseadas em Projeção em Conjuntos Convexos (*Projections onto Convex Sets* - POCS) tem sido utilizadas no contexto de restauração de imagens, projetando a solução em um hiperespaço até que algum critério de convergência seja encontrado. O resultado esperado é uma imagem melhor ao final de um número desconhecido de projeções. O número de conjuntos convexos e suas combinações permitem construir vários algoritmos de restauração de imagens baseados em POCS. O presente trabalho utiliza dois conjuntos convexos: Row-Action Projections (RAP) e Limited Amplitude (LA). O algoritmo RAP possui um parâmetro de relaxação  $\lambda$  que depende das características da imagem que será restaurada. Assim, valores errados de  $\lambda$  podem conduzir a uma pobre restauração. Propomos achar o valor de  $\lambda$  modelando o problema como de otimização e utilizando diferentes técnicas evolucionistas. Além disso, é possível utilizar os parâmetros de restauração aprendidas sobre uma imagem e empregá-los para uma outra imagem. Palavras-chave: *Restauração de Imagens. Otimização. Projeção em Conjuntos Convexos.*



## Abstract

*The process of image restoration aims to enhance images corrupted by noise and blurred. Iterative techniques can better control the restoration algorithm in order to restore blurred regions in details without increasing noise. Techniques based on Projection Sets in Convex (Projections onto Convex Sets - POCS) have been used in the context of image restoration by projecting the solution in a hyperspace until some convergence criterion is met. The expected result is a better picture at the end of an unknown number of projections. The number of convex sets and its combinations allow you to build several image restoration algorithms based on POCS. This study uses two convex sets: Row Action Projections (RAP) and Limited Amplitude (LA). The RAP algorithm has a relaxation parameter  $\lambda$  depends on the characteristics of the image that will be restored. Thus, erroneous values of  $\lambda$  can lead to a poor restoration. We propose to find the value of  $\lambda$  as the problem of modeling and optimization using different evolutionary techniques. Furthermore, is possible to use the parameters learned in restoring an image, and use them to another image.*

*Keywords: Image restoration. Optimization. Rows Actions Projections.*

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Restauração de Imagens</b>	<b>3</b>
2.1	Modelo de Degradação . . . . .	4
2.2	Filtragem Inversa . . . . .	5
2.3	Filtro de Wiener . . . . .	6
<b>3</b>	<b>Projeções em Conjuntos Convexos (POCS)</b>	<b>8</b>
3.1	Conjunto de Restrição de Amplitude Limitada . . . . .	10
<b>4</b>	<b>Algoritmos Evolucionistas</b>	<b>11</b>
4.1	Otimização por Enxame de Partículas - PSO . . . . .	11
4.2	Busca Harmônica - HS . . . . .	13
4.2.1	Etapa 1 - O Problema da Otimização e os Parâmetros do Algoritmo . . . . .	13
4.2.2	Etapa 2 - Memória Harmônica . . . . .	14
4.2.3	Etapa 3 - Gerando uma Nova Harmonia . . . . .	14
4.2.4	Etapa 4 - Atualizando a Memória Harmônica . . . . .	15
4.2.5	Etapa 5 - Critério de Parada . . . . .	15
4.3	Global-best Harmony Search - GHS . . . . .	15
4.4	Self-adaptative Global Best Harmony Search - SGHS . . . . .	16
4.5	Improved Harmony Search - IHS . . . . .	17
4.6	Novel Global Harmony Search - NGHS . . . . .	17
4.7	Algoritmo de Busca Gravitacional - GSA . . . . .	18
4.8	Algoritmo dos Vagalumes - FFA . . . . .	20
4.9	Algoritmo dos Morcegos - BA . . . . .	20
<b>5</b>	<b>Metodologia</b>	<b>23</b>
<b>6</b>	<b>Resultados Experimentais</b>	<b>26</b>

6.1	Experimentos Lena . . . . .	31
6.2	Experimentos Cameraman . . . . .	36
6.3	Aprendendo o valor de $\lambda$ - Experimento 1 . . . . .	41
6.4	Experimentos Sensoriamento Remoto - Imagem 1 . . . . .	44
6.5	Experimentos Sensoriamento Remoto - Imagem 2 . . . . .	49
6.6	Aprendendo o valor de $\lambda$ - Experimento 2 . . . . .	54
<b>7</b>	<b>Conclusões</b>	<b>57</b>

# Lista de Figuras

2.1	Modelo do processo de degradação da imagem. Extraído de: [1]. . . . .	5
3.1	Interpretação geométrica RAP. Extraído de: [2] . . . . .	9
4.1	Interpretação geométrica do PSO. Extraído de: [3] . . . . .	12
5.1	Modelo proposto para restauração de imagens. . . . .	25
6.1	Imagem Lena com $256 \times 256$ e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano. . .	27
6.2	Imagem Cameraman com $256 \times 256$ e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano. . .	28
6.3	Imagem Sensoriamento remoto 1 com $256 \times 256$ e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano. . . . .	29
6.4	Imagem Sensoriamento remoto 2 com $256 \times 256$ e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano. . . . .	30
6.5	Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI. . . . .	31
6.6	Imagens restauradas da Lena considerando os ótimos $\lambda$ encontrados e ISNR: (a) FFA para BG, (b) FFA para TA (c) FFA para BG+R e (d) FFA para TA+R. . .	34
6.7	Imagens restauradas da Lena considerando os ótimos $\lambda$ encontrados e UIQI: (a) FFA para BG, (b) PSO para TA (c) FFA para BG+R e (d) FFA para TA+R. . . .	35
6.8	Busca exaustiva utilizando o algoritmo proposto para as métricas de qualidade de imagem (a) ISNR e (b) UIQI. . . . .	36

6.9	Imagens restauradas do Cameraman considerando os ótimos $\lambda$ encontrados e ISNR: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) PSO para TA+R. . . . .	39
6.10	Imagens restauradas do Cameraman considerando os ótimos $\lambda$ encontrados e UIQI: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	40
6.11	Imagens restauradas do Cameraman usando os melhores $\lambda$ obtidos nas imagens da Lena considerando a métrica ISNR: (a) FFA para BG (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	42
6.12	Imagens restauradas do Cameraman usando os melhores $\lambda$ obtidos nas imagens da Lena considerando a métrica UIQI: (a) FFA para BG, (b) PSO para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	43
6.13	Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI . . . . .	44
6.14	Imagens restauradas Sensoriamento Remoto 1 usando os ótimos $\lambda$ encontrados (ISNR): (a) FFA para BG, (b) FFA para TA, (c) PSO para BG+R e (d) FFA para TA+R. . . . .	47
6.15	Imagens restauradas Sensoriamento Remoto 1 usando os ótimos $\lambda$ encontrados (UIQI): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	48
6.16	Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI . . . . .	49
6.17	Imagens restauradas Sensoriamento remoto 2 usando os ótimos $\lambda$ encontrados (ISNR): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	52
6.18	Imagens restauradas Sensoriamento Remoto 2 usando os ótimos $\lambda$ encontrados (UIQI): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	53
6.19	Imagens restauradas sensoriamento Remoto 2 usando os melhores $\lambda$ obtidos no experimento com as imagens sensoriamento Remoto 1 considerando a métrica ISNR (a) FFA para BG (b) FFA para TA, (c) PSO para BG+R e (d) FFA para TA+R. . . . .	55

6.20	Imagens restauradas sensoriamento Remoto 2 usando os melhores $\lambda$ obtidos no experimento com as imagens sensoriamento Remoto 1 considerando a métrica UIQI: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R. . . . .	56
------	--	----

# Lista de Tabelas

6.1	Parâmetros utilizados para cada técnica de otimização: o número de agentes utilizados para todas as técnicas é 50. Os parâmetros do PSO estão de acordo com Papa et al. [3]. . . . .	26
6.2	Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	32
6.3	Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	32
6.4	Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	33
6.5	Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	33
6.6	Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	37
6.7	Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	37
6.8	Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	37
6.9	Resultados de restauração para imagens Cameraman considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parênteses são os de $\lambda$ . . . . .	38
6.10	Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para BG (os valores em parênteses são os de $\lambda$ ). . . . .	41
6.11	Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para BG+R (os valores em parênteses são os de $\lambda$ ). . . . .	41

6.12	Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para TA (os valores em parenteses são os de $\lambda$ ). . . . .	41
6.13	Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para TA+R (os valores em parenteses são os de $\lambda$ ). . . . .	42
6.14	Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	45
6.15	Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	45
6.16	Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	45
6.17	Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	46
6.18	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	50
6.19	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	50
6.20	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	50
6.21	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de $\lambda$ . . . . .	51
6.22	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG (os valores entre parenteses são os de $\lambda$ ). . . . .	54
6.23	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG+R (os valores entre parenteses são os de $\lambda$ ). . . . .	54



6.24	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA (os valores entre parênteses são os de $\lambda$ ). . . . .	54
6.25	Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA+R (os valores entre parênteses são os de $\lambda$ ). . . . .	54

## Lista de Abreviaturas e Siglas

AL	<i>Amplitude Limitada</i>
ART	<i>Algebraic Reconstruction Technique</i>
BA	<i>Bat Algorithm</i>
BG	<i>Borramento Gaussiano</i>
BG+R	<i>Borramento Gaussiano com ruído</i>
FFA	<i>Firefly Algorithm</i>
GHS	<i>Global-best Harmony Search</i>
GSA	<i>Gravitational Search Algorithm</i>
HMCR	<i>Harmony Memory Considering Rate</i>
HMS	<i>Harmony Memory Size</i>
HS	<i>Harmony Search</i>
IHS	<i>Improved Harmony Search</i>
ISNR	<i>Improvement Signal to Noise Ratio</i>
MH	<i>Memória Harmônica</i>
NGHS	<i>Novel Global Harmony Search</i>
PAR	<i>Pitch Adjusting Rate</i>
POCS	<i>Projections onto convex sets</i>
PSF	<i>Point Spread Function</i>
PSO	<i>Particle Swarm Optimization</i>
RAP	<i>Rows Actions Projections</i>
SGHS	<i>Self-adaptative Global Best Harmony Search</i>
TA	<i>Turbulência Atmosférica</i>
TA+R	<i>Turbulência Atmosférica com ruído</i>
UIQI	<i>Universal Image Quality Index</i>

# 1 Introdução

Processamento de imagens é uma área que tem recebido considerável atenção nos últimos anos, como resultado da evolução da tecnologia de computação digital. Ao mesmo tempo, a área tem motivado o desenvolvimento de novas técnicas matemáticas necessárias para lidar com sinais bidimensionais. Uma das primeiras aplicações das imagens digitais ocorreu na indústria de jornais, onde as imagens eram enviadas por meio de um cabo, submarino de Londres para Nova Iorque [1]. A introdução de um sistema Bartlane para transmissão de dados via cabo por volta de 1920, reduziu o tempo exigido de mais de uma semana para menos de três horas no transporte de imagens através do oceano Atlântico. Um equipamento especializado de impressão codificava as imagens para transmissão, as quais eram então reconstruídas no terminal receptor.

Da década de 1960 até os dias atuais, a área de processamentos digitais cresceu rapidamente, ampliando ainda mais seus domínios de aplicação. Em medicina, por exemplo, procedimentos computacionais melhoram o contraste ou codificam os níveis de intensidade em cores, de modo a facilitar a interpretação de imagens de raios-X e outras imagens biomédicas. Geógrafos utilizam técnicas idênticas ou similares para estudar padrões de poluição em imagens aéreas e de satélites. Na arqueologia, métodos de processamento de imagens tem restaurado com sucesso imagens fotográficas borradas, que eram os únicos registros disponíveis de artefatos raros que foram perdidos ou danificados após serem fotografados. Aplicações bem-sucedidas dos conceitos de processamento de imagens podem ser encontradas em astronomia, biologia, medicina nuclear, aplicação da lei (segurança pública), defesa e indústria. Existem também imagens adquiridas por satélite, as quais são úteis no rastreamento de recursos naturais, mapeamento geográfico, monitoramento do tempo, controle de incêndios e inundações e muitas outras aplicações [4].

As imagens de sensores digitais e fotografias apresentam a característica de agrupar um elevado número de informações, as quais necessitam ser processadas para melhorar a qualidade radiométrica e geométrica dos dados. As distorções radiométricas são causadas pelo borramento dos detalhes, listras e manchas nas imagens. As degradações na imagem diminuem a precisão da informação, reduzindo assim, a utilidade dos dados. Portanto, antes que os dados das imagens sejam utilizados, é necessário que estas passem por um estágio de pré-processamento para que sejam corrigidas geométrica e radiometricamente. Dentre as técnicas de correção radiométrica encontra-se a restauração de imagens. Esse método visa a reconstrução ou recuperação da imagem que foi degradada utilizando um conhecimento *a priori* do fenômeno de degradação. A ideia é, através da modelagem da degradação, aplicar o processo inverso para obter uma aproximação da imagem original [1].

Numerosos métodos de restauração de imagens tem sido desenvolvidos para várias aplicações, dentre eles o filtro inverso, filtro de Wiener, técnicas de regularização e métodos de programação linear [5]. Outra abordagem que tem sido amplamente referenciada é a que utiliza

métodos de projeção [5]. Dentre elas, pode-se destacar o método de Projeções em Conjuntos Convexos (*Projections onto Convex Sets* - POCS), também conhecido como método de projeções convexas, o qual utiliza informações *a priori* das imagens na forma de conjuntos de restrição a fim de tentar solucionar o problema [5]. As técnicas mais usadas de projeção são *Rows-Action Projections* (RAP) e *Simultaneous Iterative Reconstruction Technique* (SIRT) [6].

Dado que o problema de restauração pode ser modelado como um sistema linear, segundo [7] provavelmente o problema mais importante na matemática é o da resolução de um sistema de equações lineares. Mais de 75% de todos os problemas matemáticos encontrados em aplicações científicas e industriais envolvem a resolução de um sistema linear em algum estágio. Usando métodos modernos é frequentemente possível reduzir problemas sofisticados a um simples sistema de equações lineares. Os sistemas lineares aparecem em áreas como negócios, economia, sociologia, ecologia, demografia, genética, engenharia e física

Diferentes abordagens podem ser empregadas para resolver o problema de sistemas lineares. O método RAP lida com essa tarefa aplicando projeções ortogonais iterativas entre os hiperplanos definidos no modelo de restauração de imagem. Inicialmente foi desenvolvido por Kaczmarz em 1937 e posteriormente sua versão bidimensional por [6]. O funcionamento do RAP é semelhante a um processo de convolução, mas extremamente dependente de um parâmetro de relaxação  $\lambda$  que controla a projeção sobre os hiperplanos. Altos ou baixos valores deste parâmetro podem tornar a restauração ineficaz, sendo que a escolha ideal de  $\lambda$  depende das características do ruído e borramento.

Para encontrar o parâmetro de relaxação  $\lambda$ , Papa et al. [3] modelaram essa tarefa como um problema de otimização em que a função de ajuste é dada por algumas métricas de qualidade da imagem, tais como *Improvement Signal to Noise Ratio* (ISNR) e *Universal Image Quality Index* (UIQI). Os autores empregaram a técnica evolucionista Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO), a qual demonstrou grande habilidade em encontrar o parâmetro de relaxação  $\lambda$ .

Nesse sentido, o presente projeto de pesquisa visa implementar e viabilizar diferentes técnicas evolucionistas para encontrar o parâmetro de relaxação  $\lambda$  utilizado em conjunto com a abordagem POCS na restauração das imagens. Este documento está organizado conforme segue: O Capítulo 2 relata a teoria sobre restauração de imagens. O Capítulo 3 relata sobre o algoritmo de projeção em conjuntos convexos RAP. O Capítulo 4 discute sobre algoritmos evolucionistas. O Capítulo 5 apresenta a metodologia utilizada. O Capítulo 6 discorre sobre os resultados dos experimentos e o Capítulo 7 a conclusão do trabalho.

## 2 Restauração de Imagens

A utilização de técnicas de correção de imagens é uma das mais importantes etapas do processamento digital de imagens. Da mesma maneira que as técnicas de realce enfatizam feições na imagem, elas realçam também as imperfeições da mesma. Sendo assim, torna-se necessário que ruídos e imperfeições sejam retirados ou atenuados antes da aplicação das técnicas de realce [8].

As principais fontes de ruído em imagens digitais surgem durante a aquisição e/ou transmissão das imagens. O desempenho de diferentes sensores (satélites e microscópios) de aquisição de imagens é afetado por uma série de fatores, como condições ambientais durante aquisição da imagem e a qualidade dos elementos sensores em si. Na aquisição de imagens, a iluminação e a temperatura do sensor são fatores importantes que afetam a quantidade de ruído da imagem resultante, fazendo com que algoritmos específicos para remoção de ruídos e devidas correções sejam desenvolvidos. As técnicas de correção de imagens fazem parte da etapa de pré-processamento, incluindo basicamente correções radiométricas, geométricas e atmosféricas. As imagens também são corrompidas durante a transmissão, principalmente em razão de interferências no canal utilizado para transmissão. Uma imagem transmitida utilizando uma rede sem fio, por exemplo, pode ser corrompida como resultado de relâmpagos ou outros distúrbios atmosféricos.

Dentre as técnicas de correção radiométrica, podem ser citadas o realce e a restauração de imagens. A diferença básica entre elas é que o objetivo da restauração é recuperar a cena original através de observações das imagens degradadas. Essas técnicas são orientadas através de modelagens dos processos de degradação, borramento e ruído e, conseqüentemente, aplicar um processo inverso para obter uma aproximação da cena original. Por outro lado, tem-se as técnicas de realce, as quais são designadas para manipular a imagem no âmbito de produzir bons resultados ao observador, sem utilizar algum modelo de degradação [9].

O campo de restauração de imagens começou com esforços de cientistas envolvidos por programas espaciais dos Estados Unidos e da antiga União Soviética nos anos 50. As imagens capturadas nas missões espaciais eram sujeitas a muitas degradações fotográficas. Como o custo para aquisição dessas imagens era muito alto, a restauração passou a ser um processo

de fundamental importância. São várias as áreas de aplicação de restauração de imagens digitais, sendo que uma das principais é a de imagens astronômicas (alguns anos atrás, umas das principais imagens a serem restauradas eram as obtidas pelo telescópio espacial Hubble) a restauração neste caso seria para a remoção do ruído Poisson ao qual tem relação direta com a baixa contagem de fótons em uma imagem. Outra área de grande importância é a de imagens médicas, a área criminal é outra em que a restauração é necessária. Um bom exemplo seria a recuperação de vídeos ou fotografias de cenas de crime onde se quer identificar um assassino ou reconhecer objetos, como placas de carros. Existem muitos outros tipos de ruídos em imagens tais como Speckle que ocorre em imagens de radar, adquiridas a laser, sua característica em geral é corromper mais as partes claras que as escuras de uma imagem, um filtro de restauração nesse caso seria o filtro de Kuan.

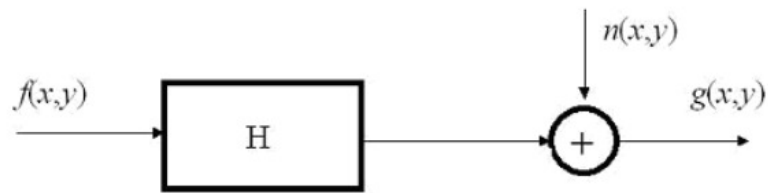
A restauração de imagens pode ser entendida também como uma técnica utilizada para corrigir as distorções introduzidas pelos sensores dos sistemas de imageamento. O efeito do sensor sobre a imagem é o de suavização (filtro passa-baixas) dos detalhes. A correção da imagem é baseada nas características do sensor. Portanto, para cada satélite, sensor e banda espectral, existe um filtro adequado.

Quanto às técnicas de correção geométrica, as mesmas se ocupam principalmente do registro de dados das imagens para que suas respostas possam ser interpretadas para uma certa posição no espaço. Na verdade, a justificativa mais contemporânea para a correção geométrica de imagens é a integração com mapas e outras informações. Diferentemente, as técnicas de correção atmosférica se preocupam em atenuar o efeito atmosférico, o qual influi na resposta de um alvo de dois modos: a absorção atmosférica subtrai valores de brilho de energia radiante do alvo, e o espalhamento atmosférico adiciona valores à resposta original.

## 2.1 Modelo de Degradação

O processo de degradação é modelado como um operador ou sistema físico  $H$ , que juntamente com um termo de ruído aditivo  $n(x, y)$ , opera sobre uma imagem de entrada  $f(x, y)$  produzindo uma imagem degradada  $g(x, y)$  [1]. Tal procedimento é ilustrado pela Figura 2.1.

A restauração de imagens digitais pode ser vista como o processo de obtenção de uma aproximação de  $f(x, y)$ , dado  $g(x, y)$ , algum conhecimento sobre a função de degradação  $H$  e algum conhecimento sobre o termo de ruído aditivo  $n(x, y)$ . Existem vários tipos de degra-



**Figura 2.1. Modelo do processo de degradação da imagem.Extraído de: [1].**

dação, sendo que podem ser destacados o borramento e o ruído. O borramento acontece no processo de formação da imagem e pode ser causado pelo movimento relativo entre o sistema óptico e a cena, por um sistema óptico fora de foco ou por aberrações do mesmo e turbulência atmosférica nas imagens aéreas.

Usualmente, os tipos de degradação por borramento podem ser modelados por um filtro passa-baixas que atenua a informação referente aos contornos da imagem, os quais são muito importantes para a percepção da visão humana. Dentre os principais modelos matemáticos que representam os operadores de degradação, podem ser destacados o borramento por movimento, o qual representa o deslocamento da câmera ou o movimento abrupto do objeto através de um eixo horizontal ou vertical e o borramento Gaussiano. Já o ruído pode ser modelado, principalmente, por duas importantes distribuições: Poisson e Gaussiana. A primeira é dependente do sinal e está relacionada à contagem de fótons envolvidos com fontes de luz ou baixa energia como, por exemplo, em imagens fotográficas obtidas em um ambiente com pouca iluminação. A segunda é um ruído aditivo, em geral com média zero, sendo amplamente utilizada para modelar uma grande variedade de ruídos não correlacionados com o sinal.

O modelo de restauração de imagens pode ser dado pelo seguinte sistema linear:

$$g = Hf + n, \quad (2.1)$$

onde  $g$ ,  $H$ ,  $f$  e  $n$  possuem as mesmas definições dadas na Figura 2.1. Assim sendo, a imagem degradada  $g$  pode ser obtida através da convolução entre a imagem original  $f$  e a função de espalhamento pontual  $H$ , sendo essa convolução do tipo circular, adicionado o termo de ruído aditivo  $n$ .

Um dos principais problemas em restauração de imagens é devolver os detalhes perdidos pelo borramento inserido pela função de espalhamento pontual (*Point Spread Function* - PSF), mas com o compromisso de manter aceitáveis os níveis de ruído. As próximas seções descrevem algumas abordagens tradicionais para restauração de imagens.

## 2.2 Filtragem Inversa

A filtragem inversa tenta recuperar uma imagem degradada por uma função de degradação  $H$  conforme descrito na Seção 2.1. Essa é uma abordagem simples, na qual verifica-se que nas condições ideais de linearidade, ausência de ruído e invariância no espaço, é possível recuperar

a imagem original pela razão entre as Transformadas de Fourier da imagem degradada e da função de espalhamento [1], como descrito na Equação 2.2:

$$G(u, v) = H(u, v)F(u, v), \quad (2.2)$$

$$\hat{F}(u, v) = \frac{G(u, v)}{H(u, v)}. \quad (2.3)$$

Entretanto, o filtro inverso é instável pois, na presença de ruído, a equação 2.2 resulta em:

$$G(u, v) = H(u, v)F(u, v) + N(u, v). \quad (2.4)$$

Pela filtragem inversa, temos:

$$\hat{F}(u, v) = \frac{G(u, v)}{H(u, v)} = \frac{H(u, v)F(u, v) + N(u, v)}{H(u, v)} = F(u, v) + \frac{N(u, v)}{H(u, v)}. \quad (2.5)$$

Essa expressão indica que, mesmo conhecendo a função de degradação, não poderemos recuperar a imagem não degradada (a transformada inversa de Fourier de  $F(u, v)$ ) justamente porque  $N(u, v)$  não é conhecida, e se a função de degradação tiver zeros ou valores muito pequenos a razão  $\frac{N(u, v)}{H(u, v)}$  pode facilmente dominar a estimativa  $\hat{F}(u, v)$  [1]. A filtragem inversa é muito sensível ao ruído quando  $H(u, v)$  tende rapidamente a zero. Existem, geralmente, problemas causados por excessiva amplificação de ruído em altas frequências, mesmo quando a resposta em frequência de  $H(u, v)$  não tende realmente para zero. Isto ocorre porque o espectro de potência da imagem degradada é tipicamente maior em baixas frequências, e diminui sensivelmente para frequências mais altas. Por sua vez, o espectro do ruído possui componentes de maior amplitude em frequências altas. Assim, a filtragem inversa conduz a uma imagem estimada com ruído inicial "amplificado" em altas frequências. Além disso, caso a função de espalhamento seja conhecida apenas de forma aproximada, é pouco provável que se consiga uma restauração satisfatória.

A instabilidade numérica que ocorre no filtro inverso devido a presença de zeros em  $H(u, v)$  e do ruído  $N(u, v)$  está intimamente ligada ao fato de a equação que caracteriza a degradação da imagem ser essencialmente um problema de mau condicionamento, podendo pequenas perturbações na imagem degradada provocar grandes variações na imagem restaurada.

### 2.3 Filtro de Wiener

O *filtro de Wiener* é uma das melhores abordagens para restauração linear de imagens. Pode ser entendido como uma técnica que incorpora tanto a função de degradação quanto as características estatísticas do ruído no processo de restauração. O método considera imagens e ruído como



sendo variáveis aleatórias, e seu objetivo é encontrar uma estimativa  $\hat{f}$  da imagem não corrompida  $f$ , de forma que o erro quadrático seja minimizado. Essa medida é dada pela Equação 2.6:

$$e^2 = E[(f - \hat{f})^2], \quad (2.6)$$

na qual  $E$  é o valor esperado do argumento, supondo que o ruído e a imagem não sejam correlacionados. Com base nessas condições, o mínimo da função de erro da Equação 2.6 é dado no domínio da frequência pela seguinte expressão:

$$\hat{F}(u, v) = \left[ \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_n(u, v)/S_f(u, v)} \right] G(u, v). \quad (2.7)$$

Os termos da equação acima são definidos por:

$H(u, v)$  = função de degradação

$H^*(u, v)$  = conjugado complexo de  $H(u, v)$

$|H(u, v)|^2 = H^*(u, v)H(u, v)$

$S_n(u, v) = |N(u, v)|^2$  = espectro de potência do ruído

$S_f(u, v) = |F(u, v)|^2$  = espectro de potência da imagem não degradada

O resultado da Equação 2.7 é conhecido como o *filtro de Wiener* [1].

### 3 Projeções em Conjuntos Convexos (POCS)

O modelo linear para restauração de imagens pode ser descrito pela Equação 2.1. Entretanto, o mesmo, sem a estimativa do ruído, pode ser representado como:

$$g = Hf. \quad (3.1)$$

Desta forma, a Equação 3.1 descreve um sistema linear, onde cada linha de  $g$  denota uma equação que pode ser representada por um hiperplano, o qual é um conjunto convexo, e sua solução pode ser encontrada pelo método POCS.

O uso da técnica POCS para solução de sistemas lineares do tipo  $Ax = b$  através de projeções ortogonais iterativas entre os hiperplanos descritos por cada equação, é também chamado de algoritmo de *Row-Action Projections*, (RAP) ou *Algebraic Reconstruction Technique* (ART), o qual foi inicialmente desenvolvido em 1937 por Kaczmarz [6].

O método utilizado foi desenvolvido por [2], que converge na intersecção dos hiperplanos, sendo a equação do RAP dada por:

$$f^{(k+1)} = f_k + \lambda \frac{g_p - h_{(p,n)} f_k}{\|h_p\|^2} h_p, \quad (3.2)$$

onde  $\lambda$  é o parâmetro de relaxação,  $g_p$  é o p-ésimo elemento do vetor  $g$  (imagem degradada),  $h_{(p,n)}$  é a p-ésima linha correspondente a n-ésima coluna da matriz  $H$  e  $f^{(k+1)}$  é a projeção de  $f_k$  no hiperplano correspondente.

A formulação do RAP pode ser dada por:

$$\begin{bmatrix} g_0 \\ \dots \\ g_p \\ \dots \\ g_n \end{bmatrix} = \begin{bmatrix} h_{00} & \dots & \dots & \dots & h_{0n} \\ \dots & \dots & \dots & \dots & \dots \\ h_{p0} & \dots & \dots & \dots & h_{pn} \\ \dots & \dots & \dots & \dots & \dots \\ h_{n0} & \dots & \dots & \dots & h_{nn} \end{bmatrix} \cdot \begin{bmatrix} f_0 \\ \dots \\ \dots \\ \dots \\ f_n \end{bmatrix}$$

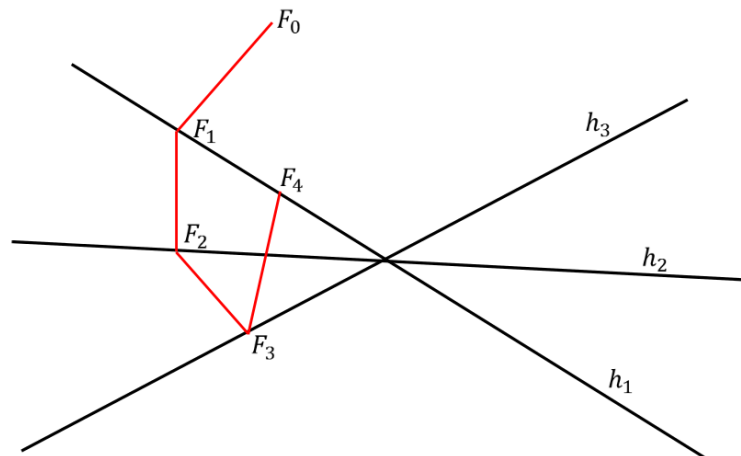
Onde o primeiro elemento é um vetor coluna de tamanho  $n^2 \times 1$ , o elemento do meio (matriz  $H$ ) é uma matriz quadrada toeplitz bloco circulante  $n^2 \times n^2$ , e o último elemento é a imagem observada  $f$  no formato lexicográfico, isto é, um vetor coluna de tamanho  $n^2 \times 1$ . Na primeira iteração,  $g_0$  é o resultado do produto escalar entre a primeira linha da matriz  $H$  com o vetor coluna  $f$ , que na primeira iteração é a imagem observada (degradada), conforme a Equação 3.3:

$$g_0 = \sum_{n=0}^{n-1} h_{0n} f_n, \quad (3.3)$$

podemos generalizar para:

$$g_i = \sum_{n=0}^{n-1} h_{in} f_n. \quad (3.4)$$

O produto escalar realizado acima nada mais é que a convolução, pois  $h_0$  tem  $n$  termos assim como  $f$ . Logo, é permitida a multiplicação matriz  $\times$  vetor. Suponha que seja usado uma *Point Spread Function* - PSF (informação sobre o borramento) de  $3 \times 3$  elementos. Logo, na linha  $h_0$  existirão apenas nove elementos (referentes aos valores da PSF) não nulos, e  $n - 9$  elementos nulos, esses zeros são propositais para simular o processo de convolução. Para calcular, por exemplo,  $g_1$ , o vetor coluna  $f$  não muda, mas o  $h_1$  vai mudar, pois simula a máscara da convolução andando, da primeira linha para a segunda, três valores que eram não nulos passam a ser nulos e 3 valores que eram nulos passam a ser não nulos. O artifício matemático utilizado para simular o processo de convolução é a matriz esparsa Toeplitz bloco circulante. Quando é feita analogia da convolução em um ponto ser igual ao produto de dois vetores (vetor linha de  $H \times$  vetor coluna  $f$ ), é possível interpretar isso geometricamente conforme a Figura 3.1:



**Figura 3.1. Interpretação geométrica RAP.Extraído de: [2]**

Cada vez que é calculada a convolução em um ponto da imagem, na verdade é projetado  $f$  em uma linha de  $H$ . Como  $H$  possui muitas linhas, cada uma delas define um hiperplano que, quando é projetado sucessiva vezes a idéia é que vai chegar no centro, se existir intersecção entre as retas. Essa interpretação geométrica só é possível pois é feito o processo de convolução mas com produto entre vetores. O RAP é um algoritmo iterativo, a imagem restaurada na primeira iteração é utilizado na segunda, a imagem restaurada na segunda iteração é utilizada na terceira assim até chegar no número máximo de iterações. Cabe destacar que o sucesso da implementação do método RAP depende do número de iterações e do valor de  $\lambda$ .

### 3.1 Conjunto de Restrição de Amplitude Limitada

Além dos conjuntos de restrição expressos pela Equação 3.1, é possível também considerar algum conhecimento *a priori* da imagem a ser restaurada. Este conjunto descreve o limite superior e inferior do valor do pixel. Caso o valor do pixel referente a restauração em uma iteração seja superior a 255 seu valor é alterado para 255, caso seu valor seja inferior a zero seu valor é alterado para zero. A Equação 3.5 descreve este conjunto, onde  $\alpha$  é o limite inferior,  $\beta$  é o limite superior e  $\Omega$  é a região onde se encontra a imagem

$$C_{AL} = \{s : s \in S \text{ and } \alpha \leq s(k, l) \leq \beta \forall k, l \in \Omega\}, \quad (3.5)$$

onde  $S$  é o espaço de Hilbert. A regra de projeção no conjunto  $C_{AL}$  é descrita pela Equação 3.6:

$$P_{C_{AL}}y(k, l) = \begin{cases} \alpha, & \text{if } y(k, l) < \alpha \\ y(k, l), & \text{if } \alpha \leq y(k, l) \leq \beta \\ \beta, & \text{if } y(k, l) > \beta. \end{cases} \quad (3.6)$$

## 4 Algoritmos Evolucionistas

Com a existência de uma grande quantidade de aplicações da engenharia, física e matemática os algoritmos evolucionistas são considerados uma ferramenta indispensável na análise de soluções de problemas referentes a otimização ao qual pode-se maximizar ou minimizar uma função de forma clara e objetiva, podendo com isso melhorar consideravelmente o desempenho de muitos processos que incluem tais funções.

Segundo Yang [10], as décadas de 60 e 70 do século XX foram muito importantes para o desenvolvimento dos algoritmos evolucionistas. Primeiramente, John Holland e seus colaboradores da Universidade de Michigan desenvolveram a técnica de Algoritmos Genéticos (*Genetic Algorithms* - GA) a qual é um método de otimização e busca que utiliza alguns conceitos da genética e se baseia na evolução de uma população de seres vivos. Inspirado também na teoria do princípio da seleção natural e sobrevivência de indivíduos mais aptos, do naturalista e fisiologista Charles Darwin, que segundo sua teoria, quanto melhor for a genética de um indivíduo em uma determinada população, maior será a chance dele se adaptar ao meio em que está inserido e maior será sua probabilidade de sobreviver e gerar descendentes, enquanto os indivíduos menos aptos tendem a desaparecer.

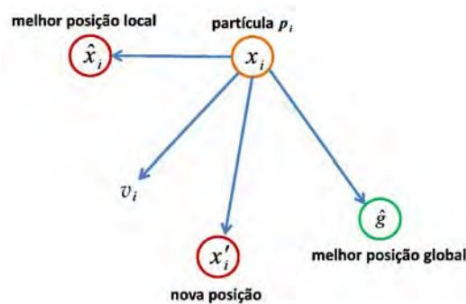
Outro significativo progresso foi em 1995 quando foi desenvolvido o algoritmo de Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO) pelo americano e psicólogo James Kennedy e o engenheiro Russell C. Eberhart [11]. PSO é um algoritmo de otimização inspirado em enxame de partículas de peixes, pássaros e pelo comportamento humano. Um algoritmo evolucionista de forma genérica tem como passo inicial a geração de uma população aleatória formada de possíveis soluções que, dependendo do problema, podem ser simples ou complexas. Essa população passa por uma avaliação chamada aptidão, que vai refletir na qualidade da solução, isto é, os membros mais aptos são selecionados enquanto os menos aptos são descartados. Nessa fase de escolha ficam definidos quem serão os pais e quem serão os filhos. A partir daí os filhos começam a sofrer processos de recombinação e mutação em suas características fundamentais gerando novos descendentes para a próxima população. Essa etapa é repetida inúmeras vezes até que se consiga chegar a uma solução satisfatória.

As próximas seções tratam de apresentar os algoritmos: Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO), Busca Harmônica (*Harmony Search* - HS) e suas variantes, Algoritmo de Busca Gravitacional (*Gravitational Search Algorithm* - GSA), Algoritmo dos vagalumes (*Firefly Algorithm* - FFA) e Algoritmo dos Morcegos (*Bat Algorithm* - BA).

### 4.1 Otimização por Enxame de Partículas - PSO

Recentemente, várias aplicações tem utilizado técnicas evolucionárias como métodos heurísticos para encontrar soluções ótimas ou pseudo-ótimas. Uma atenção particular tem sido dada

à técnica denominada Otimização por Enxame de Partículas (*Particle Swarm Optimization - PSO*), devido à sua simplicidade e eficácia. Basicamente, PSO é um algoritmo modelado em inteligência coletiva que encontra uma solução em um espaço de busca com base na dinâmica do comportamento social [12]. Cada possível solução do problema é modelada como uma partícula do enxame que imita seu vizinho baseando-se em uma função objetivo. Outras definições consideram PSO como um algoritmo de pesquisa baseado em processos estocásticos e populacionais, onde a aprendizagem do comportamento social permite a cada solução possível (partícula) "voar" dentro desse espaço (enxame) à procura de outras partículas que possuam melhores características e, assim, maximizando a função objetivo. Cada partícula tem uma memória para armazenar sua melhor solução local (máximos locais) e a melhor solução global (máximos globais). Levando-se em conta essas informações, cada partícula tem a capacidade de imitar as outras que proporcionam a ela melhores posições no enxame. Este processo simula a interação social entre um bando de aves a procura de comida, por exemplo. Esse mecanismo sócio-recognitivo pode ser resumido em três grandes princípios [12]: (i) avaliação, (ii) comparação e (iii) imitação. Cada partícula pode avaliar outras dentro de sua vizinhança através de alguma função objetivo, pode compará-la com seu próprio valor e, finalmente, pode decidir se é uma boa escolha imitá-la. O enxame é modelado como sendo um espaço multidimensional  $R^m$ , em que cada partícula  $p_i = (x_i, v_i) \in R^m$  tem duas características principais: (i) sua posição  $x_i$  e (ii) velocidade  $v_i$ . A melhor solução (posição no enxame) local  $\hat{x}_i$  e global  $\hat{g}$  são também conhecidas.



**Figura 4.1. Interpretação geométrica do PSO.**Extraído de: [3]

Após a definição do tamanho do enxame, ou seja, o número de partículas, cada uma delas é inicializada com valores aleatórios de velocidade e posição. Cada indivíduo é então avaliado com relação a alguma função objetivo e seu máximo local é atualizado. No final, o valor máximo global é atualizado com a partícula que alcançou a melhor posição no enxame. Este processo é repetido até que algum critério de convergência seja atingido. A posição atualizada e equações da velocidade da partícula  $p_i$  da forma mais simples que governam o PSO são, respectivamente, dadas por:

$$v_i = wv_i + c_1r_1(\hat{x}_i - x_i) + c_2r_2(\hat{s} - x_i) \quad (4.1)$$

e

$$x_i = x_i + v_i, \quad (4.2)$$

onde  $w$  é a força de inércia que controla o poder de interação entre as partículas, e  $r_1, r_2 \in [0, 1]$  são variáveis aleatórias que dão a ideia de estocasticidade ao método PSO. As constantes  $c_1$  e  $c_2$  são também utilizadas para guiar as partículas e definidas como parâmetros de entrada para o algoritmo.

## 4.2 Busca Harmônica - HS

Busca Harmônica é inspirado no processo musical pela busca por um estado perfeito de harmonia, como é feito durante a improvisação do jazz. Cada músico de jazz toca um possível acorde musical, que juntos, formarão uma harmonia. Caso a harmonia gerada seja boa, esta experiência é guardada em uma memória do músico, para ser usada futuramente, aumentando as chances de melhorar a harmonia gerada em uma próxima rodada.

Analogamente, no processo de otimização cada possível solução chamada harmonia é inicialmente gerada de forma aleatória dentro de um intervalo determinado, e combinadas, geram uma solução de acordo com a função objetivo. Caso esta seja uma boa solução, estas variáveis são guardadas em uma memória, para que sejam usadas na geração de novas soluções, aumentando as possibilidades de melhorar os resultados. O processo de otimização do algoritmo HS consiste em cinco etapas [13].

### 4.2.1 Etapa 1 - O Problema da Otimização e os Parâmetros do Algoritmo

Para descrever como funciona a Busca Harmônica, um problema de otimização é especificado:

$$\text{Maximizar ou Minimizar } f(x^j) \text{ sujeito a } x^j \in X_j, \forall j = 1, 2, \dots, N, \quad (4.3)$$

onde  $f(x^j)$  é a função objetivo,  $x^j$  é o conjunto das variáveis de decisão e  $X_j$ , é o conjunto de possíveis valores para cada variável de decisão, e  $N$  é o número de variáveis do projeto.

Os parâmetros do algoritmo HS necessários para resolver o problema de otimização também são especificados nessa etapa: o tamanho da memória harmônica (*Harmony Memory*

*Size - HMS*), a taxa de memória harmônica considerada (*Harmony Memory Considering Rate - HMCR*), a taxa de passo de ajuste (*Pitch Adjusting Rate - PAR*), e o critério de parada. Os parâmetros HMCR e PAR são usados para melhorar o vetor solução, ou seja, eles podem ajudar o algoritmo a encontrar soluções a nível global e local.

#### 4.2.2 Etapa 2 - Memória Harmônica

Memória Harmônica (MH) é inicializada com vetores solução gerados aleatoriamente com seus respectivos valores da função objetivo:

$$MH = \left[ \begin{array}{cccc|c} x_1^1 & x_1^2 & \dots & x_1^N & f(x^1) \\ x_2^1 & x_2^2 & \dots & x_2^N & f(x^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{HMS}^1 & x_{HMS}^2 & \dots & x_{HMS}^N & f(x^{HMS}) \end{array} \right], \quad (4.4)$$

onde  $x_i^j$  denota a variável de decisão  $j$  da harmonia  $i$ .

#### 4.2.3 Etapa 3 - Gerando uma Nova Harmonia

Um novo vetor de harmonia  $\hat{x}=(\hat{x}^1,\hat{x}^2,\dots,\hat{x}^N)$  é gerado com base nas considerações da memória e randomização (improvisação musical), como descrito a seguir:

$$\hat{x}^i \leftarrow \begin{cases} \hat{x}^i \in \{x_i^1, x_i^2, \dots, x_i^N\} & \text{com probabilidade HMCR,} \\ \hat{x}^i \in X_i & \text{com probabilidade (1-HMCR).} \end{cases} \quad (4.5)$$

Assim, temos que HMCR é a probabilidade de escolher um valor a partir de valores armazenados na MH, e (1-HMCR) é a probabilidade de escolher aleatoriamente um valor viável não se limitando aos armazenados na MH.

Além disso, cada componente  $i$  do novo vetor de harmonia  $\hat{x}$  é examinado para determinar se ele deve ser ajustado:

$$\hat{x}^i \leftarrow \begin{cases} \text{Sim com probabilidade PAR,} \\ \text{Não com probabilidade (1-PAR).} \end{cases} \quad (4.6)$$

O ajuste de cada variável de decisão é frequentemente utilizado com o intuito de melhorar as soluções e evitar ótimos locais. Trata-se de um mecanismo que muda os valores vizinhos de algumas variáveis de decisão na harmonia. Assim, se a decisão de ajuste para a variável de decisão  $\hat{x}^i$  for “Sim”,  $\hat{x}^i$  é substituído da seguinte forma:

$$\hat{x}^i \leftarrow \hat{x}^i \pm rb. \quad (4.7)$$



onde  $b$  é uma distância arbitrária, e  $r \sim U(0,1)$ .

#### **4.2.4 Etapa 4 - Atualizando a Memória Harmônica**

Se a nova harmonia for melhor do que a pior harmonia na MH, esta última é substituída por esta nova harmonia. Note que a melhor harmonia é definida como aquela que maximiza a função objetivo.

#### **4.2.5 Etapa 5 - Critério de Parada**

O algoritmo de Busca Harmônica finaliza sua execução quando ele satisfaz o critério de parada. Caso contrário, as Etapas 3 e 4 são repetidas a fim de improvisar novamente uma nova harmonia. As subseções seguintes descrevem as variações do HS utilizadas.

### **4.3 Global-best Harmony Search - GHS**

Inspirados no conceito de inteligência de enxames, como proposto no algoritmo [12], Omram et al. [14] propuseram uma nova abordagem para a busca harmônica, chamada de *Global-best Harmony Search* (GHS).

O GHS modifica o ajuste de *pitch* do HS de forma que a nova harmonia possa imitar a melhor harmonia na MH, adicionando uma dimensão social ao HS. O GHS tem exatamente os mesmos passos do HS, exceto pelo Passo 3, que é modificado como segue:

### Algoritmo 1 – PASSO 3 GHS

1. **Para**  $i = 1$  até  $N$  **faça**
2.     **Se**  $rand(0,1) < HMCR$ . **Então**
3.          $\hat{x}_i = \hat{x}_i^j \forall j = 1, 2, \dots, N$ ,
4.         **Se**  $rand(0,1) < PAR(t)$ . **Então** /\* ajuste de *pitch* \*/
5.              $\hat{x}_i = \hat{x}_{best}$
6.         **Senão**
7.              $\hat{x}_i = LB_i + r \times (UB_i - LB_i)$

onde  $j \in U[1, \dots, HMS]$ ,  $best$  é o índice da melhor harmonia na MH,  $LB_i$  e  $UB_i$  são os limites superiores e inferiores de cada variável de decisão, respectivamente. Além disso o GHS emprega o procedimento para a atualização dinâmica do parâmetro PAR como na equação 4.11:

$$PAR(t) = PAR_{min} + \frac{(PAR_{max} - PAR_{min})}{T} \times t. \quad (4.8)$$

#### 4.4 Self-adaptative Global Best Harmony Search - SGHS

Baseados no GHS, Quan-Ke Pan et al. [15] desenvolveram um GHS auto-adaptativo (*Self-adaptative Harmony Search - SGHS*), que emprega um novo esquema de improvisação e um novo método de ajuste adaptativo de parâmetro.

O GHS toma vantagem do melhor vetor de harmonia  $x_{best}$  para produzir um novo vetor  $\hat{x}$ . Entretanto, a regra de ajuste de *pitch* pode alterar  $\hat{x}_{best}$ , de forma que  $\hat{x}$  seja um vetor pior que  $x_{best}$ . Logo, para manter as boas características de  $x_{best}$ , um novo ajuste de *pitch* é apresentado a seguir:

$$\hat{x}^i = x_{best}^i, i = 1, 2, \dots, N. \quad (4.9)$$

Neste caso,  $\hat{x}^i$  é atribuído de acordo com a variável correspondente  $x_{best}^i$  em  $x_{best}$ , enquanto no GHS  $\hat{x}^i$  é determinado selecionando-se aleatoriamente uma variável de  $x_{best}$ .

O novo método de improvisar uma nova harmonia  $\hat{x}$  no SGHS pode ser resumido:

### Algoritmo 2 – PASSO 3 SGHS

1. **Para**  $i = 0$  até  $N$  **faça**
2.     **Se**  $rand(0,1) < HMCR(t)$ . **Então**
3.          $\hat{x}^i = \hat{x}^j \pm r \times bw$ , onde  $r \in (0, 1)$ ,  $\forall j = 1, 2, \dots, N$ ,
4.         **Se**  $rand(0,1) < PAR(t)$ . **Então**
5.              $\hat{x}^i = x_{best}^i$

6. **Senão**  
 7.  $\hat{x}^i = LB_i + r \times (UB_i - LB_i)$

O parâmetro  $bw$  é uma distância arbitrária. Utilizado para bem equilibrar a exploração da memória harmônica, o valor  $bw$  diminui dinamicamente com as gerações crescentes (IT), como segue:

$$bw(t) = \begin{cases} bw = bw_{max} - \frac{(bw_{max} - bw_{min})}{IT} \times 2t, & Se, t < IT/2, \\ bw = bw_{min} & if, t \geq IT/2. \end{cases} \quad (4.10)$$

#### 4.5 Improved Harmony Search - IHS

A fim de melhorar o desempenho do algoritmo e eliminar desvantagens pelo uso de valores fixos de HMCR e PAR, Mahdavi [14] propuseram uma busca harmônica aperfeiçoada *Improved Harmony Search - IHS*, atualizando as variáveis PAR e  $bw$  dinamicamente.

As atualizações de PAR e  $bw$  no algoritmo IHS são dadas por:

$$PAR(t) = PAR_{min} + \frac{(PAR_{max} - PAR_{min})}{T} \times t, \quad (4.11)$$

sendo  $PAR(t)$  a taxa de ajuste de *pitch* na geração  $t$ ,  $PAR_{min}$  a taxa de ajuste mínima,  $PAR_{max}$  a taxa de ajuste máxima, e  $T$  o número total de iterações. O parâmetro  $bw$  é atualizado como segue:

$$bw(t) = bw_{max} e^{\left( \frac{\ln\left(\frac{bw_{max}}{bw_{min}}\right)}{T} \times \frac{t}{T} \right)}, \quad (4.12)$$

onde  $bw(t)$  é a largura de banda na geração  $t$ ,  $bw_{min}$  é a largura de banda mínima e  $bw_{max}$  é a largura de banda máxima.

A atualização do parâmetro PAR segue de acordo com a equação 4.13:

$$PAR(t) = PAR_{min} + (PAR_{max} - PAR_{min}) \times t. \quad (4.13)$$

#### 4.6 Novel Global Harmony Search - NGHS

Também inspirado na inteligência de enxames, Zou et al. [16] propuseram uma variação do HS chamada *Novel Global Harmony Search (NGHS)*, onde a nova harmonia procura imitar a melhor harmonia na Memória Harmônica. O NGHS difere do HS em três aspectos:

- No passo 1, a taxa de memória harmônica considerada (HMCR) e o ajuste de *pitch* (PAR) são excluídos do NGHS, e uma probabilidade de mutação genética ( $p_m$ ) é adicionada;
- No Passo 3, o NGHS modifica os passos de improvisação do HS, como demonstrado no algoritmo 3;
- No Passo 4, o NGHS substitui a pior harmonia  $x_{worst}$  da memória harmônica pela nova harmonia  $\hat{x}$  mesmo se  $\hat{x}$  for pior que  $x_{worst}$ .

### Algoritmo 3 – PASSO 3 NGHS

1. **Para**  $i = 0$  até  $N$  **faça**
2.      $x_r = 2 \times \hat{x}_{best}^i - \hat{x}_{worst}^i$
3.     **Se**  $x_r > UB_i$  **Então**
4.          $x_r = UB_i$
5.     **Senão. Se**  $x_r < LB_i$ . **Então**
6.          $x_r = LB_i$
7.      $\hat{x}^i = \hat{x}_{worst}^i - r \times (x_r - \hat{x}_{worst}^i)$  /\* atualizar posição \*/
8.     **Se**  $rand(0, 1) \leq p_m$ . **Então**
9.          $\hat{x}^i = LB_i + rand(0, 1) \times (UB_i - LB_i)$  /\* mutação genética \*/

## 4.7 Algoritmo de Busca Gravitacional - GSA

A gravidade é uma das quatro interações fundamentais da natureza, juntamente com a força forte, o eletromagnetismo e a força nuclear fraca. A ideia que diz respeito às regras de gravidade é o fato de que um objeto com massa atrai outro. Uma das teorias mais aceitas é a lei da gravitação universal de Newton, que diz que “todas as partículas no universo atraem outras com uma força que é diretamente proporcional ao produto das suas massas e inversamente proporcional ao quadrado da distância entre elas” [17], ou seja:

$$F = G \frac{M_1 M_2}{R^2}, \quad (4.14)$$

onde  $M_1$  e  $M_2$  correspondem às massas de duas partículas,  $R$  é a distância entre elas,  $G$  é uma constante gravitacional e  $F$  é a magnitude da força gravitacional. A “constante gravitacional”  $G$  é dependente do tempo, e diminui com a idade do universo:

$$G(t) = G(t_0) \frac{t_0^\beta}{t}, \quad \beta < 1, \quad (4.15)$$

em que  $G(t)$  é o valor da constante gravitacional no tempo  $t$  e  $G(t_0)$  é o valor da constante gravitacional no momento da “criação do universo” que está sendo considerado.

A segunda lei de Newton diz que quando uma força  $F$  é aplicada a uma massa, a sua aceleração  $a$  depende somente dessa força e da sua massa  $M$ :

$$a = \frac{F}{M}. \quad (4.16)$$

Com base nas afirmações acima, Rashedi *et al.* [18] propuseram o Algoritmo de Busca Gravitacional (*Gravitational Search Algorithm - GSA*), que pode ser definido como segue. Seja  $X = \{x_1, x_2, \dots, x_n\}$  um universo com  $n$  massas, de modo que  $x_i \in \mathbb{R}^m$ . Pode-se definir, em um momento específico  $t$ , a força que age sobre a massa  $i$  a partir da massa  $j$  na dimensão  $d$  da seguinte forma:

$$F_{ij}^d(t) = G(t) \frac{M_i(t)M_j(t)}{R_{ij}(t) + \epsilon} (x_j^d(t) - x_i^d(t)), \quad (4.17)$$

onde  $R_{ij}(t)$  é a distância euclidiana entre as massas  $i$  e  $j$ , e  $\epsilon$  é uma pequena constante.

A fim de dar um comportamento estocástico para o GSA, Rashedi *et al.* [18] assumem a força total que age sobre a massa  $i$  em uma dimensão  $d$  como uma soma aleatória ponderada das forças exercidas de outros agentes:

$$F_i^d(t) = \sum_{j=1, j \neq i}^m \gamma_j F_{ij}^d(t), \quad (4.18)$$

onde  $\gamma_j$  denota um número gerado aleatoriamente entre 0 e 1. A aceleração da massa  $i$  em tempo  $t$  e dimensão  $d$  é dada por:

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)}, \quad (4.19)$$

em que a massa  $M_i$  é calculada do seguinte modo:

$$M_i(t) = \frac{q_i(t)}{\sum_{j=1}^m q_j(t)}, \quad (4.20)$$

com

$$q_i(t) = \frac{C_i(t) - w(t)}{b(t) - w(t)}. \quad (4.21)$$

Os termos  $w(t)$  e  $b(t)$  significam, respectivamente, as massas com melhor e pior valor de função objetivo. O termo  $C_i(t)$  denota o vetor de características da massa  $i$ , ou seja, o conjunto de variáveis a serem otimizadas, as quais estão representadas pela massa  $i$ .

Finalmente, para evitar soluções ótimas locais, somente as  $k$  melhores massas, ou seja, aquelas com maiores valores de função objetivo, vão atrair as outras. Seja  $\mathcal{K}$  um conjunto dessas massas. O valor de  $k$  é definido como  $k_0$  no início do algoritmo e diminui com o tempo. Assim, a Equação 4.18 é reescrita como:

$$F_i^d(t) = \sum_{j \in \mathcal{K}, j \neq i} \gamma_j F_{ij}^d(t). \quad (4.22)$$

A velocidade e a posição de atualização das massas são dadas por:

$$v_i^d(t+1) = \gamma_j v_i^d(t) + a_i^d(t) \quad (4.23)$$

e

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1). \quad (4.24)$$

#### 4.8 Algoritmo dos Vagalumes - FFA

A técnica FFA, proposta por Yang [10], é derivada do comportamento dos vaga-lumes, os quais utilizam a luz como atrativo na busca de parceiros para acasalamento e comunicação. O brilho de um vaga-lume é determinado por alguma função objetivo, sendo que a intensidade de luz percebida  $I$  depende da sua distância  $d$  da fonte, como segue:

$$I = I_0 e^{-\gamma d}, \quad (4.25)$$

onde  $I_0$  corresponde à intensidade original de luz e  $\gamma$  denota o coeficiente de absorção da luz.

Como a atratividade entre os vaga-lumes é proporcional à intensidade de luz vista pelos vaga-lumes vizinhos, podemos então definir a atratividade  $\beta$  de um vaga-lume por:

$$\beta = \beta_0 e^{-\gamma d^2}, \quad (4.26)$$

onde  $\beta_0$  corresponde à atratividade quando  $d = 0$ .

Temos, então, que o vaga-lume  $i$  é atraído a outro vaga-lume  $k$  com uma função objetivo melhor, movendo-se de acordo com a seguinte equação:

$$x_i^j(t+1) = x_i^j(t) + \beta_0 e^{-\gamma d_{i,k}^2} (x_k^j - x_i^j) + \phi(\sigma_i - \frac{1}{2}), \quad (4.27)$$

onde o segundo termo denota a atração entre ambos vaga-lumes,  $d_{i,k}^2$  corresponde à distância entre os vaga-lumes  $i$  e  $k$ ,  $\phi$  é um fator de randomização,  $\sigma_i \sim U(0, 1)$  e  $j$  denota a variável de otimização.

#### 4.9 Algoritmo dos Morcegos - BA

A ecolocalização é uma sofisticada capacidade biológica utilizada para determinar a distância e/ou posição de obstáculos e/ou animais em um ambiente através da emissão de ondas ultrassônicas e analisar o tempo gasto para que as ondas emitidas, reflitam no alvo e voltem à fonte sobre a forma de eco. Os morcegos têm este sentido bem desenvolvido, pois são animais de

hábitos noturnos e não possuem uma boa visão. Nos morcegos, os pulsos são emitidos pela boca e narinas. Com base no tempo que os ecos demoram a voltar os morcegos detectam presas e obstáculos em seu caminho.

Inspirado nesse fenômeno, Yang [10] propôs um algoritmo Metaheurística, o Algoritmo dos morcegos (BA), se comporta como uma população de morcegos procurando por alimento usando sua capacidade de ecolocalização.

Cada morcego  $i$  é modelado com uma posição  $\vec{x}_i \in \mathbb{R}^n$ , velocidade  $\vec{v}_i \in \mathbb{R}^n$  e frequência  $f_i$ . Para cada iteração  $t$ , como  $t = 1, \dots, M$ ,  $M$  é o número máximo de iteração, o movimento do morcego  $i$  na dimensão  $j$  é dado por atualizar a velocidade e posição usando as seguintes equações:

$$f_i = f_{min} + (f_{max} - f_{min})\beta, \quad (4.28)$$

$$v_i^j(t) = v_i^j(t-1) + (\hat{x} - x_i^j(t-1))f_i, \quad (4.29)$$

$$x_i^j(t) = x_i^j(t-1) + v_i^j(t), \quad (4.30)$$

onde  $\beta$  é um número randômico que  $\in [0, 1]$ . O resultado da equação de  $f_i$  (Equação 4.28) é usada para controlar a velocidade e extensão do movimento do morcego. A variável  $\hat{x}$  representa a solução corrente do melhor global, e  $f_{min}$  e  $f_{max}$  significa os valores máximo e mínimos de frequência, que são especificados pelo usuário.

O Algoritmo dos Morcegos trabalha semelhante ao algoritmo de Otimização por Enxames de Partículas (ver Seção 4.1) como  $f_i$  essencialmente controla o alcance do movimento dos morcegos. Para mitigar possíveis soluções locais ao redor do máximo conhecido global, o Algoritmo dos Morcegos emprega um passeio aleatório.

Basicamente, uma solução é selecionada entre as melhores soluções correntes, em seguida, o passeio aleatório é aplicado localmente e gera uma nova solução para cada morcego:

$$x_{new} = x_{old} + \epsilon \bar{A}(t), \quad (4.31)$$

em que  $\bar{A}(t)$  é a média de volume de todos os morcegos na iteração  $t$ , e  $\epsilon \in [-1, 1]$  é um número randômico. Para cada iteração do algoritmo, o volume  $A_i$  e a taxa de emissão  $r_i$  são atualizados como segue:

$$A_i(t+1) = \alpha A_i(t), \quad (4.32)$$

e

$$r_i(t + 1) = r_i(0)[1 - e^{-\gamma t}], \quad (4.33)$$

onde  $\alpha$  e  $\gamma$  são constantes. Na primeira iteração do algoritmo a taxa de emissão  $r_i(0)$  e o volume  $A_i(0)$  para cada morcego devem ser diferentes e randomicamente escolhidos. Por exemplo,  $A_i(0) \in [1, 2]$  e  $r_i(0) \in [0, 1]$ . Entretanto, o volume e as taxas de emissão são atualizadas somente se novas soluções são melhoradas que significa que esses morcegos estão se movendo ao redor da solução ótima.



## 5 Metodologia

Nesta seção é discutida a metodologia empregada no presente trabalho. O algoritmo de restauração baseado em POCS é dado por:

$$f^{(k+1)} = P_{C_{LA}} P_{C_{RAP}} f^{(k)}, \quad (5.1)$$

em que  $P_{C_{RAP}}$  e  $P_{C_{LA}}$  denotam as projeções no conjunto RAP (Seção 3) e AL (Seção 3.1), respectivamente.

A imagem degradada é primeiro projetada no conjunto  $P_{C_{RAP}}$ , que significa resolver a Equação 3.2 iterativamente. A imagem é então restaurada de acordo com algum valor de  $\lambda$ . Em seguida, o conjunto de restrição  $P_{C_{LA}}$  funciona como um limite inferior e superior para esta imagem restaurada, reduzindo os valores para dentro da faixa de 256 tons de cinza, já que as imagens possuem 8bits/pixel. Esta etapa caracteriza uma iteração do algoritmo POCS (foi empregada a versão RAP proposta por Samaddar e Mammone [2]). Neste trabalho foi modelada a tarefa de encontrar o valor de  $\lambda$  (Equação 3.2) como um problema de otimização, dado que o resultado da restauração da imagem depende fortemente da escolha deste parâmetro de relaxação. Uma vez que podem existir imagens com milhares de pixels, torna-se impraticável realizar uma busca exautiva para  $\lambda$ . A fim de orientar o algoritmo de otimização no processo de restauração, Papa et al. [3] propuseram o uso de algumas métricas de qualidade de imagem como função de ajuste. No contexto de restauração de imagens, pode-se avaliar se um algoritmo obteve resultados adequados usando *Improvement Signal to Noise Ratio* (ISNR) [9] e *Universal Image Quality Index* (UIQI) [19], por exemplo, onde  $f$ ,  $g$ ,  $\hat{f}$  são, respectivamente, a imagem original, degradada e restaurada. A métrica de qualidade ISNR é dada por:

$$ISNR = 10 \log_{10} \left[ \frac{(g - f)^2}{(\hat{f} - f)^2} \right]. \quad (5.2)$$

Em relação a UIQI, a mesma é dada por:

$$UIQI = \frac{4\sigma_{f\hat{f}}f_m\hat{f}_m}{(\sigma_f^2 + \sigma_{\hat{f}}^2)(f_m^2 + \hat{f}_m^2)}, \quad (5.3)$$

onde  $f_m$  e  $\hat{f}_m$  significa a média do brilho dos *pixels* da imagem original e restaurada respectivamente. Tem-se, ainda que:

$$\sigma_{f\hat{f}} = \frac{1}{N} (f - f_m)^2 (\hat{f} - \hat{f}_m)^2, \quad (5.4)$$

$$\sigma_f = \frac{1}{N}(f - f_m), \quad (5.5)$$

$$\sigma_{\hat{f}} = \frac{1}{N}(\hat{f} - \hat{f}_m)^2, \quad (5.6)$$

onde  $N$  significa o número de pixels da imagem.

O intervalo de variação do  $UIQI$  é  $[-1, 1]$ , onde o melhor valor  $UIQI = 1$  é obtido somente quando a imagem restaurada é igual a imagem original. Neste trabalho nós propomos comparar várias técnicas metaheurísticas contra a técnica PSO abordada por [3], uma vez que o PSO é geralmente mais custoso computacionalmente. Ademais as técnicas metaheurísticas já apresentadas nunca foram utilizadas em conjunto com POCS em tarefas de restauração de imagens. Em um primeiro momento, um espaço de busca é preenchido por agentes posicionados randomicamente, sendo que existe uma interação entre esses agentes a fim de conduzir ao máximo valor de ISNR ou  $UIQI$ . Em relação ao problema de otimização, ele é composto por apenas uma variável de decisão,  $\lambda$ , que é o parâmetro de relaxação do conjunto de restrição RAP. Assim sendo, temos um espaço de busca de uma dimensão.

A idéia do algoritmo de restauração é a seguinte: dada uma imagem original  $f$ , podemos corrompê-la com borramento e ruído para obter  $g$  de acordo com a Equação 2.1. Em seguida, inicializamos todos os agentes com valores que estão entre o intervalo de  $[0, 2]$  (esses valores são normalmente utilizados para  $\lambda$  de acordo com Samaddar e Mammone [2]). O valor de cada agente é usado como  $\lambda$  na Equação 3.2, a qual calcula a projeção da imagem atual restaurada no conjunto de restrição RAP. Além disso, a imagem resultante é projetada em seguida sobre o conjunto de amplitude limitada. Para isso, usamos  $\alpha = 0$  e  $\beta = 255$  (caso o valor do *pixel* seja maior que 255, ele se torna 255; e caso o valor do *pixel* seja menor que 0, o valor se torna 0). Esse conjunto de passos corresponde a uma iteração  $k$  da Equação 5.1, sendo que uma imagem restaurada parcial  $\hat{f}^{(k+1)}$  é utilizada como entrada para a próxima iteração da técnica POCS. No final do algoritmo de restauração, podemos computar os valores de ISNR e  $UIQI$ . Os resultados das métricas serão utilizados para mover os agentes para novas posições que maximizam a função objetivo.

O valor de  $\lambda$  encontrado é então empregado para restaurar outra imagem degradada com as mesmas especificações de borramento e ruído. Assim, podemos obter  $\lambda$  com uma imagem de treinamento (imagem protótipo), e em seguida usá-lo para restaurar outra imagem com processo de degradação semelhante.

A Figura 5.1 mostra esse procedimento.

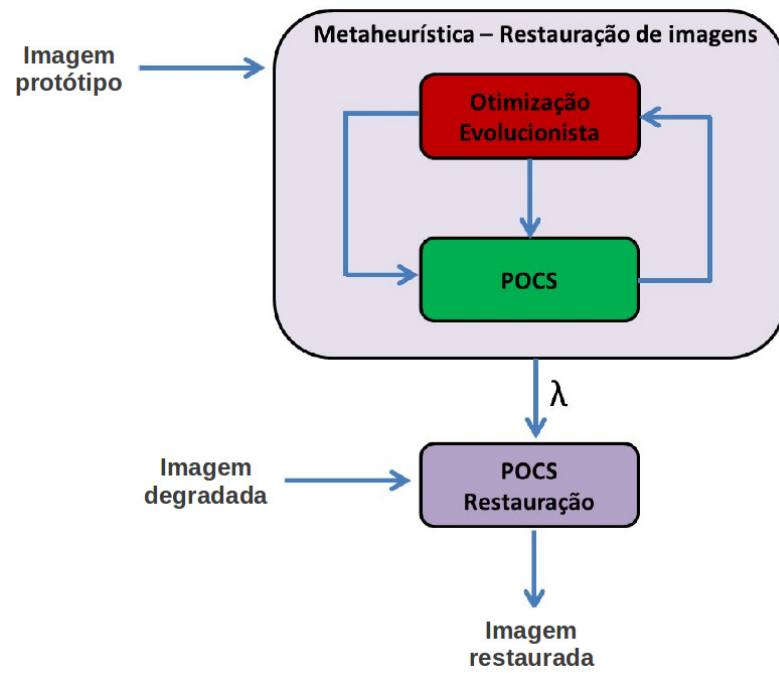


Figura 5.1. Modelo proposto para restauração de imagens.

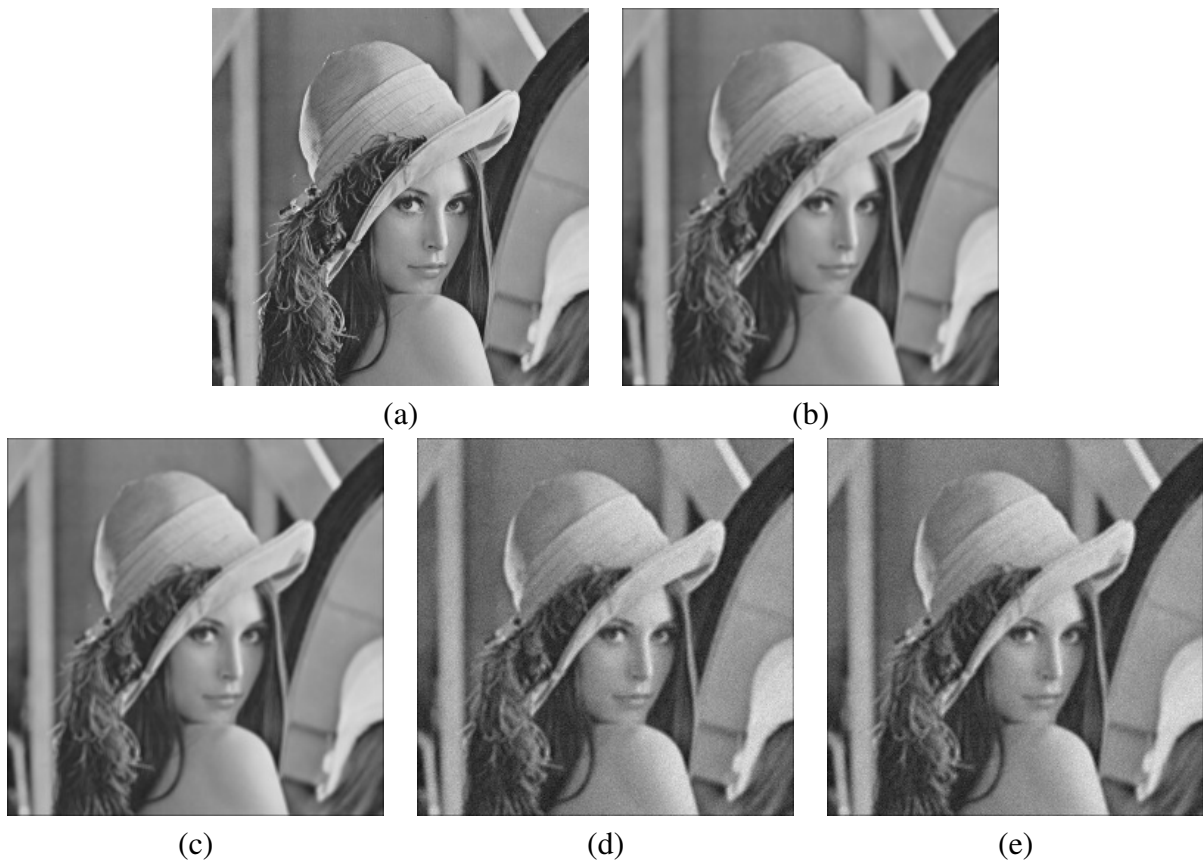
## 6 Resultados Experimentais

Os experimentos conduzidos mostram a robustez e eficiência das diferentes técnicas metaheurísticas para uma escolha adequada dos valores de  $\lambda$ . Os experimentos foram realizados utilizando a imagem Lena (Figura 6.1a), Cameraman (Figure 6.2a), Sensoriamento remoto 1 (Figura 6.3a) e sensoriamento remoto 2 (Figura 6.4a) em quatro situações diferentes: A imagem original foi degradada com borramento gaussiano (Figuras 6.1b, 6.2b, 6.3b e 6.4b), turbulência atmosférica (Figuras 6.1c, 6.2c, 6.3c e 6.4c), borramento gaussiano com ruído gaussiano (Figuras 6.1d, 6.2d, 6.3d e 6.4d) e turbulência atmosférica com ruído gaussiano (Figuras 6.1e, 6.2e, 6.3e e 6.4e).

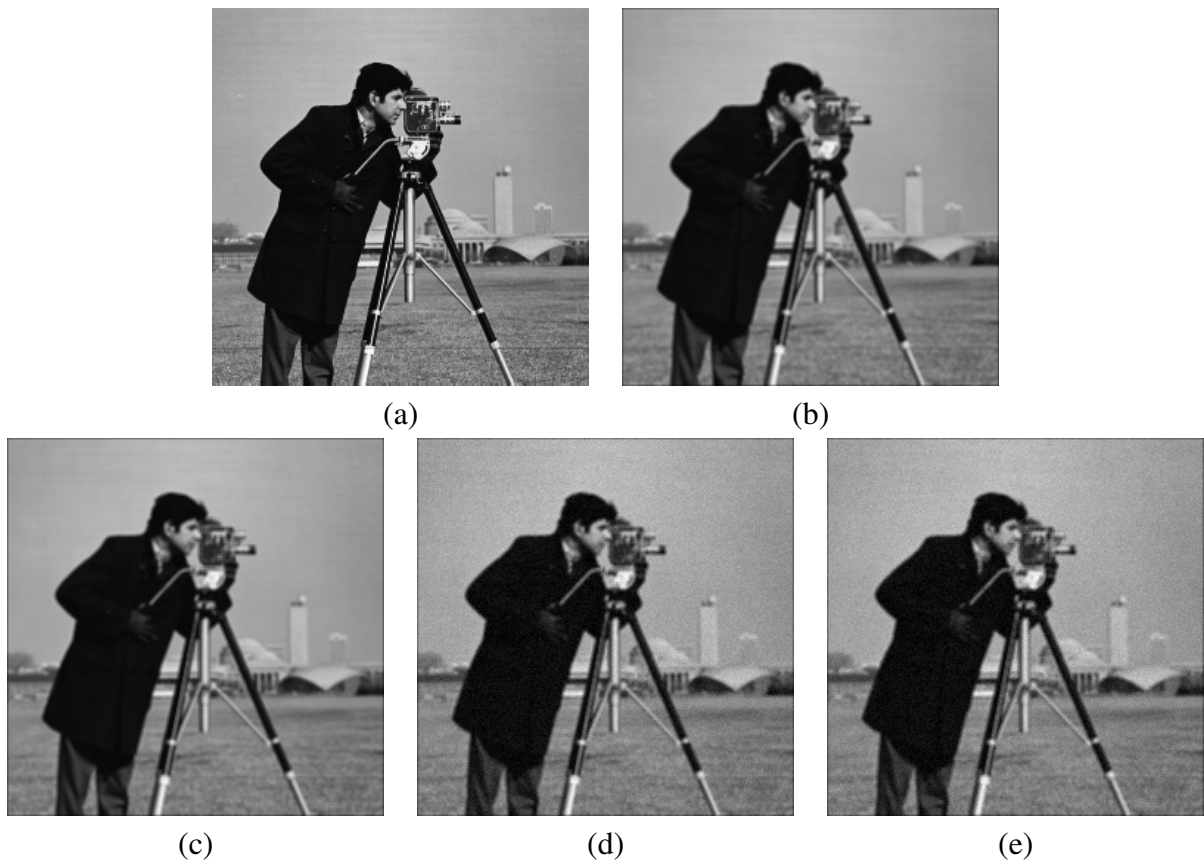
Em relação ao borramento gaussiano, foi empregada uma janela  $3 \times 3$  com  $\sigma = 1.8$  e para turbulência atmosférica foi utilizada uma janela  $3 \times 3$  com  $k = 0.0025$ , a qual significa turbulência atmosférica grave segundo Gonzales [1]. Foi adicionado o ruído gaussiano com média 0 e variância 0,0005. O número de iterações para convergência da técnica POCS foi 6. Esses valores foram empiricamente escolhidos. Para cada imagem, foram conduzidas duas rodadas de experimentos: primeiro uma busca exaustiva foi executada para encontrar o ótimo valor de  $\lambda \in [0, 2]$  que lidera o melhor valor de ISNR e UIQI. Dessa forma, a eficácia das técnicas metaheurísticas em encontrar o valor adequado de  $\lambda$  pode ser avaliado, i.e., estamos interessados em saber o quanto esses valores estão próximos dos valores ótimos encontrados na busca exaustiva. A Tabela 6.1 mostra os parâmetros utilizados para o PSO, HS e suas variantes, FFA, GSA and BA (esses parâmetros foram escolhidos de acordo com a orientação de seus autores e foram os mesmos para todas as imagens).

**Tabela 6.1. Parâmetros utilizados para cada técnica de otimização: o número de agentes utilizados para todas as técnicas é 50. Os parâmetros do PSO estão de acordo com Papa et al. [3].**

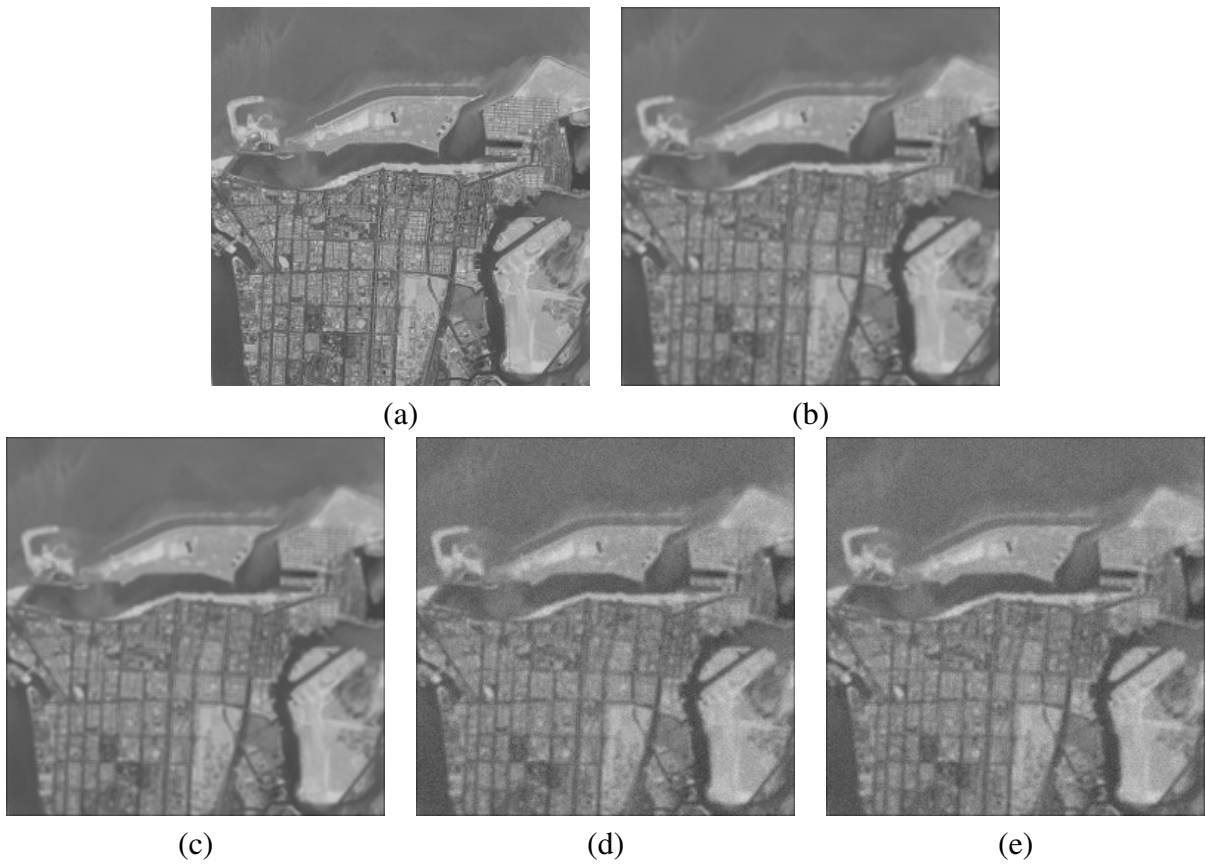
Técnicas	Parâmetros
HS	HMCR= 0.9, PAR= 0.3, BW= 0.01
GHS	HMCR= 0.9, PARmin= 0.01, PARmax= 0.99
IHS	HMCR= 0.9, PARmin= 0.01, PARmax= 0.99 BWmin= 0.0001, BWmax= 0.2
NGHS	$p_m=0.033$
SGHS	HMCRm= 0.98, PARm= 0.9, BWmin= 0.0005 BWmax= 0.4, LP= 2
PSO	$c_1 = c_2 = 2, w = 1.255$
FFA	$\alpha = 0.01, \text{Beta} = 1, \gamma = 0.8$
GSA	$G_0=100, \text{Alpha} = 20$
BA	$\alpha = 0.9, \gamma = 0.9$



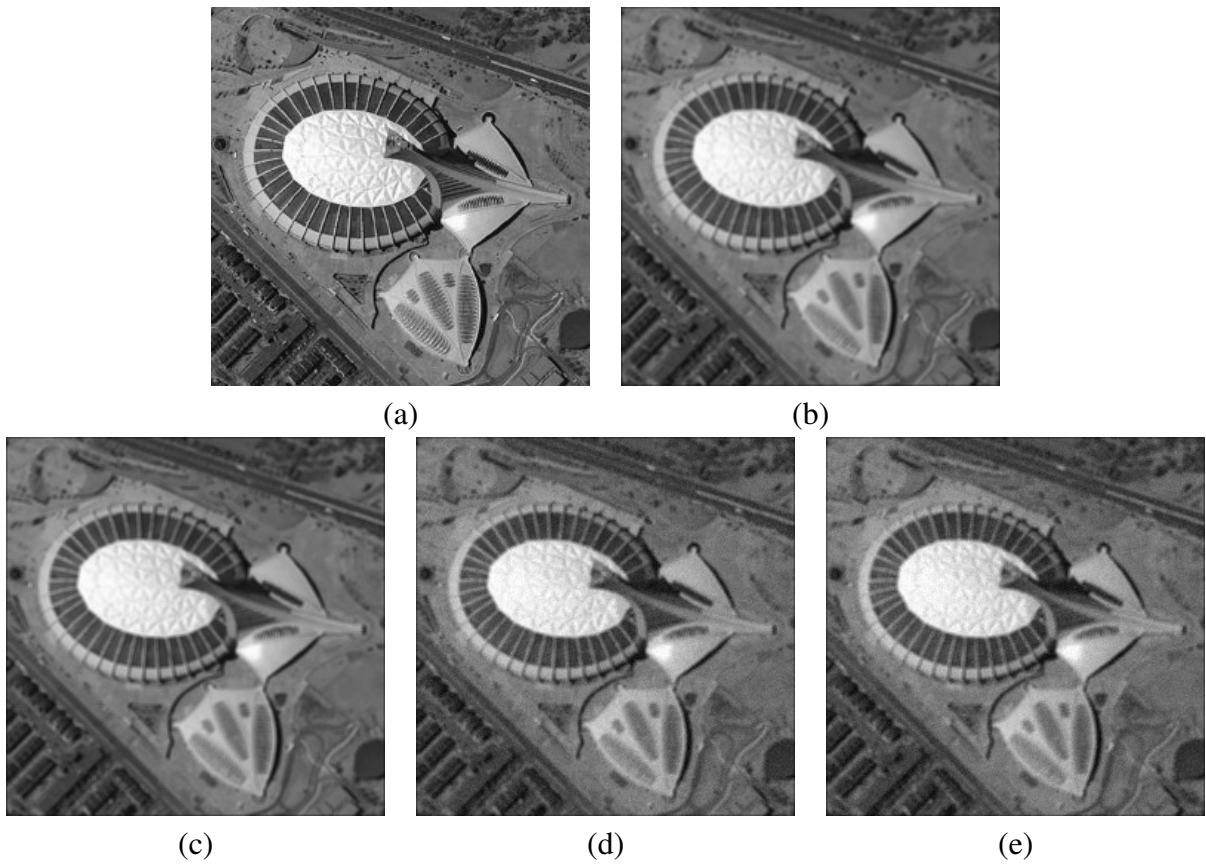
**Figura 6.1. Imagem Lena com  $256 \times 256$  e 8bits/pixel: (a) original, (b) Borrimento Gaussiano, (c) Borrimento por Turbulência Atmosférica, (d) Borrimento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano.**



**Figura 6.2. Imagem Cameraman com  $256 \times 256$  e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano.**



**Figura 6.3. Imagem Sensoriamento remoto 1 com  $256 \times 256$  e 8bits/pixel: (a) original, (b) Borrimento Gaussiano, (c) Borrimento por Turbulência Atmosférica, (d) Borrimento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano.**



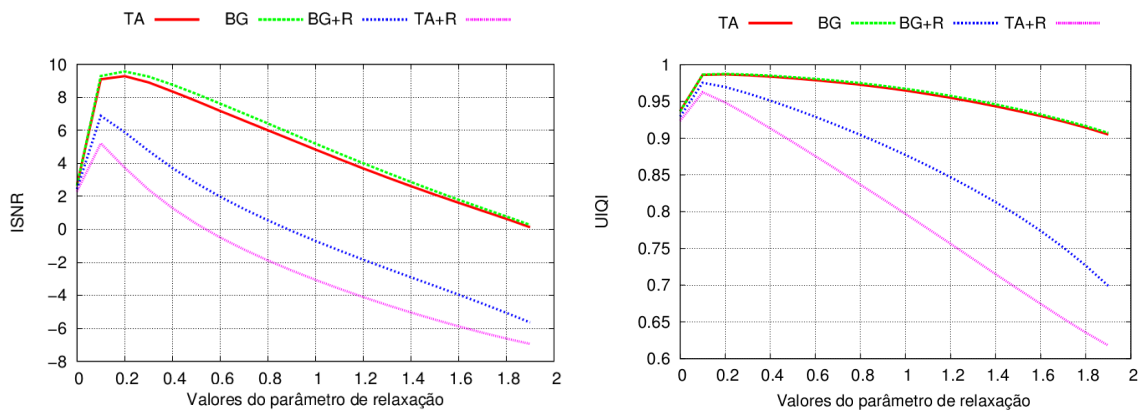
**Figura 6.4. Imagem Sensoriamento remoto 2 com  $256 \times 256$  e 8bits/pixel: (a) original, (b) Borramento Gaussiano, (c) Borramento por Turbulência Atmosférica, (d) Borramento Gaussiano com Ruído Gaussiano e (e) Turbulência Atmosférica com Ruído Gaussiano.**



## 6.1 Experimentos Lena

A primeira rodada de experimentos teve como objetivo executar o algoritmo POCS proposto (Equação 5.1) para restaurar as imagens degradadas da Lena (Figuras 6.1b, 6.1c, 6.1d e 6.1e) usando os valores de  $\lambda$  no intervalo  $[0, 2]$  com passos de 0.1. Esta abordagem exaustiva foi empregada a fim de descobrir os valores de  $\lambda$  que maximizam as funções de ISNR e UIQI.

A Figura 6.5 mostra a curva dos resultados obtidos para borramento gaussiano (BG), borramento por turbulência atmosférica (TA), borramento gaussiano com ruído gaussiano (BG+R) e borramento por turbulência atmosférica com ruído gaussiano (TA+R). Os valores ótimos ,i.e., aqueles que maximizam ISNR e que aproximam UIQI para 1, obtidos considerando ISNR, foram:  $\lambda = 0.2$  para BG com valor da métrica 9.576981,  $\lambda = 0.1$  para BG+R com valor da métrica 6.8816,  $\lambda = 0.2$  para TA com valor da métrica 9.287093  $\lambda = 0.1$  para TA+R com valor da métrica 5.216516. Considerando UIQI, foram  $\lambda = 0.2$  para BG com valor da métrica 0.987547,  $\lambda = 0.1$  para BG+R com valor da métrica 0.975470,  $\lambda = 0.2$  para TA com valor da métrica 0.986777 e  $\lambda = 0.1$  para TA+R com valor da métrica 0.962526.



**Figura 6.5. Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI.**

Foram aplicadas as técnicas PSO, HS, IHS, GHS, NGHS, SGHS, FFA, GSA e BA para encontrar o valor de  $\lambda$  pseudo-ótimo que maximiza as métricas de qualidade de imagem ISNR e UIQI. A ideia é comparar os valores obtidos por essas técnicas com os valores encontrados no experimento anterior (busca exaustiva).

As Tabelas 6.2, 6.3, 6.4 e 6.5 mostram os resultados para ISNR e UIQI (os valores em parênteses são os de  $\lambda$ ). Podemos notar que todas as técnicas encontraram um resultado similar de restauração para todas as situações, mas as técnicas em que os valores chegaram mais próximos daqueles encontrados na busca exaustiva foi o FFA para todos os casos, e somente o PSO para UIQI (TA). Entretanto, se considerarmos o custo computacional, o HS e suas variantes tem executado em 59,10 minutos para todos os experimentos, enquanto PSO, FFA, BA, GSA necessitam, em média, 515.28 minutos. Dessa forma, a técnica HS foi 8.05 vezes mais rápida que as outras técnicas, e com resultados muito próximos de ISNR e UIQI. É importante observar

que todas as técnicas acharam os valores semelhantes de  $\lambda$  obtidos pela busca exaustiva.

**Tabela 6.2. Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG ISNR	BG UIQI
HS	9.568419 (0.15)	0.987564 (0.18)
NGHS	9.589759 (0.17)	0.987571 (0.19)
SGHS	9.584118 (0.16)	0.987528 (0.16)
GHS	9.568419 (0.15)	0.987291 (0.13)
IHS	9.544083 (0.14)	0.987571 (0.19)
PSO	9.594222 (0.177072)	0.987568 (0.193290)
FFA	<b>9.594598 (0.172780)</b>	<b>0.987577 (0.188444)</b>
GSA	9.592121 (0.181385)	0.987570 (0.193099)
BA	9.592019 (0.18)	0.987566 (0.183488)

**Tabela 6.3. Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG+R ISNR	BG+R UIQI
HS	6.923564 (0.07)	0.975704 (0.08)
NGHS	6.553789 (0.14)	0.975470 (0.1)
SGHS	6.942856 (0.08)	0.975704 (0.08)
GHS	6.857821 (0.06)	0.975659 (0.09)
IHS	6.881600 (0.1)	0.975659 (0.09)
PSO	6.939488 (0.084449)	0.975724 (0.084694)
FFA	<b>6.943174 (0.080100)</b>	<b>0.975735 (0.084386)</b>
GSA	6.942681 (0.078905)	0.975717 (0.085582)
BA	6.94285 (0.08)	0.975733 (0.083622)

**Tabela 6.4. Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	TA ISNR	TA UIQI
HS	9.320056 (0.15)	0.986847 (0.16)
NGHS	9.305294 (0.19)	0.986772 (0.14)
SGHS	9.302682 (0.14)	0.986853 (0.18)
GHS	9.320056 (0.15)	0.986853 (0.18)
IHS	9.320056 (0.15)	0.986824 (0.15)
PSO	9.328550 (0.160435)	<b>0.986861 (0.172542)</b>
FFA	<b>9.329515 (0.162103)</b>	0.986860 (0.179744)
GSA	9.327530 (0.166338)	0.986860 (0.167851)
BA	9.329169 (0.16)	0.986853 (0.18)

**Tabela 6.5. Resultados de restauração para imagens da Lena considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	TA+R ISNR	TA+R UIQI
HS	5.216516 (0.1)	0.964783 (0.07)
NGHS	5.570081 (0.06)	0.963762 (0.04)
SGHS	5.557183 (0.05)	0.960215 (0.12)
GHS	5.557183 (0.05)	0.964783 (0.07)
IHS	5.203150 (0.03)	0.953289 (0.17)
PSO	5.572796 (0.054284)	0.964967 (0.054284)
FFA	<b>5.574383 (0.056439)</b>	<b>0.965037 (0.059458)</b>
GSA	5.570081 (0.06)	0.964783 (0.07)
BA	5.570081 (0.06)	0.964952 (0.065051)

A Figura 6.6 apresenta as imagens restauradas com os valores de  $\lambda$  obtido pelo melhor resultado de cada experimento considerando ISNR: FFA para todos os casos.



**Figura 6.6. Imagens restauradas da Lena considerando os ótimos  $\lambda$  encontrados e ISNR: (a) FFA para BG, (b) FFA para TA (c) FFA para BG+R e (d) FFA para TA+R.**

A Figura 6.7 apresenta as imagens restauradas com os valores de  $\lambda$  obtido pelo melhor resultado de cada experimento considerando UIQI: FFA para todos os casos somente PSO para TA.

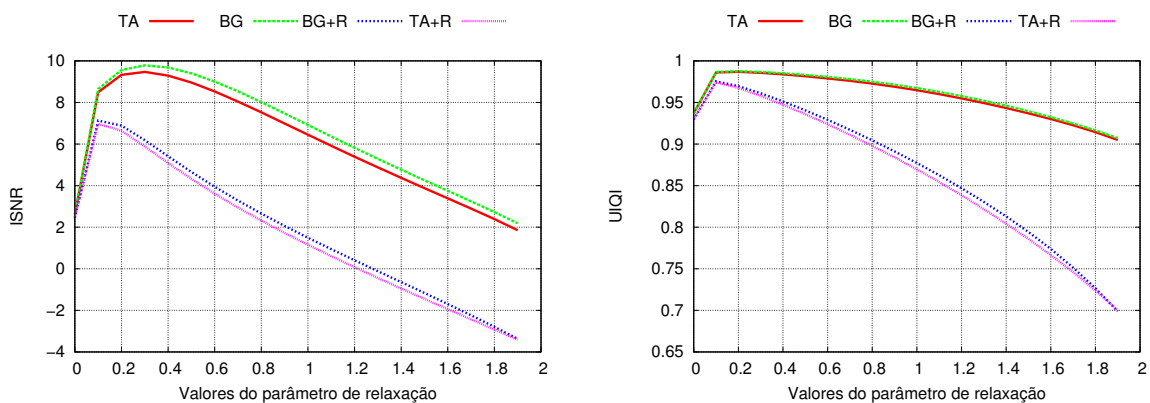


**Figura 6.7. Imagens restauradas da Lena considerando os ótimos  $\lambda$  encontrados e UIQI: (a) FFA para BG, (b) PSO para TA (c) FFA para BG+R e (d) FFA para TA+R.**

## 6.2 Experimentos Cameraman

Esta rodada de experimentos teve como objetivo executar o algoritmo POCS proposto (Equação 5.1) para restaurar as imagens degradadas do Cameraman (Figuras 6.2b, 6.2c, 6.2d e 6.2e) usando os valores de  $\lambda$  no intervalo  $[0, 2]$  com passos de 0.1 (este procedimento é o mesmo que na seção anterior, i.e., com as imagens da Lena). Esta abordagem exaustiva foi empregada a fim de descobrir os valores de  $\lambda$  que maximizam as funções de ISNR e UIQI.

A Figura 6.8 mostra a curva dos resultados obtidos para borramento gaussiano (BG), borramento por turbulência atmosférica (TA), borramento gaussiano com ruído gaussiano (BG+R) e borramento por turbulência atmosférica com ruído gaussiano (TA+R). Os valores ótimos obtidos considerando ISNR foram  $\lambda = 0.3$  para BG com valor da métrica 9.784966,  $\lambda = 0.1$  para BG+R com valor da métrica 7.127350,  $\lambda = 0.3$  para TA com valor da métrica 9.470736 e  $\lambda = 0.1$  para TA+R com valor da métrica 6.974157. Considerando UIQI, foram  $\lambda = 0.3$  para BG com valor da métrica 0.989249,  $\lambda = 0.1$  para BG+R com valor da métrica 0.979397,  $\lambda = 0.3$  para TA com valor da métrica 0.988594 e  $\lambda = 0.1$  para TA+R com valor da métrica 0.978794.



**Figura 6.8. Busca exaustiva utilizando o algoritmo proposto para as métricas de qualidade de imagem (a) ISNR e (b) UIQI.**

Foram aplicadas as técnicas PSO, HS, IHS, GHS, NGHS, SGHS, FFA, GSA e BA para encontrar o valor de  $\lambda$  ótimo que maximiza as métricas de qualidade de imagem ISNR e UIQI. A ideia é comparar os valores obtidos por essas técnicas com os valores encontrados no experimento anterior (busca exaustiva). As Tabelas 6.6, 6.7, 6.8 e 6.9 mostram os resultados para ISNR e UIQI (os valores em parenteses são os de  $\lambda$ ). Podemos ver que todas as técnicas encontraram um resultado similar de restauração para todas as situações. Entretanto, se considerarmos o custo computacional, obteve valores muito próximos do experimento 6.1. É importante observar que todas as técnicas acharam os valores semelhantes de  $\lambda$  obtidos pela busca exaustiva.

**Tabela 6.6. Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG ISNR	BG UIQI
HS	9.784966 (0.3)	0.989121 (0.25)
NGHS	9.778504 (0.33)	0.989249 (0.3)
SGHS	9.782567 (0.29)	0.989258 (0.28)
GHS	9.761069 (0.27)	0.989227 (0.31)
IHS	9.783403 (0.31)	0.989283 (0.29)
PSO	9.785612 (0.299049)	0.989291 (0.294077)
FFA	<b>9.786348 (0.302802)</b>	<b>0.989294 (0.293525)</b>
GSA	9.784004 (0.310976)	0.989280 (0.289577)
BA	9.784966(0.3)	0.989274 (0.295264)

**Tabela 6.7. Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG+R ISNR	BG+R UIQI
HS	7.164250 (0.12)	0.979599 (0.13)
NGHS	7.117686 (0.15)	0.979133 (0.09)
SGHS	7.083639 (0.159945)	0.979456 (0.15)
GHS	7.143272 (0.14)	0.978912 (0.18)
IHS	7.143272 (0.14)	0.979591 (0.12)
PSO	7.164554 (0.122701)	0.979599 (0.13)
FFA	<b>7.165692 (0.117706)</b>	<b>0.979621 (0.126113)</b>
GSA	7.159225 (0.13)	0.979597 (0.129215)
BA	7.164250 (0.12)	0.979591 (0.12)

**Tabela 6.8. Resultados de restauração para imagens Cameraman, considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

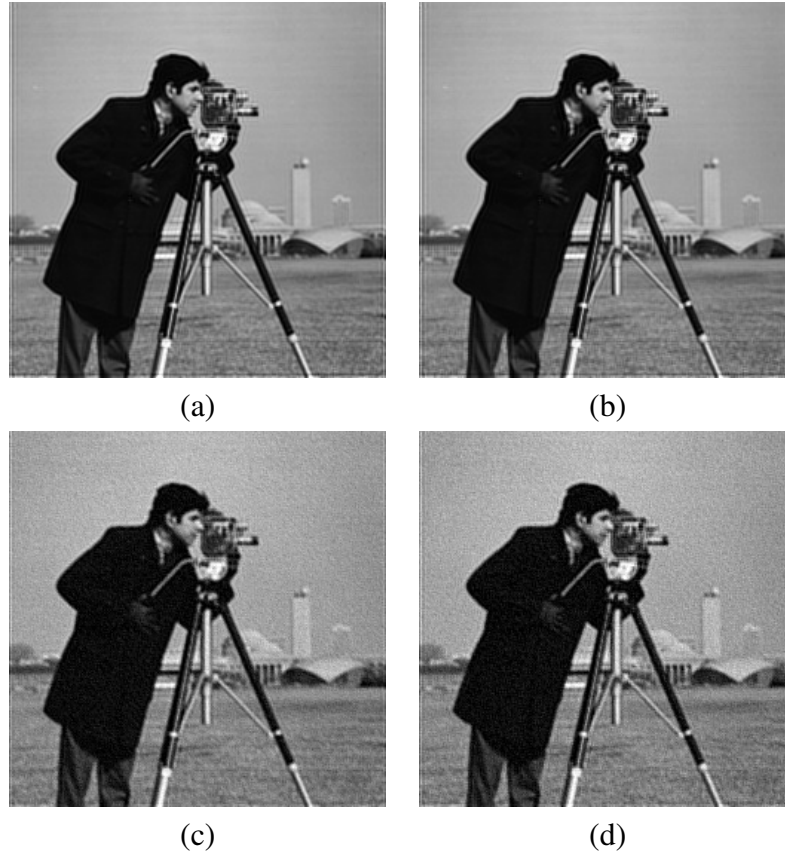
Técnicas	TA ISNR	TA UIQI
HS	9.473675 (0.29)	0.988583 (0.28)
NGHS	9.461668 (0.31)	0.988578 (0.29)
SGHS	9.470736 (0.3)	0.988583 (0.28)
GHS	9.476247 (0.28)	0.988510 (0.34)
IHS	9.473675 (0.29)	0.988525 (0.33)
PSO	9.476247 (0.28)	0.988589 (0.295778)
FFA	<b>9.476610 (0.2793)</b>	<b>0.988596 (0.276220)</b>
GSA	9.470736 (0.3)	0.988594 (0.3)
BA	9.474569 (0.276136)	0.988594 (0.3)

**Tabela 6.9. Resultados de restauração para imagens Cameraman considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	TA+R ISNR	TA+R UIQI
HS	6.867662 (0.08)	0.978894 (0.12)
NGHS	6.988524 (0.12)	0.978630 (0.15)
SGHS	6.934617 (0.09)	0.978894 (0.12)
GHS	6.973883 (0.13)	0.978867 (0.11)
IHS	6.989814 (0.11)	0.978794 (0.1)
PSO	<b>6.992042 (0.112026)</b>	0.978894 (0.115704)
FFA	6.929259 (0.145605)	<b>0.978904 (0.114598)</b>
GSA	6.990329 (0.118178)	0.978876 (0.111777)
BA	6.991446 (0.111868)	0.978891 (0.113306)

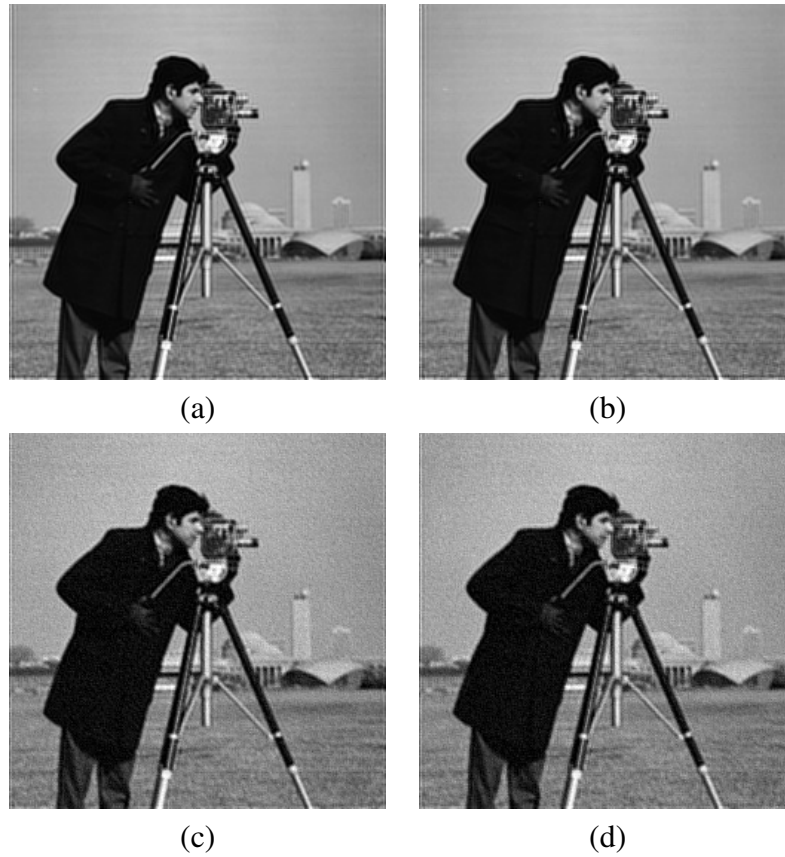


A Figura 6.9 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando ISNR: FFA para todos os casos, somente PSO para TA+R.



**Figura 6.9. Imagens restauradas do Cameraman considerando os ótimos  $\lambda$  encontrados e ISNR: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) PSO para TA+R.**

A Figura 6.10 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando UIQI: FFA para todos os casos.



**Figura 6.10. Imagens restauradas do Cameraman considerando os ótimos  $\lambda$  encontrados e UIQI: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

### 6.3 Aprendendo o valor de $\lambda$ - Experimento 1

A ideia desse experimento é mostrar que é possível aprender o valor de  $\lambda$  com uma imagem e restaurar outra com esse mesmo  $\lambda$  (o domínio das imagens não pode ser diferente). A técnica que encontrou na maioria das vezes os melhores valores de  $\lambda$  nas diferentes imagens degradadas da Lena e Cameraman foi o FFA. Os experimentos foram realizados como segue: Os valores de  $\lambda$  encontrados nas imagens da Lena foram usados para restaurar as imagens do Cameraman (Figuras 6.2b, 6.2c, 6.2d e 6.2e). De acordo com as Tabelas 6.10, 6.11, 6.12 e 6.13, podemos observar que os resultados da restauração (métricas), são muito semelhantes aos resultados das métricas encontrados nas imagens do Cameraman.

**Tabela 6.10. Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para BG (os valores em parenteses são os de  $\lambda$ ).**

Técnica	BG ISNR	BG UIQI
FFA	9.407320(0.172780)	0.988498(0.188444)

**Tabela 6.11. Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para BG+R (os valores em parenteses são os de  $\lambda$ ).**

Técnica	BG+R ISNR	BG+R UIQI
FFA	7.001468 (0.080100)	0.978891 (0.084386)

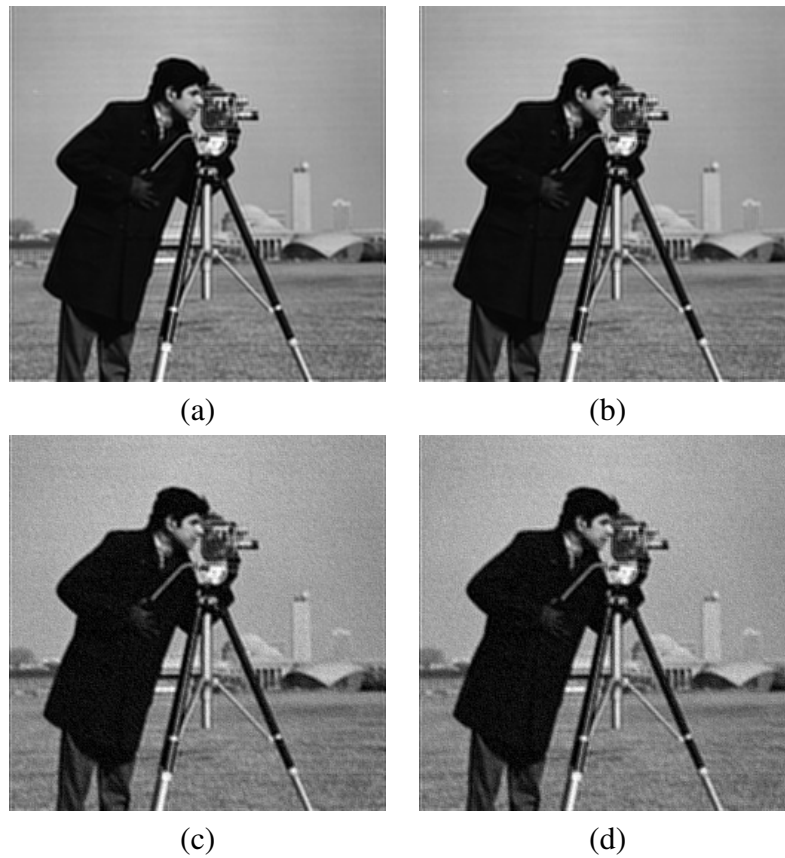
**Tabela 6.12. Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para TA (os valores em parenteses são os de  $\lambda$ ).**

Técnica	TA ISNR	TA UIQI
FFA	9.132328 (0.162103)	—
PSO	—	0.987819 (0.172542)

**Tabela 6.13. Resultados de restauração para as imagens do Cameraman considerando ISNR e UIQI para TA+R (os valores em parenteses são os de  $\lambda$ ).**

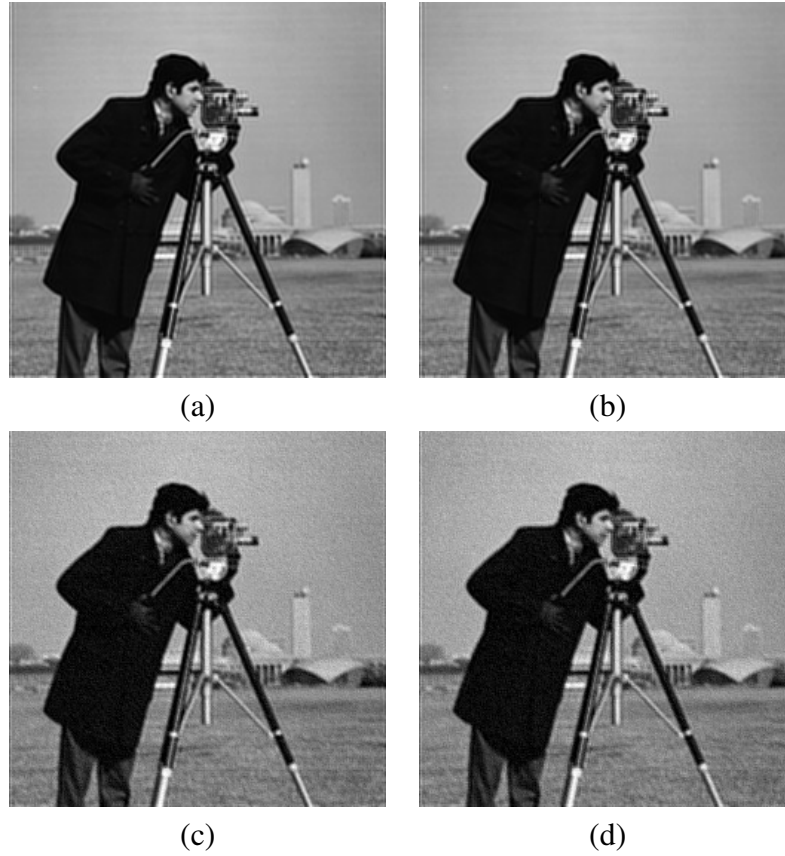
Técnica	TA+R ISNR	TA+R UIQI
FFA	6.537885 (0.056439)	0.976612 (0.059458)

A Figura 6.11 apresenta as imagens restauradas do Cameraman com os melhores  $\lambda$  encontrados nos experimentos das imagens da Lena considerando ISNR: FFA obteve o melhor desempenho para todos os casos.



**Figura 6.11. Imagens restauradas do Cameraman usando os melhores  $\lambda$  obtidos nas imagens da Lena considerando a métrica ISNR: (a) FFA para BG (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

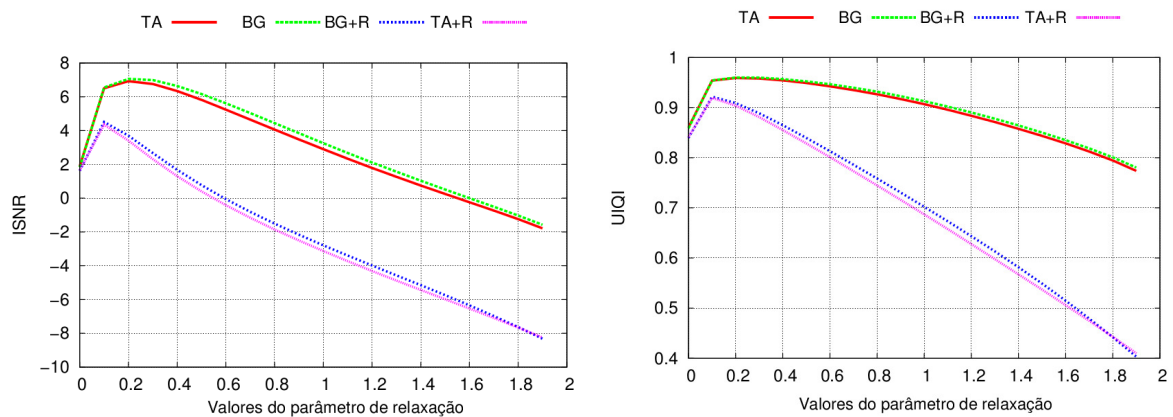
A Figura 6.12 apresenta as imagens restauradas do Cameraman com os melhores  $\lambda$  encontrados nos experimentos das imagens da Lena considerando UIQI: FFA para todos os casos, somente PSO para TA UIQI.



**Figura 6.12.** Imagens restauradas do Cameraman usando os melhores  $\lambda$  obtidos nas imagens da Lena considerando a métrica UIQI: (a) FFA para BG, (b) PSO para TA, (c) FFA para BG+R e (d) FFA para TA+R.

#### 6.4 Experimentos Sensoriamento Remoto - Imagem 1

Esta rodada de experimentos teve como objetivo executar o algoritmo POCS proposto (Equação 5.1) para restaurar as imagens degradadas do Sensoriamento Remoto 1 (Figuras 6.3b, 6.3c, 6.3d e 6.3e) usando os valores de  $\lambda$  no intervalo  $[0, 2]$  com passos de 0.1. (Este procedimento é o mesmo que nas seções anteriores, i.e., com as imagens Cameraman e Lena). Esta abordagem exaustiva foi empregada a fim de descobrir os valores de  $\lambda$  que maximizam as funções de ISNR e UIQI. A Figura 6.13 mostra a curva dos resultados obtidos para borramento gaussiano (BG), borramento por turbulência atmosférica (TA), borramento gaussiano com ruído gaussiano (BG+R) e borramento por turbulência atmosférica com ruído gaussiano (TA+R). Os valores ótimos obtidos considerando ISNR foram  $\lambda = 0.2$  para BG com valor da métrica 7.061809,  $\lambda = 0.1$  para BG+R com valor da métrica 4.512799,  $\lambda = 0.2$  para TA com valor da métrica 6.925353 e  $\lambda = 0.1$  para TA+R com valor da métrica 4.348913. Considerando UIQI, foram  $\lambda = 0.2$  para BG com valor da métrica 0.960047,  $\lambda = 0.1$  para BG+R com valor da métrica 0.921872,  $\lambda = 0.2$  para TA com valor da métrica 0.959119 e  $\lambda = 0.1$  para TA+R com valor da métrica 0.919481.



**Figura 6.13. Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI**

Foram aplicadas as técnicas PSO, HS, IHS, GHS, NGHS, SGHS, FFA, GSA e BA para encontrar o valor de  $\lambda$  pseudo-ótimo que maximiza as métricas de qualidade de imagem ISNR e UIQI. A ideia é comparar os valores obtidos por essas técnicas com os ótimos valores encontrados no experimento anterior (busca exaustiva). As Tabelas 6.14, 6.15, 6.16 e 6.17 mostram os resultados quantitativos para ISNR e UIQI (os valores em parênteses são os de  $\lambda$ ). Pode ser notado que todas as técnicas encontraram um resultado similar de restauração para todas as situações, se considerarmos o custo computacional o HS foi semelhante ao experimento anterior. É importante observar que ambos os HS, FFA, GSA, BA e PSO encontraram valores de  $\lambda$  similares ao encontrado na busca exaustiva.

**Tabela 6.14. Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG ISNR	BG UIQI
HS	7.076010 (0.22)	0.960277 (0.22)
NGHS	7.077588 (0.23)	0.960251 (0.27)
SGHS	7.072010 (0.21)	0.960342 (0.25)
GHS	7.076010 (0.22)	0.960342 (0.25)
IHS	7.076010 (0.22)	0.960404 (0.24)
PSO	7.078755 (0.2272)	0.960404 (0.239377)
FFA	<b>7.080752 (0.2251)</b>	<b>0.960425 (0.241399)</b>
GSA	7.076214 (0.24)	0.960404 (0.24)
BA	7.078483 (0.22899)	0.960404 (0.24)

**Tabela 6.15. Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG+R ISNR	BG+R UIQI
HS	4.548818 (0.07)	0.922128 (0.09)
NGHS	4.503212 (0.06)	0.920138 (0.06)
SGHS	4.560532 (0.08)	0.917799 (0.05)
GHS	4.403119 (0.12)	0.922128 (0.09)
IHS	4.463554 (0.11)	0.922055 (0.08)
PSO	<b>4.559663 (0.074997)</b>	0.922141 (0.086818)
FFA	4.315814 (0.132400)	<b>0.922145 (0.082745)</b>
GSA	4.548818 (0.07)	0.922055 (0.08)
BA	4.556717 (0.072717)	0.922128 (0.09)

**Tabela 6.16. Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

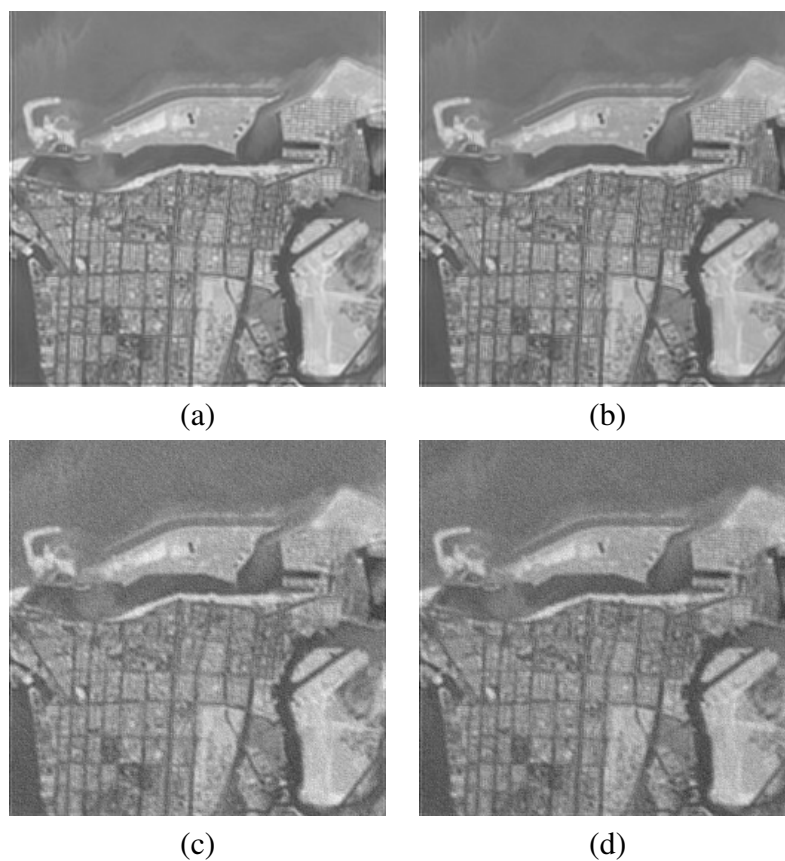
Técnicas	TA ISNR	TA UIQI
HS	6.924005 (0.22)	0.959208 (0.23)
NGHS	6.925353 (0.2)	0.959217 (0.22)
SGHS	6.920197 (0.19)	0.959119 (0.2)
GHS	6.904992 (0.24)	0.959218 (0.21)
IHS	6.904992 (0.24)	0.959217 (0.22)
PSO	6.929440 (0.21)	0.959237 (0.21772)
FFA	<b>6.930244 (0.211831)</b>	<b>0.959242 (0.212698)</b>
GSA	6.929440 (0.21)	0.959226 (0.22509)
BA	6.925353 (0.2)	0.959208 (0.23)

**Tabela 6.17. Resultados de restauração para imagens Sensoriamento Remoto 1 considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	TA+R ISNR	TA+R UIQI
HS	4.395118 (0.06)	0.906015 (0.19)
NGHS	4.428479 (0.07)	0.919966 (0.09)
SGHS	4.311590 (0.05)	0.919481 (0.1)
GHS	3.406391 (0.2)	0.920090 (0.08)
IHS	4.033407 (0.14)	0.918624 (0.06)
PSO	4.431266 (0.074619)	0.920086 (0.084694)
FFA	<b>4.431734 (0.072891)</b>	<b>0.920108 (0.081719)</b>
GSA	4.428092 (0.069443)	0.920090 (0.08)
BA	4.428479 (0.07)	0.920083 (0.085026)

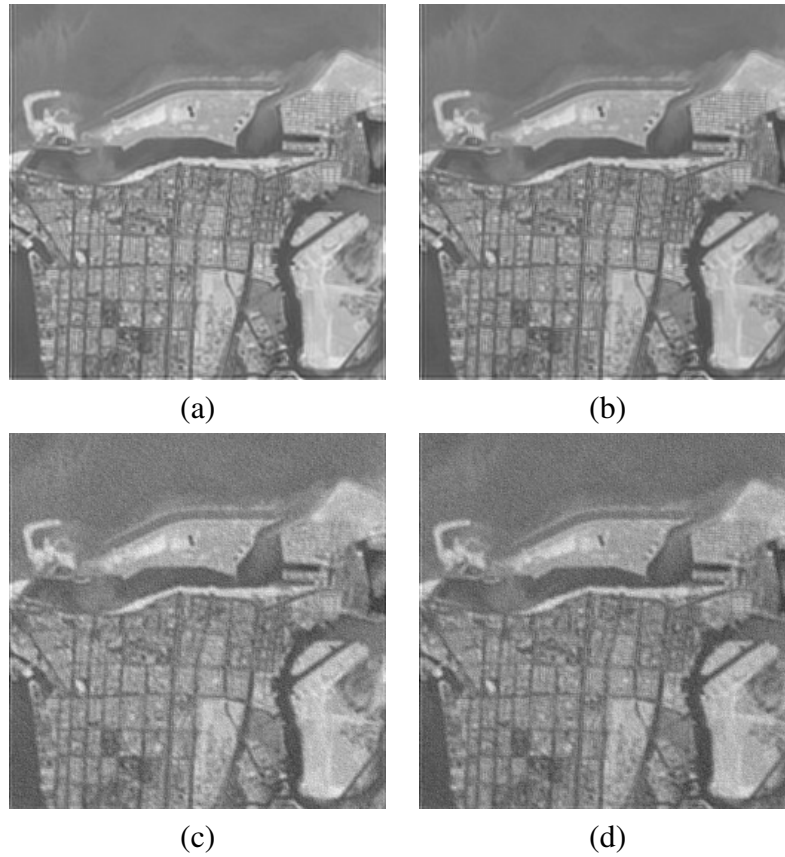


A Figura 6.14 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando ISNR: FFA para todos os casos somente PSO para BG+R.



**Figura 6.14. Imagens restauradas Sensoriamento Remoto 1 usando os ótimos  $\lambda$  encontrados (ISNR): (a) FFA para BG, (b) FFA para TA, (c) PSO para BG+R e (d) FFA para TA+R.**

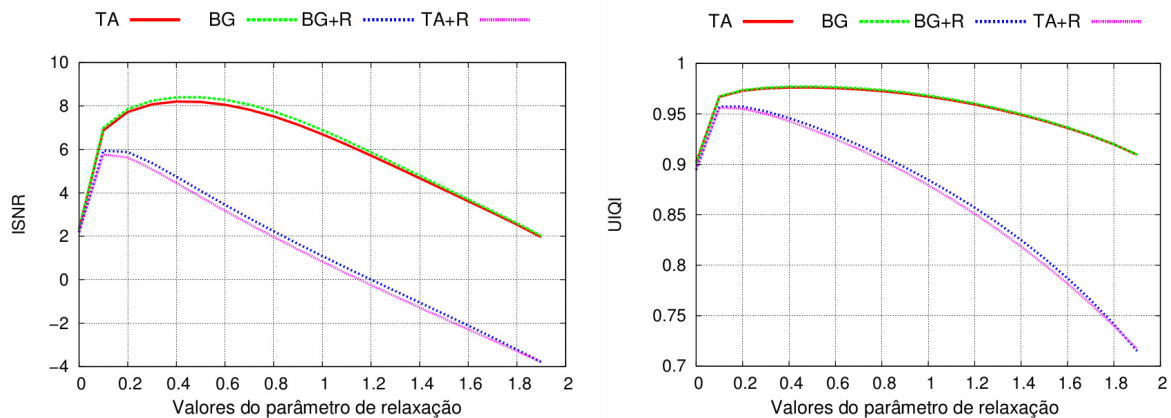
A Figura 6.15 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando UIQI: FFA para todos os casos.



**Figura 6.15. Imagens restauradas Sensoriamento Remoto 1 usando os ótimos  $\lambda$  encontrados (UIQI): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

## 6.5 Experimentos Sensoriamento Remoto - Imagem 2

Esta rodada de experimentos teve como objetivo executar o algoritmo POCS proposto (Equação 5.1) para restaurar as imagens degradadas do Sensoriamento Remoto 2 (Figuras 6.3b, 6.3c, 6.3d e 6.3e) usando os valores de  $\lambda$  no intervalo  $[0, 2]$  com passos de 0.1. (Este procedimento é o mesmo que nas seções anteriores, i.e., com as imagens Cameraman, Lena, Sensoriamento Remoto 1). Esta abordagem exaustiva foi empregada a fim de descobrir os valores de  $\lambda$  que maximizam as funções de ISNR e UIQI. A Figura 6.16 apresenta a curva dos resultados obtidos para borramento gaussiano (BG), borramento por turbulência atmosférica (TA), borramento gaussiano com ruído gaussiano (BG+R) e borramento por turbulência atmosférica com ruído gaussiano (TA+R). Os valores ótimos obtidos considerando ISNR foram  $\lambda = 0.5$  para BG com valor da métrica 8.401933,  $\lambda = 0.1$  para BG+R com valor da métrica 5.947758,  $\lambda = 0.4$  para TA com valor da métrica 8.203694 e  $\lambda = 0.1$  para TA+R com valor da métrica 5.766292. Considerando UIQI, foram  $\lambda = 0.5$  para BG com valor da métrica 0.976999,  $\lambda = 0.2$  para BG+R com valor da métrica 0.957197,  $\lambda = 0.5$  para TA com valor da métrica 0.976090 e  $\lambda = 0.1$  para TA+R com valor da métrica 0.955716.



**Figura 6.16. Busca exaustiva utilizando o algoritmo proposto para as métricas: (a) ISNR e (b) UIQI**

Foram aplicadas as técnicas PSO, HS, IHS, GHS, NGHS, SGHS, FFA, GSA e BA para encontrar o valor de  $\lambda$  ótimo que máxima as métricas de qualidade de imagem ISNR e UIQI. A ideia é comparar os valores obtidos por essas técnicas com os ótimos valores encontrados no experimento anterior (busca exaustiva). As Tabelas 6.18, 6.19, 6.20 e 6.21 mostram os resultados quantitativos para ISNR e UIQI (os valores em parenteses são os de  $\lambda$ ). Pode ser notado que todas as técnicas encontraram um resultado similar de restauração para todas as situações, o custo computacional do HS foi semelhante em relação ao experimento anterior. É importante observar que ambos os HS, FFA, GSA, BA e PSO encontraram valores de  $\lambda$  similares ao encontrado na busca exaustiva.

**Tabela 6.18. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG ISNR	BG UIQI
HS	8.416521 (0.44)	0.976999 (0.5)
NGHS	8.409868 (0.42)	0.977026 (0.49)
SGHS	8.415059 (0.45)	0.977051 (0.48)
GHS	8.416392 (0.46)	0.977065 (0.47)
IHS	8.417183 (0.47)	0.977051 (0.48)
PSO	8.417503 (0.471706)	0.977051 (0.48)
FFA	<b>8.417764 (0.463979)</b>	<b>0.977092 (0.471501)</b>
GSA	8.416905 (0.440208)	0.977063 (0.469667)
BA	8.417570 (0.470174)	0.977051 (0.48)

**Tabela 6.19. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	BG+R ISNR	BG+R UIQI
HS	5.984724 (0.11)	0.957741 (0.18)
NGHS	6.018192 (0.14)	0.958100 (0.13)
SGHS	5.947758 (0.1)	0.957741 (0.18)
GHS	5.984724 (0.11)	0.957578 (0.11)
IHS	6.006382 (0.12)	0.957578 (0.11)
PSO	6.019426 (0.138167)	0.958163 (0.14883)
FFA	<b>6.019664 (0.133999)</b>	<b>0.958166 (0.1487)</b>
GSA	6.019497 (0.136991)	0.958165 (0.14)
BA	6.018192 (0.14)	0.958165 (0.145139)

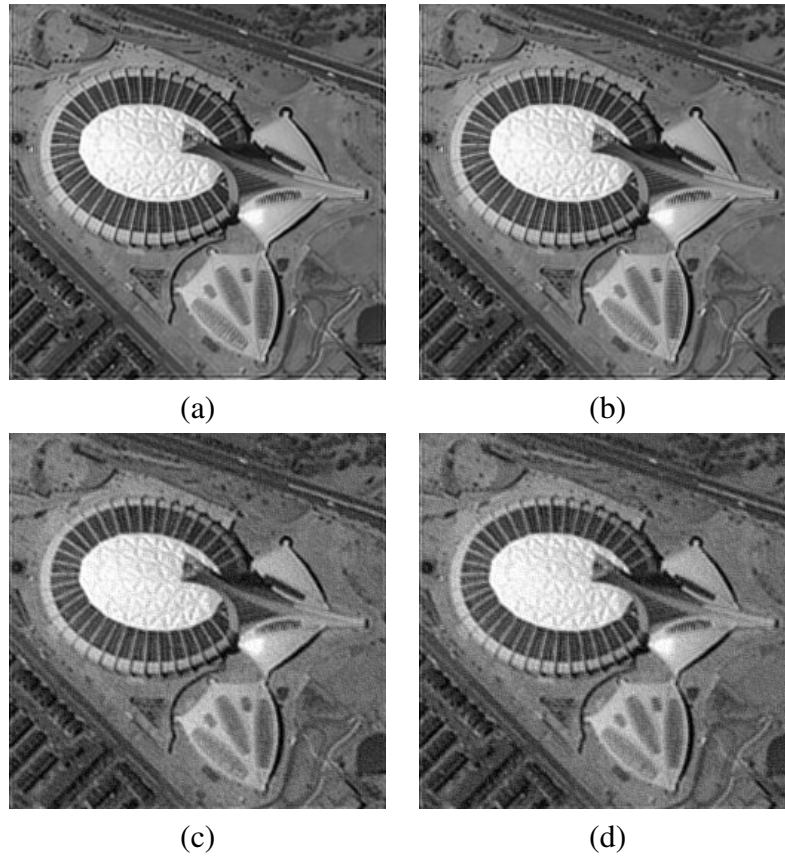
**Tabela 6.20. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

Técnicas	TA ISNR	TA UIQI
HS	8.211999 (0.4336)	0.976048 (0.51)
NGHS	8.206758 (0.41)	0.976145 (0.45)
SGHS	8.198660 (0.39)	0.976151 (0.47)
GHS	8.209858 (0.44)	0.976152 (0.43)
IHS	8.203117 (0.47)	0.976090 (0.5)
PSO	8.211972 (0.43)	0.976162 (0.445404)
FFA	<b>8.212492 (0.434193)</b>	<b>0.976165 (0.458163)</b>
GSA	8.211972 (0.43)	0.976161 (0.4642)
BA	8.212356 (0.434075)	0.976159 (0.450819)

**Tabela 6.21. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA+R, os melhores resultados estão em negrito e os valores entre parenteses são os de  $\lambda$ .**

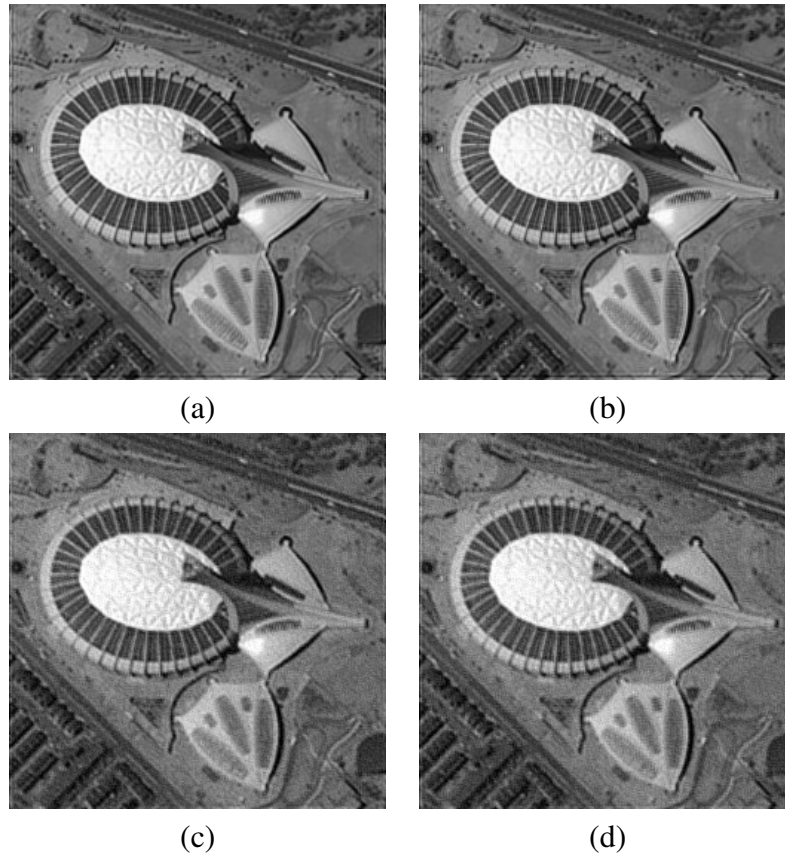
Técnicas	TA+R ISNR	TA+R UIQI
HS	5.792414 (0.15)	0.956471 (0.13)
NGHS	5.540574 (0.22)	0.956471 (0.13)
SGHS	5.792414 (0.15)	0.956353 (0.12)
GHS	5.812706 (0.13)	0.956497 (0.14)
IHS	5.812706 (0.13)	0.956353 (0.12)
PSO	5.812706 (0.13)	0.956514 (0.137936)
FFA	<b>5.813609 (0.128001)</b>	<b>0.956515 (0.137929)</b>
GSA	5.796649 (0.11)	0.956471 (0.13)
BA	5.811665 (0.121896)	0.956484 (0.142815)

A Figura 6.17 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando ISNR: FFA para todos os casos.



**Figura 6.17. Imagens restauradas Sensoriamento remoto 2 usando os ótimos  $\lambda$  encontrados (ISNR): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

A Figura 6.18 apresenta as imagens restauradas com os valores de  $\lambda$  obtidos com as melhores técnicas encontradas considerando UIQI: FFA para todos os casos.



**Figura 6.18. Imagens restauradas Sensoriamento Remoto 2 usando os ótimos  $\lambda$  encontrados (UIQI): (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

## 6.6 Aprendendo o valor de $\lambda$ - Experimento 2

A ideia desse experimento é mostrar que é possível aprender o valor de  $\lambda$  com uma imagem e restaurar outra com esse mesmo  $\lambda$  (o domínio das imagens não pode ser diferente). Esse é o mesmo experimento realizado na Seção 6.3. Os experimentos foram realizados como segue: Os melhores valores de  $\lambda$  encontrados nas imagens do Sensoriamento Remoto 1 foram usados para restaurar as imagens do Sensoriamento Remoto 2 (Figuras 6.4b, 6.4c, 6.4d e 6.4e).

De acordo com as Tabelas 6.22, 6.23, 6.24 e 6.25, pode ser observado que os resultados da restauração (métricas), são muito semelhantes aos resultados das métricas encontrados nas imagens do Sensoriamento Remoto 2.

**Tabela 6.22. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG (os valores entre parênteses são os de  $\lambda$ ).**

Técnicas	BG ISNR	BG UIQI
FFA	7.983434 (0.2251)	0.974758 (0.241399)

**Tabela 6.23. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com BG+R (os valores entre parênteses são os de  $\lambda$ ).**

Técnicas	BG+R ISNR	BG+R UIQI
FFA	—	0.955859 (0.082745)
PSO	5.768957 (0.074997)	—

**Tabela 6.24. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA (os valores entre parênteses são os de  $\lambda$ ).**

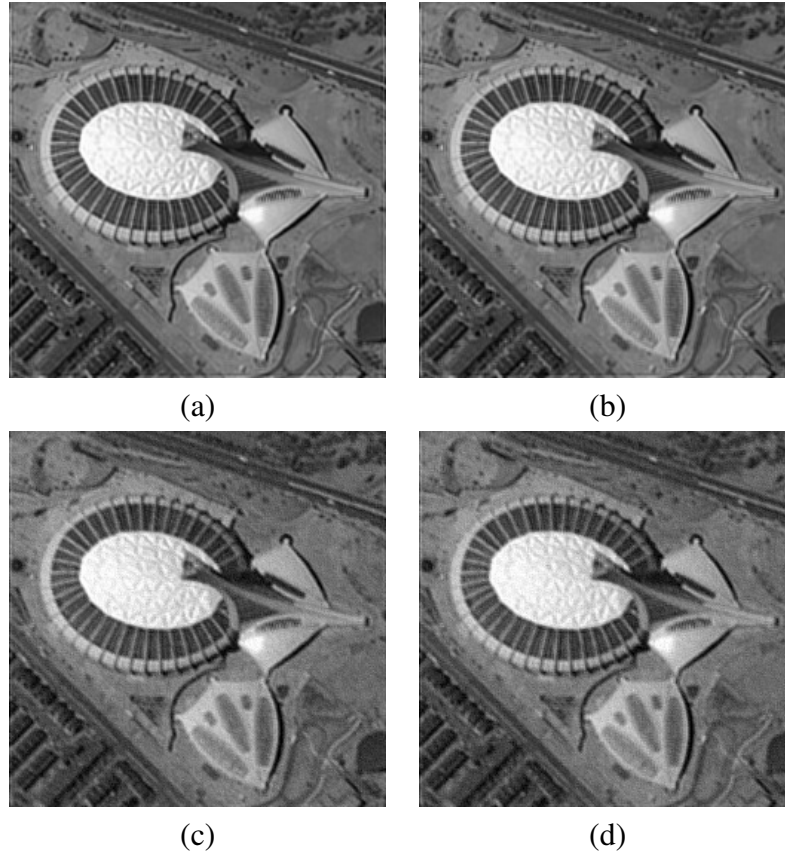
Técnicas	TA ISNR	TA UIQI
FFA	7.781003 (0.211831)	0.973381 (0.212698)

**Tabela 6.25. Resultados de restauração para imagens Sensoriamento Remoto 2 considerando as métricas ISNR e UIQI para experimentos com TA+R (os valores entre parênteses são os de  $\lambda$ ).**

Técnicas	TA+R ISNR	TA+R UIQI
FFA	5.586565 (0.072891)	0.954423 (0.081719)

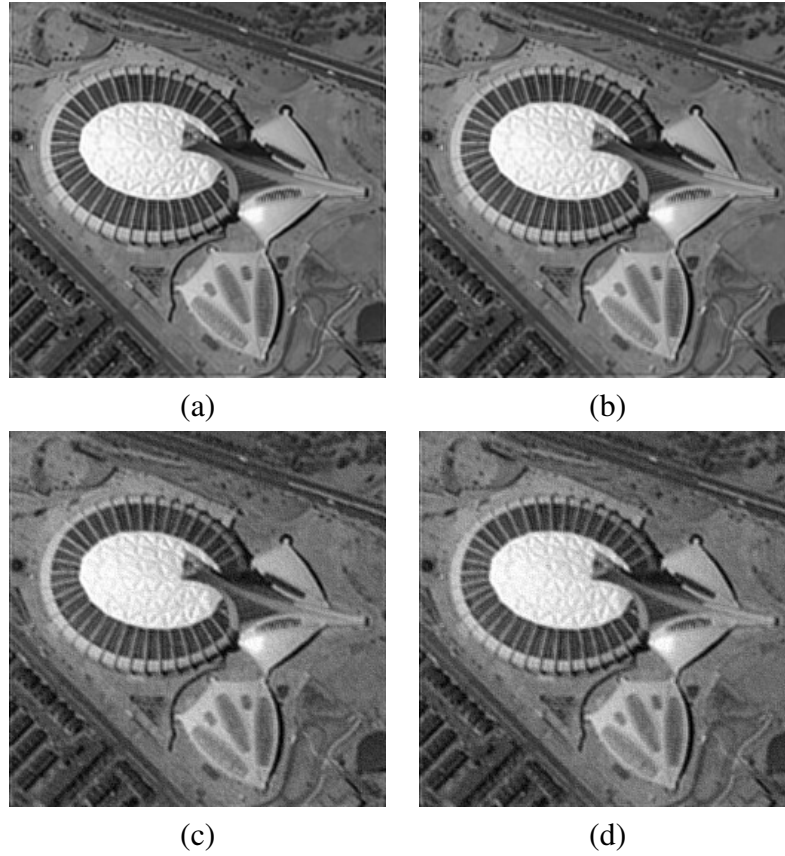


A Figura 6.19 apresenta as imagens restauradas do Sensoriamento Remoto 2 usando os melhores  $\lambda$  encontrados nos experimentos com as imagens do Sensoriamento Remoto 1 considerando ISNR: FFA para todos os casos somente PSO para BG+N.



**Figura 6.19. Imagens restauradas sensoriamento Remoto 2 usando os melhores  $\lambda$  obtidos no experimento com as imagens sensoriamento Remoto 1 considerando a métrica ISNR (a) FFA para BG (b) FFA para TA, (c) PSO para BG+R e (d) FFA para TA+R.**

A Figura 6.20 apresenta as imagens restauradas do Sensoriamento Remoto 2 usando os melhores  $\lambda$  encontrados nos experimentos com as imagens do Sensoriamento Remoto 1 considerando UIQI: FFA para todos os casos.



**Figura 6.20. Imagens restauradas sensoriamento Remoto 2 usando os melhores  $\lambda$  obtidos no experimento com as imagens sensoriamento Remoto 1 considerando a métrica UIQI: (a) FFA para BG, (b) FFA para TA, (c) FFA para BG+R e (d) FFA para TA+R.**

## 7 Conclusões

O processo de restauração de imagens considera recuperar uma imagem degradada por efeitos de borrimento e ruído. Caso possamos modelar o problema de restauração de imagens como um sistema linear, várias abordagens são propostas para resolver o conjunto de equações que levam as imagem restauradas. Uma dessas abordagens é baseada em projeções em conjuntos convexos, que emprega sucessivas projeções de forma iterativa para reconstruir a imagem. A principal preocupação são os parâmetros que controlam a projeção: valores baixos podem nos levar a resultados pobres de restauração, e altos valores podem aumentar o ruído. Portanto, propomos descobrir um parâmetro quase ótimo para orientar as projeções usando várias técnicas evolucionistas. Foram conduzidos os experimentos com as imagens Lena, Cameraman e Sensoriamento Remoto imagens 1 e 2 em duas etapas: primeiramente, uma busca exaustiva para encontrar os ótimos valores do parâmetro de relaxação  $\lambda$  no intervalo de  $[0, 2]$ . Em seguida foi conduzida a busca metaheurística para encontrar os valores de  $\lambda$  quase ótimos a fim de comparar as diferentes abordagens; depois usamos os melhores parâmetros de relaxação obtidos nas imagens da Lena e Sensoriamento remoto 1 para restaurar as imagens do Cameraman e Sensoriamento remoto 2 modelado com o mesmo borrimento e ruído. Podemos concluir que a técnica FFA encontrou os melhores resultados que maximizam as métricas ISNR e UIQI na maioria dos casos, as outras técnicas encontraram resultados similares para as imagens, sendo que o HS e suas variantes foram mais eficientes do ponto de vista computacional e com resultados muito próximos ao FFA. Como trabalho futuro, almeja-se a implementação de outras técnicas evolucionistas para encontrar o ótimo  $\lambda$  em imagens que estão na presença de diferentes tipos de degradações. Até o momento, foi obtido como publicação: A hybrid image restoration algorithm based on Projections Onto Convex Sets and Harmony Search no congresso *International Symposium on Circuits and Systems*, qualis A1, e submetido com o título Projections onto Convex Sets Parameter Estimation through Evolutionary Optimization and its Application for Image Restoration para o periódico *Natural Computing*, qualis B3.

## Referências

- [1] Rafael.C. Gonzalez and Richard.E. Woods. *Processamento Digital de Imagens (Terceira Edição)*. Pearson Education do Brasil., Rua Nelson Francisco, 26 São Paulo, Brasil, 2010.
- [2] Sumitro Samaddar and Richard J. Mammone. Image restoration using a row action projection method with adaptative smoothing. *Optical Engineering*, 34 No. 4:1147, 1995.
- [3] J.P. Papa, Leila M. G. Fonseca, and Lino A. S. de Carvalho. Projections onto convex sets through particle swarm optimization and its application for remote sensing image restoration. *Pattern Recognition Letters*, 31(13):1876–1886, 2010.
- [4] A.K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall: Nova Jersey, 1989.
- [5] H. Stark, Y. Yang, and Y. Yang. *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics*. John Wiley & Sons, 1998.
- [6] Richard J. Mammone. *Computation Methods of Signal Recovery and Recognition*. John Wiley & Sons, 1992.
- [7] Steven J. Leon. *Álgebra Linear com Aplicações*. Gen - Grupo Editorial Nacional, 2011.
- [8] L.M.G. Fonseca. Apostila de processamento digital de imagens. *INPE*, 1:CAP–224, 2000.
- [9] A.K. Katsaggelos. *Digital Image Restoration*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1991.
- [10] Xin She Yang. *Nature-Inspired Metaheuristic Algorithm*. Luniver Press, 2010.
- [11] J. Kennedy and R Eberhart. Particle swarm optimization. *IEEE*, pages 1942–1948, 1995.
- [12] J. Kennedy and R.C. Eberhart. *Swarm Intelligence*. Morgan Kaufman, 2001.
- [13] Z. W. Geem. *Music-Inspired Harmony Search Algorithm: Theory and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2009.
- [14] Mahamed G.H. Omran and Mehrdad Mahdavi. Global-best harmony search. *Applied Mathematics and Computation*, 198(2):643 – 656, 2008.
- [15] Quan-Ke Pan, P.N. Suganthan, M. Fatih Tasgetiren, and J.J. Liang. A self-adaptive global best harmony search algorithm for continuous optimization problems. *Applied Mathematics and Computation*, 216(3):830 – 848, 2010.
- [16] Dexuan Zou, Liqun Gao, Jianhua Wu, and Steven Li. Novel global harmony search algorithm for unconstrained problems. *Neurocomputing*, 73(16-18):3308 – 3318, 2010.
- [17] D. Halliday, R. Resnick, J. Walker, D. Halliday, R. Resnick, and J. Walker. *Fundamentals of Physics*, volume 6th Edition, chapter Extended. 2000.
- [18] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi. GSA: A gravitational search algorithm. *Information Sciences*, 179(13):2232–2248, 2009.

- [19] Z. Wang and A.C. Bovik. A universal image quality index. *IEEE Signal Processing Letters*, 9(3):81–84, 2002.

Autorizo a reprodução xerográfica para fins de pesquisa.

São José do Rio Preto, 24 / 02 / 2014

Rafael Gonçalves Pires  
Assinatura