

**ESTUDO DE MÉTODOS NUMÉRICOS
PARA ELIMINAÇÃO DE RUÍDOS
EM IMAGENS DIGITAIS**

Karina Miranda D'Ippólito

Dissertação de Mestrado
Pós-Graduação em Matemática Aplicada
MAP - 089

Estudo de Métodos Numéricos para Eliminação de Ruídos em Imagens Digitais

Karina Miranda D'Ippólito

Dissertação apresentada ao Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto, São Paulo, para a obtenção do título de Mestre em Matemática Aplicada.

Orientadora: Profa. Dra. Heloisa Helena Marino Silva

São José do Rio Preto

25 de fevereiro de 2005

A Deus.

Aos meus pais, Rossini e Estela.

Aos meus irmãos, Vi, Li, Rô e Jú.

Aos meus sobrinhos, Mu, Taí, Ti e Lelê.

Ao meu namorado Geraldo.

Dedico

Agradecimentos

A Deus, por ter me concedido a oportunidade de prosseguir com meus estudos.

Aos meus pais e à minha família, que sempre me apoiaram.

À Profa. Dra. Heloisa Helena Marino Silva, pela orientação e incentivo na produção deste trabalho.

Ao Prof. Dr. Maurílio Boaventura, que apoiou e contribuiu muito para o desenvolvimento deste trabalho.

Ao Prof. Ms. Evanivaldo Castro Silva Júnior, pelo apoio computacional.

Ao meu irmão Rossini, pelos inúmeros favores.

À grande amiga Claudia, que adquiri nesse programa de pós-graduação.

Ao Daniel, pela sua amizade, força e por compartilhar seu tempo no estudo das disciplinas.

À Andrea, pela amizade.

Ao André, pela dedicação, companheirismo e pelo suporte computacional.

Aos demais colegas de pós-graduação do ano de 2003.

Aos funcionários do IBILCE, pela dedicação com que prestaram o seu apoio.

Às companheiras de morada, Cleiry, Fabiana, Lorena, Luciana, Mariela, Patrícia e Solange.

Ao CNPq, pelo auxílio financeiro.

Resumo

O objetivo deste trabalho é apresentar um estudo sobre a aplicação de métodos numéricos para a resolução do modelo proposto por Barcelos, Boaventura e Silva Jr. [7], para a eliminação de ruídos em imagens digitais por meio de uma equação diferencial parcial, e propor uma análise da estabilidade do método iterativo comumente aplicado a este modelo. Uma análise comparativa entre os vários métodos abordados é realizada através de resultados experimentais em imagens sintéticas e imagens da vida real.

Palavras-chave: Métodos numéricos, equação diferencial parcial, estabilidade, restauração de imagens.

Abstract

The purpose of this work is to present a study on the application of numerical methods for the resolution of model considered by Barcelos, Boaventura and Silva Jr [7], for image denoising through a partial differential equation, and to consider a stability analysis of an iterative method usually applied to this model. A comparative analysis among various considered methods is carried out through experimental results for synthetic and real images.

Key-words: Numerical methods, partial differential equation, stability, image restoration.

Sumário

Introdução	1
1 Soluções Aproximadas para Equações Diferenciais Ordinárias	4
1.1 Problema de Valor Inicial	5
1.1.1 Problema Bem-posto	5
1.2 Métodos Discretos	6
1.3 Método de Euler	10
1.4 Métodos de Runge-Kutta	12
1.4.1 Fórmulas de Runge-Kutta de Ordem Superior	16
1.5 Método Previsor-Corretor	17
1.6 Integração Trapezoidal	20
1.6.1 Método de Euler Melhorado	20
1.7 Método de Euler Implícito	21
2 Modelo BBS para a Eliminação de Ruídos em Imagens	23
2.1 Representação de uma Imagem Digital	23
2.2 Modelos para Eliminação de Ruídos em Imagens	24
2.3 Modelo BBS	25
2.3.1 Termo de Difusão	27
2.3.2 Método de Diferenças Finitas	28
2.3.3 Discretização da Função $g(\nabla(G_\sigma * u))$	33

2.3.4	Estabilidade para o Método de Euler Aplicado ao Modelo	
	BBS	35
3	Resultados	39
3.1	Implementação Numérica	39
3.2	Considerações Finais	69
	Referências Bibliográficas	71

Introdução

O uso de equações diferenciais parciais (EDP) no processo de restauração de imagens tornou-se, nos últimos anos, sólido e proveitoso possibilitando o surgimento de vários modelos matemáticos. Dentre estes, estudamos o modelo, recentemente proposto por Barcelos, Boaventura e Silva Jr. [7], que denominamos, no contexto deste trabalho, modelo *BBS*, dado por:

$$u_t = g|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) - \lambda(1 - g)(u - I), \quad (1)$$

onde

$$u(x, y, 0) = I(x, y), \quad g(s) = \frac{1}{1 + ks^2}, \quad k \text{ constante},$$

$$s = |\nabla(G_\sigma * u)|$$

e

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp(-(x^2 + y^2)/2\sigma^2)$$

é um núcleo Gaussiano de escala temporal σ .

O primeiro termo do lado direito de (1) representa o termo de difusão que permite a eliminação dos ruídos da imagem inicial $I(x, y)$. O segundo, é um termo forçante que faz com que a imagem $u(x, y)$ permaneça perto de $I(x, y)$. A formulação desse modelo apresenta um balanceamento entre os dois termos da equação dada pela função g que funciona como um detector de bordas.

A eliminação de ruídos em imagens através de uma equação diferencial parcial baseia-se fundamentalmente num processo de evolução dessas equações no tempo, ou seja, a partir de uma imagem inicial $u(x, y, 0)$ no tempo $t = 0$, calcula-se a solução da equação para sucessivos instantes de tempo, obtendo-se imagens “melhoradas” a cada nível subsequente.

A equação diferencial é, em geral, resolvida por métodos numéricos. Um dos métodos mais tradicionais é o método de diferenças finitas em que as derivadas presentes nas equações são aproximadas por diferenças entre valores da solução discretizada, ou seja, a equação diferencial é substituída por uma equação de diferenças cuja solução é obtida, em cada instante de tempo t_n , numa malha bidimensional de pontos.

Se denotarmos

$$\mathcal{L}(u) = g|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) - \lambda(1 - g)(u - I),$$

podemos escrever o problema (1) na forma

$$u_t = \mathcal{L}(u), \tag{2}$$

com $u(x, y, 0) = I(x, y)$.

A solução deste problema é encontrada substituindo-se as derivadas presentes em $\mathcal{L}(u)$ por diferenças finitas e a equação diferencial é então aproximada, numa malha regular de passo $h = \Delta x = \Delta y$, pela equação discretizada

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^n), \quad \text{com} \quad u_{i,j}^0 = I(x_i, y_j), \tag{3}$$

onde Δt representa o incremento temporal e $u_{i,j}^n$ a solução aproximada de $u(x_i, y_j)$ no tempo t_n .

Este processo iterativo tem sido utilizado, de um modo geral, para encontrar a solução das equações que modelam o problema de restauração de imagens. As pesquisas, nesta área, têm sido mais voltadas para questões envolvendo os modelos matemáticos.

O nosso objetivo, neste trabalho, é aplicar métodos numéricos, comumente utilizados para a resolução de equações diferenciais ordinárias, para a resolução do problema (2), considerado como sendo uma equação diferencial ordinária na variável temporal, e mostrar, através de resultados experimentais, seus efeitos na eliminação de ruídos em imagens digitais. Também, apresentamos um resultado sobre a estabilidade do esquema numérico (3) aplicado ao modelo *BBS*.

Este trabalho encontra-se em capítulos.

No Capítulo 1, apresentamos alguns conceitos e resultados referentes à resolução de uma equação diferencial ordinária com a finalidade de apresentarmos os métodos numéricos que utilizamos para resolver o problema (2).

No Capítulo 2, apresentamos o modelo *BBS* fazendo um breve histórico dos modelos matemáticos que fundamentaram sua formulação. Em seguida, apresentamos a discretização da equação diferencial que representa o modelo e finalizamos com um resultado sobre a estabilidade do esquema numérico (3) aplicado para a resolução da equação.

O Capítulo 3 destina-se aos resultados experimentais obtidos na reconstrução de imagens digitais pela resolução da equação (1) utilizando os métodos numéricos apresentados no Capítulo 1. Para fazermos uma análise comparativa do desempenho dos vários métodos abordados, apresentamos um conceito de erro, que utilizamos como uma medida de “precisão”, além da comparação gráfica, isto é, da visualização das imagens obtidas. Finalizamos o capítulo com algumas considerações a respeito dos resultados.

Capítulo 1

Soluções Aproximadas para Equações Diferenciais Ordinárias

As equações diferenciais aparecem freqüentemente em problemas da ciência e engenharia que envolvem a mudança de alguma variável com relação a outra. Muitos destes problemas requerem a solução de um problema de valor inicial, que é, a solução para uma equação diferencial a qual satisfaz uma condição inicial dada. Esta solução pode ser obtida através de métodos analíticos, os quais se restringem a algumas formas especiais de função, e de métodos numéricos, que não possuem uma limitação, contudo, a solução numérica é encontrada como uma tabela de valores da função calculada em diferentes valores da variável independente, em vez de uma relação funcional como na solução analítica.

Neste capítulo, trataremos de alguns métodos numéricos para aproximar soluções de equações diferenciais ordinárias, sujeitas às condições iniciais, com o objetivo de aplicá-los ao modelo *BBS* que estamos interessados em resolver.

1.1 Problema de Valor Inicial

Consideremos o problema de valor inicial (PVI), de primeira ordem, da forma

$$\begin{cases} y' &= f(x, y) \\ y(a) &= \eta \\ a \leq x \leq b \text{ e } -\infty < y < \infty \end{cases}, \quad (1.1.1)$$

onde f é uma função real.

Sua solução é uma função $y = y(x)$ contínua e diferenciável que satisfaz (1.1.1) e é única dado que o PVI satisfaz as condições estabelecidas no seguinte Teorema:

Teorema 1.1. *Seja $f(x, y)$, $f : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, uma função definida e contínua para todo (x, y) na região D dada por $a \leq x \leq b$ e $-\infty < y < \infty$, com a e b números finitos, e L uma constante tal que*

$$\|f(x, y) - f(x, y^*)\| \leq L\|y - y^*\| \quad (1.1.2)$$

é válida para todo $(x, y), (x, y^*) \in D$. Então, para algum $\eta \in \mathbb{R}$, existe uma única solução $y(x)$ do PVI (1.1.1), onde $y(x)$ é contínua e diferenciável para todo $(x, y) \in D$. (LAMBERT, [16], p.5)

A desigualdade (1.1.2) é conhecida como uma condição de Lipschitz e L como uma constante de Lipschitz.

Além da questão de existência e unicidade da solução do problema (1.1.1), precisamos saber se a solução depende continuamente dos dados do problema, isto é, se pequenas perturbações nos dados implicam em pequenas variações da solução.

1.1.1 Problema Bem-posto

O problema de valor inicial (1.1.1) é um problema bem-posto se:

i) existe uma solução única $y(x)$ para o problema;

ii) para qualquer $\varepsilon > 0$, existe uma constante positiva $k(\varepsilon)$ com a propriedade que, sempre que $|\varepsilon_0| < \varepsilon$ e $\delta(x)$ for contínua com $|\delta(x)| < \varepsilon$ em $[a, b]$, existe uma única solução, $z(x)$, para o problema

$$z' = f(x, z) + \delta(x), \quad a \leq x \leq b, \quad z(a) = \eta + \varepsilon_0, \quad (1.1.3)$$

com

$$|z(x) - y(x)| < k(\varepsilon)\varepsilon, \quad \text{para todo } a \leq x \leq b.$$

O problema dado pela equação (1.1.3) é denominado um problema perturbado associado ao problema original (1.1.1). Os métodos numéricos estarão sempre sujeitos a resolver um problema perturbado, uma vez que qualquer erro introduzido na representação resultará em um problema deste tipo. A menos que o problema original seja bem-posto, existe pouca razão para esperar que a solução numérica de um problema perturbado se aproxime com exatidão da solução do problema original.

O teorema seguinte fornece condições que garantem que um problema de valor inicial é bem-posto. A prova deste teorema pode ser encontrada em (BIRKHOFF; ROTA, [8], p.142-147).

Teorema 1.2. *Suponha $D = \{(x, y) / a \leq x \leq b \text{ e } -\infty < y < \infty\}$. Se f é contínua e satisfaz uma condição de Lipschitz na variável y sobre o conjunto D , então o problema de valor inicial (1.1.1) é dito bem-posto. (BURDEN; FAIRES, [9], p.237)*

1.2 Métodos Discretos

A principal meta de métodos discretos é a redução de sistemas contínuos a equivalentes sistemas discretos.

A aproximação básica envolve a substituição de um domínio contínuo D por um conjunto de pontos discretos limitados por D . Ao invés de encontrar

uma solução definida em toda parte do domínio, são obtidas apenas aproximações nos pontos isolados x_i . Valores intermediários, integrais, derivadas, ou outros operadores podem ser obtidos a partir desta solução discreta por técnicas de interpolação.

Os métodos numéricos que apresentaremos são fundamentados da seguinte maneira: dado o PVI (1.1.1), construímos $x_1, x_2, \dots, x_N$ de modo que $x_i = a + ih$, $h = \frac{b-a}{N}$, $i = 1, 2, \dots, N$, e calculamos a aproximação y_i da solução exata $y(x_i)$ nestes pontos, usando informações de passos anteriores.

Estes métodos podem ser escritos na forma geral

$$\sum_{j=0}^k \alpha_j y_{i+j} = h \phi_f(y_{i+k}, y_{i+k-1}, \dots, y_i, x_i; h), \quad (1.2.4)$$

onde f indica que a dependência de ϕ sobre $y_{i+k}, y_{i+k-1}, \dots, y_i, x_i$ se dá através da função $f(x, y)$. Duas condições são impostas sobre (1.2.4):

$$\phi_{f=0}(y_{i+k}, y_{i+k-1}, \dots, y_i, x_i; h) \equiv 0 \quad \text{e}$$

$$\begin{aligned} |\phi_f(y_{i+k}, y_{i+k-1}, \dots, y_i, x_i; h) - \phi_f(y_{i+k}^*, y_{i+k-1}^*, \dots, y_i^*, x_i; h)| \\ \leq M \sum_{j=0}^k |y_{i+j} - y_{i+j}^*|, \quad M \text{ constante.} \end{aligned}$$

Estas condições não são de modo algum restritivas; para os métodos numéricos da forma (1.2.4), considerados neste trabalho, a primeira condição é satisfeita, enquanto a segunda é uma consequência do fato de admitirmos que o problema de valor inicial (1.1.1) satisfaz uma condição de Lipschitz.

Definição 1.1. (*Métodos de Passo Simples*)

Um método é de passo simples (ou passo um) quando a aproximação y_{i+1} for calculada a partir somente do valor y_i do passo anterior. Sendo ϕ a função incremento, um método de passo simples é definido na forma

$$y_{i+1} = y_i + h \phi(x_i, y_i; h).$$

Definição 1.2. (*Métodos de Passo Múltiplo*)

Um método de passo múltiplo (ou passo m) a ser usado para a resolução do problema de valor inicial

$$y' = f(x, y), \quad a \leq x \leq b, \quad y(a) = \eta,$$

é um método cuja equação de diferenças para determinar a aproximação y_{i+1} no ponto da malha x_{i+1} pode ser representada pela seguinte equação, onde m é um inteiro maior do que 1:

$$\begin{aligned} y_{i+1} &= \alpha_{m-1} y_i + \alpha_{m-2} y_{i-1} + \dots + \alpha_0 y_{i+1-m} \\ &+ h[\beta_m f(x_{i+1}, y_{i+1}) + \beta_{m-1} f(x_i, y_i) + \dots + \beta_0 f(x_{i+1-m}, y_{i+1-m})], \end{aligned} \quad (1.2.5)$$

para $i = m - 1, m, \dots, N - 1$, e os valores iniciais dados $y_0 = \eta_0$, $y_1 = \eta_1$, $y_2 = \eta_2, \dots, y_{m-1} = \eta_{m-1}$, com $h = \frac{b-a}{N}$.

Quando $\beta_m = 0$, o método é chamado explícito ou aberto, e a equação (1.2.5) encontra y_{i+1} explicitamente em termos de valores anteriormente determinados. Quando $\beta_m \neq 0$, o método é dito implícito ou fechado e, y_{i+1} ocorre em ambos os lados de (1.2.5) e assim é determinado implicitamente.

Definição 1.3. (*Erro Local*)

Supondo que o valor calculado por um método de passo m seja exato, isto é, $y_{i+j} = y(x_{i+j})$ para $j = 0, 1, 2, \dots, m-1$, então o erro local em x_{i+m} é definido como

$$e_{i+m} = y(x_{i+m}) - y_{i+m}.$$

Definição 1.4. (*Ordem*)

Um método de passo simples tem ordem q se a função incremento ϕ for tal que

$$y(x+h) = y(x) + h\phi(x, y; h) + O(h^{q+1}).$$

A ordem de aproximação de um método indica que, à medida que h tende a zero, o termo $O(h^q)$ tende a zero na mesma velocidade que h^q . Em geral, um maior valor de q indica uma melhor aproximação do método.

Definição 1.5. (*Consistência*)

Um método numérico é dito consistente com o PVI (1.1.1) se a sua ordem $q \geq 1$.

Definição 1.6. (*Estabilidade*)

O método dado por (1.2.5) é estável se todas as soluções da equação de diferenças

$$y_{i+1} = \alpha_{m-1} y_i + \alpha_{m-2} y_{i-1} + \dots + \alpha_0 y_{i+1-m}$$

são limitadas quando $i \rightarrow \infty$.

De um modo geral, suponha que a um método numérico, para a resolução do problema (1.1.1), associamos a equação de diferenças (1.2.5).

A estabilidade do método é analisada através das raízes do polinômio característico

$$p(\lambda) = \alpha_{m-1} \lambda^{m-1} + \alpha_{m-2} \lambda^{m-2} + \dots + \alpha_1 \lambda + \alpha_0,$$

formado pelos coeficientes α_i usados na equação de diferenças (1.2.5).

Sejam λ_i , $i = 1, 2, \dots, m-1$, as raízes da equação $p(\lambda) = 0$. Se $|\lambda_i| \leq 1$, $i = 1, 2, \dots, m-1$, e todas as raízes com valor absoluto igual a 1, se existirem, são raízes simples (isto é, $|\lambda_i| = 1$ então $p'(\lambda_i) \neq 0$), então, dizemos que o método satisfaz a condição da raiz.

Num contexto mais amplo, a estabilidade de um método de discretização para equações diferenciais é caracterizada por meio da seguinte definição:

Definição 1.7.

i) Se um método satisfaz a condição da raiz e $\lambda = 1$ é a única possibilidade para

as raízes de módulo 1, então ele é chamado fortemente estável;

ii) os métodos que satisfazem a condição da raiz e têm mais de uma raiz com módulo igual a 1 são chamados fracamente estáveis;

iii) os métodos que não satisfazem a condição da raiz são chamados instáveis.
(CUNHA, [13], p.192)

Definição 1.8. (*Convergência*)

Um método de passo m é convergente se, para o PVI (1.1.1),

$$\lim_{h \rightarrow 0} y_i = y(x_i), \quad ih = x - a$$

é válido para todo $x \in [a, b]$, e os valores iniciais são tais que

$$\lim_{h \rightarrow 0} y_j(h) = \eta, \quad j = 0, 1, \dots, m-1.$$

A consistência de um método significa que a solução numérica corresponde à solução do PVI. Esta consistência limita a magnitude do erro local cometido em cada passo e a estabilidade controla a propagação do erro durante os cálculos. Um método é convergente se ele for consistente e estável.

Passamos agora para a definição dos métodos numéricos que serão utilizados na resolução de um PVI da forma (1.1.1) associado à equação que modela o problema de restauração de imagens o qual tratamos neste trabalho.

1.3 Método de Euler

O objetivo do método de Euler é obter uma aproximação para o problema de valor inicial bem-posto (1.1.1). Para definir este método, usaremos o desenvolvimento da série de Taylor.

Teorema 1.3. *Suponha que $f \in C^n[a, b]$, f^{n+1} existe em $[a, b]$, e $x_0 \in [a, b]$. Para todo $x \in [a, b]$, existe $\xi(x)$ entre x_0 e x tal que*

$$f(x) = P_n(x) + R_n(x),$$

onde

$$\begin{aligned} P_n(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \\ &= \sum_{k=0}^n \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k, \end{aligned}$$

e

$$R_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)^{n+1}.$$

(BURDEN; FAIRES, [9], p.8)

Aqui, $P_n(x)$ é chamado de enésimo polinômio de Taylor para f sobre x_0 e $R_n(x)$ o termo restante (ou erro de truncamento) associado a $P_n(x)$. A série infinita obtida tomando o limite de $P_n(x)$ quando $n \rightarrow \infty$ é chamada série de Taylor para f sobre x_0 .

Através do desenvolvimento desta série, tomando $n = 1$, podemos definir o método de Euler como a seguir.

Suponha que $y(x)$, solução única do *PVI* (1.1.1), tenha derivadas contínuas até segunda ordem em $[a, b]$, tal que para cada $i = 0, 1, 2, \dots, N - 1$,

$$y(x_{i+1}) = y(x_i) + (x_{i+1} - x_i)y'(x_i) + \frac{(x_{i+1} - x_i)^2}{2}y''(\xi_i), \quad \xi_i \text{ em } (x_i, x_{i+1}).$$

Se $h = x_{i+1} - x_i$, então

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(\xi_i)$$

e desde que $y(x)$ satisfaz a equação diferencial (1.1.1),

$$y(x_{i+1}) = y(x_i) + hf(x_i, y(x_i)) + \frac{h^2}{2}y''(\xi_i).$$

O método de Euler produz $y_i \approx y(x_i)$ para cada $i = 1, 2, \dots, N$, excluindo o termo restante $\frac{h^2}{2}y''(\xi_i)$.

Assim,

$$\begin{cases} y_0 &= \eta \\ y_{i+1} &= y_i + hf(x_i, y_i), \quad i = 0, 1, 2, \dots, N - 1 \end{cases} \quad (1.3.6)$$

é chamada equação de diferenças associada ao método de Euler.

Note que, pela Definição 1.4, o método de Euler tem precisão $O(h)$.

1.4 Métodos de Runge-Kutta

Desenvolvemos, anteriormente, um dos métodos numéricos mais simples para problemas de valor inicial. Agora, mostraremos como obter uma melhor precisão sem mudar h , mas alterando a fórmula recursiva (1.3.6). No presente desenvolvimento, vamos supor que podemos usar casas decimais suficientes de modo que o erro de arredondamento possa ser desprezado.

Consideremos, então, a fórmula geral

$$\frac{y_{i+1} - y_i}{h} = \eta f(x_i, y_i) + \beta f(x_i + \gamma h, y_i + \delta h), \quad (1.4.7)$$

onde η , β , γ e δ são parâmetros a serem determinados. Naturalmente, se $\eta = 1$ e $\beta = 0$, a equação (1.4.7) resulta na fórmula de Euler. A razão para considerarmos (1.4.7) está no fato de que $f(x, y)$ é conhecida para todo x em $[x_i, x_{i+1}]$ e para todo y . Conseqüentemente, podemos incorporar este fato numa fórmula que produzirá uma melhor precisão.

Primeiramente, vamos reescrever (1.4.7) na forma equivalente

$$y_{i+1} = y_i + \eta h f(x_i, y_i) + \beta h f(x_i + \gamma h, y_i + \delta h). \quad (1.4.8)$$

Agora, usaremos a expansão da fórmula de Taylor para uma função de duas variáveis para desenvolvermos $f(x_i + \gamma h, y_i + \delta h)$.

Teorema 1.4. *Suponha que $f(x, y)$ e todas as derivadas parciais de ordem menor ou igual a $n + 1$ sejam contínuas em $D = \{(x, y) / a \leq x \leq b \text{ e } c \leq y \leq d\}$. Seja $(x_0, y_0) \in D$. Para todo $(x, y) \in D$, existe ξ entre x e x_0 e η entre y e y_0 com*

$$f(x, y) = P_n(x, y) + R_n(x, y), \quad (1.4.9)$$

onde

$$\begin{aligned}
 P_n(x, y) &= f(x_0, y_0) + [(x - x_0) \frac{\partial f}{\partial x}(x_0, y_0) + (y - y_0) \frac{\partial f}{\partial y}(x_0, y_0)] \\
 &+ [\frac{(x - x_0)^2}{2} \frac{\partial^2 f}{\partial x^2}(x_0, y_0) + (x - x_0)(y - y_0) \frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) \\
 &+ \frac{(y - y_0)^2}{2} \frac{\partial^2 f}{\partial y^2}(x_0, y_0)] + \dots \\
 &+ [\frac{1}{n!} \sum_{j=0}^n \binom{n}{j} (x - x_0)^{n-j} (y - y_0)^j \frac{\partial^n f}{\partial x^{n-j} \partial y^j}(x_0, y_0)]
 \end{aligned} \tag{1.4.10}$$

e

$$R_n(x, y) = \frac{1}{(n+1)!} \sum_{j=0}^{n+1} \binom{n+1}{j} (x - x_0)^{n+1-j} (y - y_0)^j \frac{\partial^{n+1} f}{\partial x^{n+1-j} \partial y^j}(\xi, \eta).$$

A função P_n é chamada o n -ésimo polinômio de Taylor em duas variáveis para a função f em (x_0, y_0) , e $R_n(x, y)$ é o termo do resto associado a $P_n(x, y)$. (BURDEN; FAIRES, [9], p.254-255)

Assim, usando a notação

$$\frac{\partial f}{\partial x} = f_x, \quad \frac{\partial f}{\partial y} = f_y, \quad \frac{\partial^2 f}{\partial x^2} = f_{xx}, \quad \frac{\partial^2 f}{\partial x \partial y} = f_{xy}, \quad \frac{\partial^2 f}{\partial y^2} = f_{yy}, \dots,$$

temos

$$\begin{aligned}
 f(x_i + \gamma h, y_i + \delta h) &= f(x_i, y_i) + \gamma h f_x(x_i, y_i) + \delta h f_y(x_i, y_i) \\
 &+ \frac{1}{2} [\gamma^2 h^2 f_{xx}(x_i, y_i) + 2\gamma \delta h^2 f_{xy}(x_i, y_i) \\
 &+ \delta^2 h^2 f_{yy}(x_i, y_i)] + O(h^3).
 \end{aligned} \tag{1.4.11}$$

A substituição de (1.4.11) em (1.4.8) e a reordenação dos termos nos dá:

$$\begin{aligned}
 y_{i+1} &= y_i + h(\eta + \beta) f(x_i, y_i) + \frac{h^2}{2} [2\beta \gamma f_x(x_i, y_i) + 2\beta \delta f_y(x_i, y_i)] \\
 &+ \frac{h^3}{6} [3\beta \gamma^2 f_{xx}(x_i, y_i) + 6\beta \gamma \delta f_{xy}(x_i, y_i) \\
 &+ 3\beta \delta^2 f_{yy}(x_i, y_i)] + \beta O(h^4).
 \end{aligned} \tag{1.4.12}$$

Note que o lado direito de (1.4.12) é da forma particular

$$y_{i+1} = y_i + hF_1 + \frac{h^2}{2}F_2 + \frac{h^3}{6}F_3 + O(h^4), \quad (1.4.13)$$

onde F_1 , F_2 e F_3 são os respectivos coeficientes em (1.4.12).

Note também que (1.4.13) é um tanto parecida com uma expansão de Taylor para uma função de uma variável. Assim, suponhamos que a solução exata de (1.1.1) é $Y(x)$ e que esta pode ser escrita como a série de Taylor

$$Y_{i+1} = Y_i + hY'_i + \frac{h^2}{2}Y''_i + \frac{h^3}{6}Y'''_i + \dots \quad (1.4.14)$$

De $Y' = f(x, Y)$, segue pela regra da cadeia que

$$\begin{aligned} Y'' &= f_x + f_y f, \\ Y''' &= f_{xx} + 2f f_{xy} + f^2 f_{yy} + f_x f_y + f f_y^2, \end{aligned}$$

e após substituição em (1.4.14), produz

$$\begin{aligned} Y_{i+1} &= Y_i + hf(x_i, Y_i) + \frac{h^2}{2}[f_x(x_i, Y_i) + f_y(x_i, Y_i)f(x_i, Y_i)] \\ &+ \frac{h^3}{6}\{f_{xx}(x_i, Y_i) + 2f(x_i, Y_i)f_{xy}(x_i, Y_i) + f_{yy}(x_i, Y_i)[f(x_i, Y_i)]^2 \\ &+ f_x(x_i, Y_i)f_y(x_i, Y_i) + f(x_i, Y_i)[f_y(x_i, Y_i)]^2\} + O(h^4). \end{aligned} \quad (1.4.15)$$

Agora, (1.4.12) é uma expansão para a solução numérica, enquanto (1.4.15) é uma expansão para a solução exata. Vamos examinar quais as condições para a escolha de η , β , γ e δ tal que, se $y_i = Y_i$ então (1.4.12) para y_{i+1} concorda com (1.4.15) para Y_{i+1} .

Para as escolhas especiais $\eta = 1$ e $\beta = 0$, (1.4.12) está de acordo com (1.4.15), mas não incluindo os termos de h^2 . Esta escolha, como vista anteriormente, produz a fórmula de Euler. Se desejamos construir uma fórmula para que (1.4.12) esteja de acordo com (1.4.15), pelo menos até os termos de h^2 , e neste sentido, é uma melhor aproximação que a fórmula de Euler, necessitamos apenas de coeficientes correspondentes iguais para os termos de h e de h^2 . Logo,

basta tomarmos

$$\begin{cases} \eta + \beta &= 1 \\ 2\beta\gamma f_x + 2\beta\delta f_y &= f_x + f f_y \end{cases},$$

que é um sistema de duas equações com quatro parâmetros η , β , γ e δ . Observando a segunda equação deste sistema, podemos transformá-lo em um sistema de três equações

$$\begin{cases} \eta + \beta &= 1 \\ 2\beta\gamma &= 1 \\ 2\beta\delta &= f \end{cases},$$

exigindo que os correspondentes coeficientes de f_x e f_y sejam iguais. Agora, temos um sistema de três equações para quatro parâmetros. Este sistema tem infinitas soluções. Consideremos, por exemplo,

$$\eta = \beta = \frac{1}{2}, \quad \gamma = 1, \quad \delta = f(x_i, y_i).$$

Substituindo estes parâmetros em (1.4.8), temos:

$$y_{i+1} = y_i + \frac{h}{2}f(x_i, y_i) + \frac{h}{2}f(x_{i+1}, y_i + hf(x_i, y_i)), \quad (1.4.16)$$

que é uma fórmula de Runge-Kutta de segunda ordem. O nome “Runge-Kutta” provém de dois matemáticos, os quais foram os primeiros a desenvolver os métodos e as fórmulas discutidas acima. A fórmula particular (1.4.16) é uma fórmula de segunda ordem porque a expansão de Taylor (1.4.15) para a solução exata Y_{i+1} e a expansão de Taylor (1.4.12) para a solução numérica y_{i+1} estão de acordo, ou são idênticas, até os termos da ordem de h^2 , supondo que $y_i = Y_i$. A fórmula (1.4.16) é uma melhor aproximação que a fórmula de Euler porque a expansão de Taylor para y_{i+1} na fórmula de Euler e a expansão de Taylor para a solução exata estão de acordo apenas até os termos da ordem de h . Os termos da expansão da solução numérica que não concordam com os termos da expansão da solução exata de Taylor são chamados o erro de truncamento.

A fórmula (1.4.16) pode ser calculada eficientemente em três passos, definidos a seguir. Primeiramente, seja

$$K_0 = f(x_i, y_i).$$

Então, (1.4.16) torna-se

$$y_{i+1} = y_i + \frac{1}{2}K_0 + \frac{h}{2}f(x_{i+1}, y_i + K_0). \quad (1.4.17)$$

Em seguida, seja

$$K_1 = hf(x_{i+1}, y_i + K_0).$$

Assim, (1.4.17) torna-se

$$y_{i+1} = y_i + \frac{1}{2}(K_0 + K_1).$$

Resumindo, (1.4.16) é equivalente ao cálculo dos três passos:

$$\begin{aligned} K_0 &= hf(x_i, y_i), \\ K_1 &= hf(x_{i+1}, y_i + K_0), \\ y_{i+1} &= y_i + \frac{1}{2}(K_0 + K_1). \end{aligned}$$

1.4.1 Fórmulas de Runge-Kutta de Ordem Superior

Em geral, uma aproximação de Runge-Kutta de $y' = f(x, y)$ em $[x_i, x_{i+1}]$, é uma equação de diferenças da forma

$$y_{i+1} = y_i + h[\eta_1 f(\xi_1, \mu_1) + \eta_2 f(\xi_2, \mu_2) + \dots + \eta_k f(\xi_k, \mu_k)], \quad (1.4.18)$$

onde $x_i \leq \xi_j \leq x_{i+1}$, $j = 1, 2, \dots, k$. Naturalmente, (1.4.7) é um caso especial de (1.4.18).

Várias fórmulas de Runge-Kutta de ordens superiores foram desenvolvidas, como por exemplo o método de terceira ordem dado por:

$$\begin{aligned} K_0 &= hf(x_i, y_i); \\ K_1 &= hf\left(x_i + \frac{h}{2}, y_i + \frac{K_0}{2}\right); \\ K_2 &= hf\left(x_i + \frac{3}{4}h, y_i + \frac{3}{4}K_1\right); \\ y_{i+1} &= y_i + \frac{2}{9}K_0 + \frac{1}{3}K_1 + \frac{4}{9}K_2. \end{aligned}$$

Pode-se considerar que uma das fórmulas mais práticas, visando a precisão e tempo computacional, é a fórmula de Runge-Kutta de quarta ordem. Frequentemente, é a primeira fórmula a ser aplicada na análise numérica profissional para aproximar a solução de um PVI, sendo esta, dada pela fórmula:

$$\begin{aligned} K_0 &= hf(x_i, y_i); \\ K_1 &= hf\left(x_i + \frac{h}{2}, y_i + \frac{K_0}{2}\right); \\ K_2 &= hf\left(x_i + \frac{h}{2}, y_i + \frac{K_1}{2}\right); \\ K_3 &= hf(x_i + h, y_i + K_2); \\ y_{i+1} &= y_i + \frac{1}{6}(K_0 + 2K_1 + 2K_2 + K_3). \end{aligned}$$

Esta fórmula, está de acordo com a expansão de Taylor da solução exata até os termos da ordem de h^4 .

1.5 Método Previsor-Corretor

Os métodos implícitos são usualmente implementados utilizando-se um processo iterativo que requer um valor inicial. Este valor, pode ser obtido através de um método explícito. Consideremos um par de métodos de passo múltiplo, dados, conforme a Definição 1.2, por:

$$\begin{aligned} y_{i+1} &= \bar{\alpha}_{m-1} y_i + \bar{\alpha}_{m-2} y_{i-1} + \dots + \bar{\alpha}_0 y_{i+1-m} \\ &+ h[\bar{\beta}_{m+1} f(x_i, y_i) + \bar{\beta}_m f(x_{i-1}, y_{i-1}) + \dots + \bar{\beta}_0 f(x_{i+1-m}, y_{i+1-m})], \end{aligned} \tag{1.5.19}$$

$$\begin{aligned} y_{i+1} &= \alpha_{m-1} y_i + \alpha_{m-2} y_{i-1} + \dots + \alpha_0 y_{i+1-m} \\ &+ h[\beta_m f(x_{i+1}, y_{i+1}) + \beta_{m-1} f(x_i, y_i) + \dots + \beta_0 f(x_{i+1-m}, y_{i+1-m})], \end{aligned} \quad (1.5.20)$$

dos quais o primeiro (conhecido como um previsor) é explícito e o segundo (conhecido como um corretor) é implícito. Então, os dois métodos usados juntos constituem um par previsor-corretor. Por exemplo, (1.5.19) pode ser usado para calcular uma quantidade $y_{i+1}^{(0)}$ (o valor preditado) e (1.5.20) pode ser modificado substituindo-se o termo $h\beta_m f(x_{i+1}, y_{i+1})$, do lado direito, por $h\beta_m f(x_{i+1}, y_{i+1}^{(0)})$. O valor y_{i+1} calculado por esta versão modificada de (1.5.20) (o valor corrigido) é usado nos passos seguintes. A esta forma de aplicação do par de métodos (1.5.19), (1.5.20) denominamos modo *PECE* em que este nome representa os quatro estágios do cálculo de y_{i+1} e de $f(x_{i+1}, y_{i+1})$ assim descritos:

1. o cálculo de $y_{i+1}^{(0)}$ como o valor preditado (*P*);
2. a avaliação (*E*) de $f(x_{i+1}, y_{i+1}^{(0)})$;
3. o cálculo de y_{i+1} como o valor corrigido (*C*) e,
4. a avaliação (*E*) de $f(x_{i+1}, y_{i+1})$ para ser usada nos passos seguintes.

Por outro lado, o método previsor-corretor pode ser usado no modo *PEC* em que o passo (4) é omitido. Neste caso, a solução é estabelecida pelas equações

$$\begin{aligned} y_{i+1}^{(0)} &= \bar{\alpha}_{m-1} y_i + \bar{\alpha}_{m-2} y_{i-1} + \dots + \bar{\alpha}_0 y_{i+1-m} \\ &+ h[\bar{\beta}_{m+1} f(x_i, y_i^{(0)}) + \bar{\beta}_m f(x_{i-1}, y_{i-1}^{(0)}) + \dots + \bar{\beta}_0 f(x_{i+1-m}, y_{i+1-m}^{(0)})], \\ y_{i+1} &= \alpha_{m-1} y_i + \alpha_{m-2} y_{i-1} + \dots + \alpha_0 y_{i+1-m} \\ &+ h[\beta_m f(x_{i+1}, y_{i+1}^{(0)}) + \beta_{m-1} f(x_i, y_i^{(0)}) + \dots + \beta_0 f(x_{i+1-m}, y_{i+1-m}^{(0)})]. \end{aligned}$$

Também, temos os modos *PECEC* e *PECECE* nos quais uma segunda aplicação da fórmula do corretor é feita, usando no termo $h\beta_m f(x_{i+1}, y_{i+1})$ o valor de y_{i+1} encontrado no primeiro uso do corretor.

Há uma literatura extensa sobre os métodos previsor-corretor, incluindo comparações de eficiência para modos diferentes. No presente trabalho, faremos uma simples observação a respeito da precisão local de tais métodos.

Se $y_{i+1}^{(1)}, y_{i+1}^{(2)}, \dots, y_{i+1}^{(N)}$ denotam a seqüência de N valores corrigidos no método $P(EC)^N E$ ou $P(EC)^N$ (note que $(EC)^N$ denota a seqüência dos pares EC), e se $f_{i+1}^{(n)}$ denota $f(x_{i+1}, y_{i+1}^{(n)})$, $n = 1, 2, \dots, N$, então para o método $P(EC)^N E$ temos

$$y_{i+1}^{(0)} = \sum_{j=1}^k \bar{\alpha}_{m-j} y_{i+1-j}^{(N)} + h \sum_{j=1}^k \bar{\beta}_{m-j} f_{i+1-j}^{(N)},$$

e para $n = 1, 2, \dots, N$

$$y_{i+1}^{(n)} = \sum_{j=1}^k \alpha_{m-j} y_{i+1-j}^{(N)} + h[\beta_m f_{i+1}^{(n-1)} + \sum_{j=1}^k \beta_{m-j} f_{i+1-j}^{(N)}],$$

enquanto que no método $P(EC)^N$,

$$y_{i+1}^{(0)} = \sum_{j=1}^k \bar{\alpha}_{m-j} y_{i+1-j}^{(N)} + h \sum_{j=1}^k \bar{\beta}_{m-j} f_{i+1-j}^{(N-1)},$$

e para $n = 1, 2, \dots, N$

$$y_{i+1}^{(n)} = \sum_{j=1}^k \alpha_{m-j} y_{i+1-j}^{(N)} + h[\beta_m f_{i+1}^{(n-1)} + \sum_{j=1}^k \beta_{m-j} f_{i+1-j}^{(N-1)}].$$

É apropriado estimar $|y_{i+1}^{(N)} - y(x_{i+1})|$ para cada um destes esquemas e $|hf_{i+1}^{(N)} - hf(x_{i+1}, y(x_{i+1}))|$ ou $|hf_{i+1}^{(N-1)} - hf(x_{i+1}, y(x_{i+1}))|$ para os métodos $P(EC)^N E$ e $P(EC)^N$ respectivamente, visto que estes representarão contribuições aos erros que são carregados nos passos posteriores.

Teorema 1.5. *Se (1.5.19) é de ordem $\bar{q}$ e (1.5.20) é de ordem q e se $y_{i+1-j} = y(x_{i+1-j})$, $f_{i+1-j} = f(x_{i+1-j}, y(x_{i+1-j}))$ para $j = 1, 2, \dots, m$ e se f satisfaz uma condição de Lipschitz, então para um método $P(EC)^N E$ ou $P(EC)^N$*

$$\begin{aligned} |y_{i+1}^N - y(x_{i+1})| &= O(h^{\bar{q}+1}), \\ |hf_{i+1}^{N-1} - hf(x_{i+1}, y(x_{i+1}))| &= O(h^{\bar{q}+1}), \\ |hf_{i+1}^N - hf(x_{i+1}, y(x_{i+1}))| &= O(h^{\bar{q}+2}), \end{aligned}$$

em que $\tilde{q} = \min(\bar{q} + N, q)$. (BUTCHER, [10], p.142-143)

Nas próximas seções, 1.6 e 1.7, apresentamos dois métodos implícitos, que serão usados para a resolução do modelo *BBS*. No cálculo da equação implícita usamos um processo iterativo sendo o valor inicial calculado através de um método explícito. Este procedimento constitui um par previsor-corretor.

1.6 Integração Trapezoidal

Este é um método para a integração numérica de equações diferenciais ordinárias de primeira ordem, ou sistemas, que tem precisão $O(h^2)$.

Para a equação diferencial de primeira ordem $y' = f(x, y)$, podemos defini-lo implicitamente pela fórmula recursiva

$$y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i) + f(x_{i+1}, y_{i+1})], \quad (1.6.21)$$

em que $h = x_{i+1} - x_i$.

Dado um valor inicial $y_0 = \eta$ e uma partição π , a fórmula (1.6.21) define uma seqüência de valores $y_{i+1} = y_\pi(x_{i+1})$, ou seja, uma tabela de funções descrevendo aproximadamente a solução da equação diferencial $y' = f(x, y)$ satisfazendo o valor inicial $y(a) = \eta$.

Note que, quando $f = f(x)$, a fórmula (1.6.21) é equivalente à fórmula de quadratura trapezoidal dada por

$$\int_a^b F(x)dx \approx \tau_\pi[F] = \sum_{i=1}^n [F(x_i) + F(x_{i+1})] \frac{h}{2}.$$

1.6.1 Método de Euler Melhorado

O método trapezoidal é muito conveniente para obter soluções aproximadas para equações diferenciais lineares, as quais podem ser resolvidas algebricamente para y_{i+1} . Mas, por outro lado, torna-se inadequado quando temos uma equação diferencial não linear, pois encontramos dificuldades em resolver (1.6.21)

para y_{i+1} . Em geral, a expressão (1.6.21) não define os valores y_{i+1} recursivamente e sim de maneira implícita. Para determinar cada y_{i+1} , temos que resolver uma equação do tipo (1.6.21) em que y_{i+1} é desconhecida e y_i tenha sido previamente determinada.

Podemos fazer isto através de um processo iterativo usando o par previsor-corretor, onde o previsor é o método de Euler e o corretor é a regra do trapézio.

Temos, então, o par

$$y_{i+1} = y_i + hf(x_i, y_i),$$

$$y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i) + f(x_{i+1}, y_{i+1})].$$

Ao invés de resolvermos a equação implícita (1.6.21) através de várias iterações, normalmente obtemos uma melhor precisão, com o mesmo esforço computacional, usando uma malha mais fina e parando depois de uma ou duas iterações. Se pararmos depois de uma simples iteração, temos o *Método de Euler Melhorado*, que é adequado para muitos problemas não lineares que requerem precisão moderada.

1.7 Método de Euler Implícito

O método que define y_{i+1} (uma aproximação para $y(x_{i+1})$) em termos de y_i pela equação

$$y_{i+1} - hf(x_{i+1}, y_{i+1}) = y_i, \quad (1.7.22)$$

tem uma relação evidente com o método de Euler e é conhecido como método de Euler implícito.

O cálculo de y_{i+1} em (1.7.22), requer a solução para z da equação

$$z - hf(x_{i+1}, z) = y_i, \quad (1.7.23)$$

e esta pode ser encontrada, por exemplo, usando um esquema iterativo

$$z = y_i + hf(x_{i+1}, z),$$

onde podemos calcular $z^{(0)}$ através do método de Euler, como definimos na seção anterior. Se f satisfaz uma condição de Lipschitz com constante L e h satisfaz $hL < 1$, então, pelo Teorema da Contração (ORTEGA, [19], p.153), este esquema converge para a solução única de (1.7.23).

Definimos aqui os métodos que serão utilizados na aproximação do modelo *BBS*, o qual apresentamos no próximo capítulo.

Capítulo 2

Modelo BBS para a Eliminação de Ruídos em Imagens

2.1 Representação de uma Imagem Digital

Pode-se considerar uma imagem como sendo uma função limitada $u : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$, $n = 2$, para imagens bidimensionais, ou $n = 3$, para imagens tridimensionais. No caso bidimensional, o qual será tratado aqui, a função $u(x, y)$ representa a imagem original que pretendemos reconstruir a partir de uma imagem inicial $I(x, y)$ definida por $I(x, y) = u(x, y) + \eta(x, y)$, em que x e y referem-se às coordenadas espaciais de brilho (ou gradação de cinza) de qualquer par (x, y) e $\eta(x, y)$ representa o ruído.

Uma imagem digital $u(x, y)$ é resultante da discretização em coordenadas espaciais e em intensidade luminosa. É representada por uma matriz de pontos, denominados elementos de imagem, elementos de figura ou pixels.

Embora as dimensões de uma imagem digital variem de acordo com a aplicação a que se refere, é conveniente representá-la sob a forma de uma matriz quadrada composta por elementos graduados em escala de cinza, com dimensões escalonadas em potências de 2.

2.2 Modelos para Eliminação de Ruídos em Imagens

Nos últimos anos, o uso de equações diferenciais parciais no processo de restauração de imagens tem sido amplamente difundido. Isto possibilitou o surgimento de vários modelos matemáticos para eliminação de ruídos e segmentação de imagens digitais. Dentre estes modelos, partindo do princípio de que o uso da difusão isotrópica não preservava a localização das bordas da imagem, Malik e Perona [20] desenvolveram um modelo através de uma equação de difusão anisotrópica objetivando preservar exatamente a localização destas bordas.

A formulação deste modelo é expressa por:

$$\begin{aligned} u_t &= \operatorname{div}(g(|\nabla u|)\nabla u), \text{ em } \Omega \times \mathbb{R}_+ \\ u(x, y, 0) &= I(x, y), \end{aligned} \tag{2.2.1}$$

onde $g = \frac{1}{1+|\nabla u|^2}$ é uma função suave não-crescente, tal que $g(0) = 1$, $g(s) \geq 0$, e $g(s) \rightarrow 0$ quando $s \rightarrow \infty$.

A idéia é que se $|\nabla u|$ é grande, então a difusão será pequena, e portanto a localização exata das bordas será preservada.

Esse modelo teve uma forte contribuição no que diz respeito à reconstrução de imagens. Porém, apresentou problemas práticos e teóricos, como por exemplo, quando aplicado a imagens com alto nível de ruído, produz um valor bem grande para $|\nabla u|$. Isto implica em uma função g próxima de zero, e como consequência a imagem reconstruída retém quase todo o ruído da imagem original.

Frente a este problema, Alvarez, Lions e Morel [1] propuseram um modelo a partir da idéia de Malik e Perona [20] e da equação de difusão degenerada, também conhecida como fluxo de curvatura média, dada por

$$u_t = |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right). \tag{2.2.2}$$

A combinação do modelo de Malik e Perona [20] com a equação (2.2.2) resultou no seguinte modelo:

$$u_t = g(|\nabla(G_\sigma * u)|)|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right),$$

com $u(x, y, 0) = I(x, y)$ sendo a imagem inicial com ruído e $\frac{\partial u}{\partial n} \Big|_{\partial \Omega \times \mathbb{R}_+} = 0$.

Nordström [17] também propôs uma modificação no modelo de Malik e Perona [20]. Ele introduziu um termo forçante $(u - I)$ em (2.2.1) obtendo a equação

$$u_t - \operatorname{div}(g(|\nabla u|)\nabla u) = I - u. \quad (2.2.3)$$

Este termo forçante tem a propriedade de manter $u(x, y, t)$ próxima à imagem inicial $I(x, y)$. Assim, a equação (2.2.3) tem a vantagem de possuir um estado estacionário não trivial, e isto exclui a necessidade de um processo de seleção para o tempo de parada. Por outro lado, tem a desvantagem de não eliminar satisfatoriamente os ruídos de uma imagem.

A principal meta dos pesquisadores, no processo de eliminação de ruídos em imagens, é encontrar a melhor maneira de suavizar uma imagem ruidosa sem que esta perca as informações nas bordas. Diante deste ideal e baseados na fundamentação teórica apresentada pelos modelos descritos anteriormente, Barcelos, Boaventura e Silva Jr. [7] apresentaram, recentemente, um modelo, o qual será tratado na próxima seção.

2.3 Modelo BBS

O modelo proposto por Barcelos, Boaventura e Silva Jr. [7], é um dos modelos matemáticos apresentados para a eliminação de ruídos e segmentação de imagens digitais via equação de difusão. Este modelo apresenta resultados satisfatórios uma vez que elimina ruídos de uma imagem, efetuando suavização intensa nas regiões homogêneas e superficial nas regiões de contorno.

Ele é descrito pela seguinte equação parabólica

$$u_t = g|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) - \lambda(1 - g)(u - I), \quad (2.3.4)$$

$$x \in \Omega, \quad t > 0, \quad \lambda \text{ constante},$$

em que

- (i) $I(x, y)$ representa a imagem original com ruído;
- (ii) $u(x, y, t)$ é a versão suavizada de I no instante t ;
- (iii) $g = g(|\nabla(G_\sigma * u)|)$, tal que G_σ é um núcleo de convolução (aqui, uma função Gaussiana) e $(\nabla(G_\sigma * u))$ é a estimativa local de ∇u usada para eliminação de ruídos. A função $g(s) \geq 0$ é real, não-crescente, e satisfaz $g(0) = 1$ e $g(s) \rightarrow 0$ à medida que $s \rightarrow \infty$.

A primeira parcela do lado direito da equação (2.3.4) representa o modelo de Alvarez, Lions e Morel [1]. A segunda é formada pelo termo forçante $(u - I)$, sugerido por Nordström [17], multiplicado pelo termo regularizador $\lambda(1 - g)$, que é denominado seletor de moderação. Esta parcela de (2.3.4) age de forma seletiva recuperando as características iniciais da imagem I com mais intensidade nas regiões de fronteira, visto que nessas regiões $g \approx 0$ o que implica em $(1 - g) \approx 1$. Por outro lado, nas regiões homogêneas, $g \approx 1$ de modo que $(1 - g) \approx 0$ e, conseqüentemente, o termo forçante será inexpressivo, o que possibilita uma melhor suavização na imagem [22].

Definido o modelo, fazemos agora a discretização dos termos envolvidos na equação (2.3.4).

2.3.1 Termo de Difusão

Na equação (2.3.4), o termo de difusão é dado por

$$|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right).$$

Considerando que

$$\nabla u = \left[\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \right],$$

temos

$$|\nabla u| = \sqrt{u_x^2 + u_y^2},$$

$$\frac{\nabla u}{|\nabla u|} = \left[\frac{u_x}{\sqrt{u_x^2 + u_y^2}}, \frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right]$$

e

$$\operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right).$$

Logo, obtemos

$$\begin{aligned} |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) &= \sqrt{u_x^2 + u_y^2} \left[\frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) \right] \\ &= \sqrt{u_x^2 + u_y^2} \left[\frac{u_{xx} \sqrt{u_x^2 + u_y^2} - u_x \frac{\partial}{\partial x} (\sqrt{u_x^2 + u_y^2})}{u_x^2 + u_y^2} \right. \\ &\quad \left. + \frac{u_{yy} \sqrt{u_x^2 + u_y^2} - u_y \frac{\partial}{\partial y} (\sqrt{u_x^2 + u_y^2})}{u_x^2 + u_y^2} \right]. \end{aligned}$$

Ainda,

$$\frac{\partial}{\partial x} \left(\sqrt{u_x^2 + u_y^2} \right) = \frac{1}{2} (u_x^2 + u_y^2)^{-\frac{1}{2}} (2u_x u_{xx} + 2u_y u_{yx})$$

e

$$\frac{\partial}{\partial y} \left(\sqrt{u_x^2 + u_y^2} \right) = \frac{1}{2} (u_x^2 + u_y^2)^{-\frac{1}{2}} (2u_x u_{xy} + 2u_y u_{yy}).$$

Assim,

$$\begin{aligned} |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) &= \sqrt{u_x^2 + u_y^2} \frac{u_{xx} \sqrt{u_x^2 + u_y^2} - u_x \left[\frac{1}{2\sqrt{u_x^2 + u_y^2}} (2u_x u_{xx} + 2u_y u_{yx}) \right]}{u_x^2 + u_y^2} \\ &\quad + \sqrt{u_x^2 + u_y^2} \frac{u_{yy} \sqrt{u_x^2 + u_y^2} - u_y \left[\frac{1}{2\sqrt{u_x^2 + u_y^2}} (2u_x u_{xy} + 2u_y u_{yy}) \right]}{u_x^2 + u_y^2} \\ &= \frac{u_{xx}(u_x^2 + u_y^2) - u_x(u_x u_{xx} + u_y u_{yx})}{u_x^2 + u_y^2} + \frac{u_{yy}(u_x^2 + u_y^2) - u_y(u_x u_{xy} + u_y u_{yy})}{u_x^2 + u_y^2} \\ &= \frac{u_{xx}u_x^2 + u_{xx}u_y^2 - u_x^2 u_{xx} - u_x u_y u_{yx} + u_{yy}u_x^2 + u_{yy}u_y^2 - u_y u_x u_{xy} - u_y^2 u_{yy}}{u_x^2 + u_y^2} \\ &= \frac{u_{xx}u_y^2 - 2u_x u_y u_{xy} + u_{yy}u_x^2}{u_x^2 + u_y^2}. \end{aligned}$$

Logo,

$$|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{u_x^2 u_{yy} - 2u_x u_y u_{xy} + u_y^2 u_{xx}}{u_x^2 + u_y^2}. \quad (2.3.5)$$

A aproximação das derivadas presentes em (2.3.5) será desenvolvida através do método de diferenças finitas que apresentamos a seguir.

2.3.2 Método de Diferenças Finitas

No método de diferenças finitas, as derivadas existentes na equação diferencial de interesse são aproximadas utilizando-se uma expansão truncada da série de Taylor. Assim, a equação algébrica resultante de tal aproximação é expressa

em termos dos valores da função incógnita em alguns pontos específicos da malha utilizada, chamados pontos nodais.

Como queremos aproximar uma função de duas variáveis, vamos usar o Teorema 1.4 enunciado no capítulo anterior. Para tanto, consideremos uma malha regular de passo h , tal que a vizinhança do ponto (x, y) é representada a seguir:

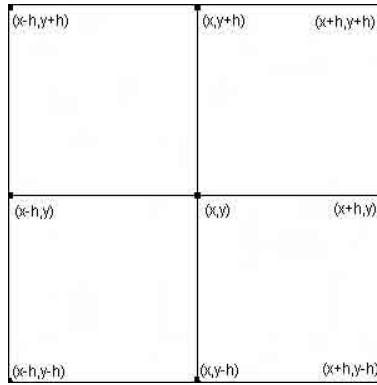


Figura 2.1: Região dos pontos vizinhos a (x, y) .

Tomando $n = 2$ em (1.4.9), encontramos:

$$u(x+h, y) = u(x, y) + hu_x(x, y) + \frac{h^2}{2!}u_{xx}(x, y) + \frac{h^3}{3!}u_{xxx}(\xi_1, y),$$

$$\xi_1 \in (x, x+h), \quad (2.3.6)$$

e

$$u(x-h, y) = u(x, y) - hu_x(x, y) + \frac{h^2}{2!}u_{xx}(x, y) - \frac{h^3}{3!}u_{xxx}(\xi_2, y),$$

$$\xi_2 \in (x-h, x). \quad (2.3.7)$$

Subtraindo (2.3.7) de (2.3.6), obtemos:

$$u_x(x, y) = \frac{u(x+h, y) - u(x-h, y)}{2h} + O(h^2). \quad (2.3.8)$$

Agora, tomando $n = 3$ em (1.4.9), temos:

$$u(x+h, y) = u(x, y) + hu_x(x, y) + \frac{h^2}{2!}u_{xx}(x, y) + \frac{h^3}{3!}u_{xxx}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_3, y),$$

$$\xi_3 \in (x, x+h), \quad (2.3.9)$$

e

$$u(x-h, y) = u(x, y) - hu_x(x, y) + \frac{h^2}{2!}u_{xx}(x, y) - \frac{h^3}{3!}u_{xxx}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_4, y),$$

$$\xi_4 \in (x-h, x). \quad (2.3.10)$$

Somando (2.3.9) e (2.3.10), obtemos:

$$u_{xx}(x, y) = \frac{u(x+h, y) - 2u(x, y) + u(x-h, y)}{h^2} + O(h^2).$$

Analogamente, para $n = 2$, temos:

$$u(x, y+h) = u(x, y) + hu_y(x, y) + \frac{h^2}{2!}u_{yy}(x, y) + \frac{h^3}{3!}u_{yyy}(x, \eta_1),$$

$$\eta_1 \in (y, y+h),$$

e

$$u(x, y-h) = u(x, y) - hu_y(x, y) + \frac{h^2}{2!}u_{yy}(x, y) - \frac{h^3}{3!}u_{yyy}(x, \eta_2),$$

$$\eta_2 \in (y-h, y),$$

que implicam em

$$u_y(x, y) = \frac{u(x, y+h) - u(x, y-h)}{2h} + O(h^2). \quad (2.3.11)$$

E para $n = 3$,

$$u(x, y+h) = u(x, y) + hu_y(x, y) + \frac{h^2}{2!}u_{yy}(x, y) + \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{yyyy}(x, \eta_3),$$

$$\eta_3 \in (y, y+h),$$

e

$$u(x, y-h) = u(x, y) - hu_y(x, y) + \frac{h^2}{2!}u_{yy}(x, y) - \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{yyyy}(x, \eta_4),$$

$$\eta_4 \in (y-h, y),$$

que implicam em

$$u_{yy}(x, y) = \frac{u(x, y + h) - 2u(x, y) + u(x, y - h)}{h^2} + O(h^2).$$

Também,

$$\begin{aligned} u(x + h, y + h) &= u(x, y) + hu_x(x, y) + hu_y(x, y) + h^2u_{xy}(x, y) + \frac{h^2}{2!}u_{xx}(x, y) \\ &+ \frac{h^2}{2!}u_{yy}(x, y) + \frac{h^3}{3!}u_{xxx}(x, y) + \frac{h^3}{2!}u_{xxy}(x, y) + \frac{h^3}{2!}u_{xyy}(x, y) \\ &+ \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_5, \eta_5) + \frac{h^4}{3!}u_{xxxxy}(\xi_5, \eta_5) \\ &+ \frac{h^4}{4}u_{xxxyy}(\xi_5, \eta_5) + \frac{h^4}{3!}u_{xyyyy}(\xi_5, \eta_5) + \frac{h^4}{4!}u_{yyyyy}(\xi_5, \eta_5), \\ &\xi_5 \in (x, x + h) \text{ e } \eta_5 \in (y, y + h); \end{aligned}$$

$$\begin{aligned} u(x - h, y - h) &= u(x, y) - hu_x(x, y) - hu_y(x, y) + h^2u_{xy}(x, y) + \frac{h^2}{2!}u_{xx}(x, y) \\ &+ \frac{h^2}{2!}u_{yy}(x, y) - \frac{h^3}{3!}u_{xxx}(x, y) - \frac{h^3}{2!}u_{xxy}(x, y) - \frac{h^3}{2!}u_{xyy}(x, y) \\ &- \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_6, \eta_6) + \frac{h^4}{3!}u_{xxxxy}(\xi_6, \eta_6) \\ &+ \frac{h^4}{4}u_{xxxyy}(\xi_6, \eta_6) + \frac{h^4}{3!}u_{xyyyy}(\xi_6, \eta_6) + \frac{h^4}{4!}u_{yyyyy}(\xi_6, \eta_6), \\ &\xi_6 \in (x - h, x) \text{ e } \eta_6 \in (y - h, y); \end{aligned}$$

$$\begin{aligned} u(x + h, y - h) &= u(x, y) + hu_x(x, y) - hu_y(x, y) - h^2u_{xy}(x, y) + \frac{h^2}{2!}u_{xx}(x, y) \\ &+ \frac{h^2}{2!}u_{yy}(x, y) + \frac{h^3}{3!}u_{xxx}(x, y) - \frac{h^3}{2!}u_{xxy}(x, y) + \frac{h^3}{2!}u_{xyy}(x, y) \\ &- \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_7, \eta_7) - \frac{h^4}{3!}u_{xxxxy}(\xi_7, \eta_7) \\ &+ \frac{h^4}{4}u_{xxxyy}(\xi_7, \eta_7) - \frac{h^4}{3!}u_{xyyyy}(\xi_7, \eta_7) + \frac{h^4}{4!}u_{yyyyy}(\xi_7, \eta_7), \\ &\xi_7 \in (x, x + h) \text{ e } \eta_7 \in (y - h, y); \end{aligned}$$

$$\begin{aligned}
u(x-h, y+h) &= u(x, y) - hu_x(x, y) + hu_y(x, y) - h^2u_{xy}(x, y) + \frac{h^2}{2!}u_{xx}(x, y) \\
&+ \frac{h^2}{2!}u_{yy}(x, y) - \frac{h^3}{3!}u_{xxx}(x, y) + \frac{h^3}{2!}u_{xxy}(x, y) - \frac{h^3}{2!}u_{xyy}(x, y) \\
&+ \frac{h^3}{3!}u_{yyy}(x, y) + \frac{h^4}{4!}u_{xxxx}(\xi_8, \eta_8) - \frac{h^4}{3!}u_{xxxy}(\xi_8, \eta_8) \\
&+ \frac{h^4}{4}u_{xxyy}(\xi_8, \eta_8) - \frac{h^4}{3!}u_{xyyy}(\xi_8, \eta_8) + \frac{h^4}{4!}u_{yyyy}(\xi_8, \eta_8), \\
&\xi_8 \in (x-h, x) \text{ e } \eta_8 \in (y, y+h).
\end{aligned}$$

Logo,

$$\begin{aligned}
u_{xy}(x, y) &= \frac{1}{4h^2} \left[u(x+h, y+h) + u(x-h, y-h) - u(x-h, y+h) \right. \\
&\left. - u(x+h, y-h) \right] + O(h^2).
\end{aligned}$$

Em resumo, para cada ponto no interior do intervalo onde a equação diferencial está definida, denotando por $u_{i,j}$ a aproximação de $u(x_i, y_j)$, temos as seguintes aproximações para as derivadas de primeira e segunda ordens:

$$u_x(x_i, y_j) \approx \frac{u_{i+1,j} - u_{i-1,j}}{2h};$$

$$u_y(x_i, y_j) \approx \frac{u_{i,j+1} - u_{i,j-1}}{2h};$$

$$u_{xx}(x_i, y_j) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2};$$

$$u_{yy}(x_i, y_j) \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2};$$

$$u_{xy}(x_i, y_j) \approx \frac{1}{4h^2} \left[u_{i+1,j+1} + u_{i-1,j-1} - u_{i-1,j+1} - u_{i+1,j-1} \right].$$

2.3.3 Discretização da Função $g(|\nabla(G_\sigma * u)|)$

A função g presente na equação (2.3.4) é definida por:

$$g(|\nabla(G_\sigma * u)|) = \frac{1}{1 + k|\nabla(G_\sigma * u)|^2}, \quad k \text{ constante.}$$

E, para a função G , consideramos a seguinte definição:

Definição 2.1. *Seja $\rho : \mathbb{R}^n \rightarrow \mathbb{R}$. O espaço escala Gaussiano de ρ é uma função $T_\rho : \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}$, dada por*

$$T_\rho(x, t) = \rho * G_t(x),$$

em que

$$G_t(x) = \frac{1}{(2\pi at)^{\frac{n}{2}}} e^{\frac{-(x_1^2 + \dots + x_n^2) - \mu}{2at}}$$

é a distribuição Gaussiana de dimensão n com média μ e variância at , onde a é uma constante positiva. O desvio-padrão é $\sigma = \sqrt{at}$ e $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$. O parâmetro t é chamado de escala.

Note que, tomando $\sigma = \sqrt{at}$, podemos escrever

$$G_\sigma(x) = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} e^{\frac{-(x_1^2 + \dots + x_n^2) - \mu}{2\sigma^2}}.$$

A função $T_\rho(x, t)$ será avaliada numericamente nos pontos de uma malha regular de passo h e na sua implementação numérica será usada a definição seguinte para o produto de convolução.

Definição 2.2. *Sejam $f, \rho \in SC_{per}(2L)$, $SC_{per}(2L)$ o espaço das funções seccionalmente periódicas de período $2L$. A convolução de f e ρ é a função $f * \rho : \mathbb{R} \rightarrow \mathbb{R}$ definida por*

$$(f * \rho)(x) = \frac{1}{2L} \int_{-L}^L f(t) \rho(x - t) dt. \quad (2.3.12)$$

Esse operador é linear, invariante por translações e, impondo algumas restrições aos sinais, a integral de convolução converge.

Desde que a malha usada para as discretizações dos termos envolvidos na equação diferencial do modelo *BBS* é definida sobre os números naturais, com $h = \Delta x = \Delta y = 1$ um valor que será esclarecido na Seção 3.1, o valor de L em (2.3.12) é igual a 1. Assim, devemos calcular a convolução das funções f e ρ , em uma vizinhança do ponto $u_{i,j}$. Consideremos essa vizinhança como sendo a região dos pontos vizinhos a (x_i, y_j) dada pela Figura 2.1. Note que, para cada ponto $(x+ah, y+bh)$, $a, b = -1, 0, 1$, da figura, estamos associando o ponto (x_{i+h}, y_{j+k}) , $k, h = -1, 0, 1$.

Assim, o cálculo do produto de convolução em relação ao ponto (x_i, y_j) se dá através do cálculo da seguinte integral

$$(f * \rho)(x_i, y_j) = \frac{1}{2} \int_{-1}^1 \frac{1}{2} \int_{-1}^1 f(x_i - x, y_j - y) \rho(x, y) dx dy.$$

Usando a regra de Simpson para o cálculo desta integral, conforme [22], considerando $f(x, y) = G_\sigma(x, y)$ e $\rho(x, y) = u(x, y)$, obtemos:

$$\begin{aligned} (G_\sigma * u)(i, j) \approx & \frac{1}{36} \left\{ G_\sigma(1, 1)u_{i-1, j-1} + G_\sigma(-1, 1)u_{i+1, j-1} + G_\sigma(1, -1)u_{i-1, j+1} \right. \\ & + G_\sigma(-1, -1)u_{i+1, j+1} + 16 G_\sigma(0, 0)u_{i, j} + 4[G_\sigma(0, 1)u_{i, j-1} \\ & \left. + G_\sigma(1, 0)u_{i-1, j} + G_\sigma(-1, 0)u_{i+1, j} + G_\sigma(0, -1)u_{i, j+1}] \right\}. \end{aligned}$$

Seja $(G_\sigma * u)(x, y) = cv(x, y)$. Para o cálculo das derivadas parciais de primeira ordem de cv , utilizamos o procedimento análogo ao que foi usado para encontrar (2.3.8) e (2.3.11). Assim,

$$\frac{\partial cv(x, y)}{\partial x} \approx \frac{cv(x+1, y) - cv(x-1, y)}{2} = d_1$$

e

$$\frac{\partial cv(x, y)}{\partial y} \approx \frac{cv(x, y+1) - cv(x, y-1)}{2} = d_2.$$

Ainda,

$$\nabla(G_\sigma * u) = \left(\frac{\partial cv}{\partial x}, \frac{\partial cv}{\partial y} \right)$$

e

$$|\nabla(G_\sigma * u)| = \left[\left(\frac{\partial G_\sigma * u}{\partial x} \right)^2 + \left(\frac{\partial G_\sigma * u}{\partial y} \right)^2 \right]^{\frac{1}{2}}.$$

Logo, $|\nabla(G_\sigma * u)|^2 = d_1^2 + d_2^2$.

Conseqüentemente, a função g é discretizada por

$$g(|\nabla(G_\sigma * u)|) = \frac{1}{1 + k(d_1^2 + d_2^2)}, \text{ com } k \text{ constante.}$$

Na próxima seção, tratamos da estabilidade do esquema numérico utilizado para a solução do modelo *BBS*.

2.3.4 Estabilidade para o Método de Euler Aplicado ao Modelo BBS

Para a análise da estabilidade do método numérico

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^n), \quad \text{com } u_{i,j}^0 = I(x_i, y_j),$$

aplicado ao modelo *BBS*, usamos a definição seguinte, que pode ser encontrada em [3].

Definição 2.3. (*Estabilidade L^∞*)

Um esquema numérico é L^∞ -estável se para $c \leq u^n(x) \leq d$ tivermos $c \leq u^{n+1}(x) \leq d$, c e d números reais.

Denotando por ξ a direção ortogonal a ∇u , podemos representar o termo de difusão (2.3.5) por

$$u_{\xi\xi} = |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right).$$

Seja θ tal que $\xi = (-\text{sen } \theta, \cos \theta)$.

Assim, podemos tomar

$$\frac{\nabla u}{|\nabla u|} = (\cos \theta, \text{sen } \theta).$$

Logo,

$$u_x = \sqrt{u_x^2 + u_y^2} \cos \theta$$

e

$$u_y = \sqrt{u_x^2 + u_y^2} \text{sen } \theta.$$

A substituição desses dois termos em (2.3.5) produz

$$\begin{aligned} u_{\xi\xi} = & \frac{1}{u_x^2 + u_y^2} [(u_x^2 + u_y^2) \cos^2 \theta u_{yy} - 2(u_x^2 + u_y^2) \text{sen } \theta \cos \theta u_{xy} \\ & + (u_x^2 + u_y^2) \text{sen}^2 \theta u_{xx}]. \end{aligned}$$

Logo,

$$u_{\xi\xi} = \text{sen}^2 \theta u_{xx} - 2 \text{sen } \theta \cos \theta u_{xy} + \cos^2 \theta u_{yy}.$$

Aproximando u_{xx} , u_{xy} e u_{yy} pelas discretizações desenvolvidas na seção 2.3.2, obtemos:

$$\begin{aligned} u_{\xi\xi} = & \text{sen}^2 \theta u_{i+1,j} - 2 \text{sen}^2 \theta u_{i,j} + \text{sen}^2 \theta u_{i-1,j} - \frac{1}{2} \text{sen } \theta \cos \theta u_{i+1,j+1} \\ & - \frac{1}{2} \text{sen } \theta \cos \theta u_{i-1,j-1} + \frac{1}{2} \text{sen } \theta \cos \theta u_{i-1,j+1} + \frac{1}{2} \text{sen } \theta \cos \theta u_{i+1,j-1} \\ & + \cos^2 \theta u_{i,j+1} - 2 \cos^2 \theta u_{i,j} + \cos^2 \theta u_{i,j-1} \\ = & -2u_{i,j} + \text{sen}^2 \theta u_{i+1,j} + \text{sen}^2 \theta u_{i-1,j} + \cos^2 \theta u_{i,j+1} + \cos^2 \theta u_{i,j-1} \\ & - \frac{1}{2} \text{sen } \theta \cos \theta u_{i+1,j+1} - \frac{1}{2} \text{sen } \theta \cos \theta u_{i-1,j-1} \\ & + \frac{1}{2} \text{sen } \theta \cos \theta u_{i-1,j+1} + \frac{1}{2} \text{sen } \theta \cos \theta u_{i+1,j-1} \\ = & \sum_{k,l=-1}^1 \lambda_{k,l} u_{i+k,j+l}. \end{aligned}$$

Assim, o esquema numérico para a implementação do modelo *BBS* torna-se

$$u_{i,j}^{n+1} = u_{i,j}^n + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 \lambda_{k,l} u_{i+k,j+l}^n - \lambda(1 - g(u_{i,j}^n))(u_{i,j}^n - I_{i,j})\Delta t. \quad (2.3.13)$$

Propomos, então, o seguinte teorema para a estabilidade deste esquema, o qual baseia-se no resultado apresentado por Alvarez e Esclarin [2].

Teorema 2.1. *Seja*

$$u_{i,j}^{n+1} = u_{i,j}^n + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 \lambda_{k,l} u_{i+k,j+l}^n + f(u_{i,j}^n)\Delta t,$$

com

$$f(u_{i,j}^n) = -\lambda(1 - g(u_{i,j}^n))(u_{i,j}^n - I_{i,j})$$

tal que

$$|f(u_{i,j}^n)| \leq C \quad \text{se} \quad u_{i,j} \in [-b, b].$$

Então, $\forall M > b$, temos que para valores pequenos de Δt ,

$$\text{se } \forall i, j, \quad |u_{i,j}^n| < M \Rightarrow |u_{i,j}^{n+1}| \leq M.$$

Demonstração:

$$\begin{aligned} u_{i,j}^{n+1} &= u_{i,j}^n + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 \lambda_{k,l} u_{i+k,j+l}^n + f(u_{i,j}^n)\Delta t \\ &\leq b + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 \lambda_{k,l} u_{i+k,j+l}^n + f(u_{i,j}^n)\Delta t. \end{aligned}$$

Então,

$$\begin{aligned} |u_{i,j}^{n+1}| &\leq b + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| |u_{i+k,j+l}^n| + |f(u_{i,j}^n)|\Delta t \\ &< b + g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| M + C\Delta t. \end{aligned}$$

Supondo que essa última expressão seja menor que M , temos:

$$b + C\Delta t < M \left[1 - g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| \right]. \quad (2.3.14)$$

Dividindo (2.3.14) por $\left[1 - g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| \right]$, obtemos

$$\frac{b + C\Delta t}{\left[1 - g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| \right]} < M. \quad (2.3.15)$$

Quando $\Delta t \rightarrow 0$, $b \leq M$.

Logo, se Δt é suficientemente pequeno para (2.3.15) ser verdadeira, então $|u_{i,j}^{n+1}| \leq M$.

A passagem de (2.3.14) para (2.3.15), só é possível se a condição seguinte for satisfeita:

$$\left[1 - g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| \right] > 0.$$

Isso implica que

$$\begin{aligned} g(u_{i,j}^n)\Delta t \sum_{k,l=-1}^1 |\lambda_{k,l}| &< 1, \\ g(u_{i,j}^n)\Delta t &< \frac{1}{\sum_{k,l=-1}^1 |\lambda_{k,l}|}. \end{aligned}$$

Como

$$\sum_{k,l=-1}^1 |\lambda_{k,l}| = 4 + 2|\sin \theta \cos \theta|,$$

$$4 \leq \sum_{k,l=-1}^1 |\lambda_{k,l}| \leq 5.$$

Assim, temos

$$g(u_{i,j}^n)\Delta t < 0.2. \quad (2.3.16)$$

Para que (2.3.16) seja sempre válida, basta tomarmos $\Delta t < 0.2$. ■

Capítulo 3

Resultados

Neste capítulo, apresentamos os resultados experimentais obtidos da implementação numérica dos métodos de Euler, Runge-Kutta de quarta ordem, Euler melhorado e Euler implícito quando aplicados ao modelo *BBS*.

Consideramos, para teste, imagens sintéticas e imagens da vida real. As imagens originais utilizadas são de 256 x 256 pixels e 256 gradações de cinza. A implementação numérica foi realizada utilizando-se a linguagem de programação Fortran 95, e para a visualização das imagens, usamos o software Matlab.

Para obtermos a imagem inicial que será reconstruída, adicionamos um certo ruído à imagem original. Esse ruído foi determinado por um gerador numérico, de tal forma que a quantidade $\eta(i, j)$ adicionada à intensidade de cada pixel da imagem original, fosse determinada a partir da relação sinal-ruído (*Signal to Noise Ratio* - *SNR*) desejada, expressa em decibéis (dB) por

$$SNR = \frac{\text{Variância da imagem original}}{\text{Variância da imagem com ruído}} \text{ dB}.$$

3.1 Implementação Numérica

A solução numérica da EDP (2.3.4) é encontrada através do método de diferenças finitas. Para a implementação, consideramos a malha (veja Figura 2.1)

espaçial retangular (x_i, y_j) , tal que $x_i = i\Delta x$ e $y_j = j\Delta y$, $i, j = 0, 1, \dots, 255$, com incrementos espaciais $\Delta x = \Delta y$.

Como mencionado anteriormente, estamos considerando uma imagem como sendo uma matriz de dimensão $m \times n$, em que o valor de cada elemento está associado à tonalidade de cinza do pixel correspondente. A escala h corresponde ao espaçamento espacial nas direções horizontal e vertical. Como esse espaçamento corresponde à distância entre os pixels horizontais e verticais vizinhos, tomamos essa distância igual a 1, a qual corresponde a uma unidade de pixel. Desta forma, consideramos, neste trabalho, $h = 1$.

O incremento temporal é $\Delta t > 0$, tal que $t_n = n\Delta t$, $n = 0, 1, \dots$. Assim, $u_{i,j}^n = u(x_i, y_j, t_n)$.

Agora, considerando

$$\mathcal{L}(u) = g|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) - \lambda(1 - g)(u - I),$$

a aproximação do modelo *BBS*, pelos métodos apresentados no Capítulo 1, fica na forma:

Método de Euler

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^n), \quad \text{com} \quad u_{i,j}^0 = I(x_i, y_j),$$

que é o mesmo método iterativo usado na implementação numérica dos modelos para eliminação de ruídos em imagens.

Método de Runge-Kutta de quarta ordem

$$u_{i,j}^{n+1} = u_{i,j}^n + \frac{\Delta t}{6} [\mathcal{L}(u_{i,j}^n) + 2\mathcal{L}(u_{i,j}^{n_1}) + 2\mathcal{L}(u_{i,j}^{n_2}) + \mathcal{L}(u_{i,j}^{n_3})],$$

em que

$$\begin{aligned} u_{i,j}^{n_1} &= u_{i,j}^n + \frac{\Delta t}{2} \mathcal{L}(u_{i,j}^n); \\ u_{i,j}^{n_2} &= u_{i,j}^n + \frac{\Delta t}{2} \mathcal{L}(u_{i,j}^{n_1}); \\ u_{i,j}^{n_3} &= u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^{n_2}) \quad \text{e} \quad u_{i,j}^0 = I(x_i, y_j). \end{aligned}$$

Método de Euler Melhorado

$$u_{i,j}^{n+1} = u_{i,j}^n + \frac{\Delta t}{2} [\mathcal{L}(u_{i,j}^n) + \mathcal{L}(u_{i,j}^{n*+1})],$$

onde

$$u_{i,j}^{n*+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^n) \quad \text{e} \quad u_{i,j}^0 = I(x_i, y_j).$$

Método de Euler Implícito

$$u_{i,j}^{n+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^{n*+1}),$$

com

$$u_{i,j}^{n*+1} = u_{i,j}^n + \Delta t \mathcal{L}(u_{i,j}^n) \quad \text{e} \quad u_{i,j}^0 = I(x_i, y_j).$$

Para fazermos uma comparação entre os quatro métodos utilizados, plotamos o gráfico da linha 128 correspondente a cada imagem e definimos um tipo de erro, o qual chamamos de erro médio (e_m), dado pela expressão:

$$e_m = \frac{\sum_{i=0}^m \sum_{j=0}^m |u_{i,j}^n - u_{i,j}^*|}{(m+1) \times (m+1)},$$

onde $m = 255$, $u_{i,j}^n$ é a imagem reconstruída e $u_{i,j}^*$ é a imagem original sem ruído.

Os valores adotados para as constantes a e k , presentes na equação que descreve o modelo, são tais que a é o desvio-padrão da imagem inicial e k é uma constante arbitrária que apresentou melhor resultado. E, embora obtivemos, para a estabilidade do método, $\Delta t < 0.2$, não consideraremos apenas esse limite nos testes realizados. Isto porque, para algumas imagens, Δt maior produz imagens semelhantes às obtidas com Δt menor, mas com menor tempo computacional.

Inicialmente, consideramos uma imagem sintética com $SNR = -0.0263$ e $\Delta t = 0.4$.

Foram realizadas 159 iterações, com os seguintes valores: $k = 210121.4233222279$, $\lambda = 1$, $\sigma = 90.08585737861927$ e $a = 127.2438129862800$.

Para cada método, encontramos os seguintes erros médios:

Euler: $e_m = 5.22843985616820$;

Runge-Kutta de quarta ordem: $e_m = 5.948650755610874$;

Euler melhorado: $e_m = 5.474321007787193$;

Euler implícito: $e_m = 5.179005933727841$.

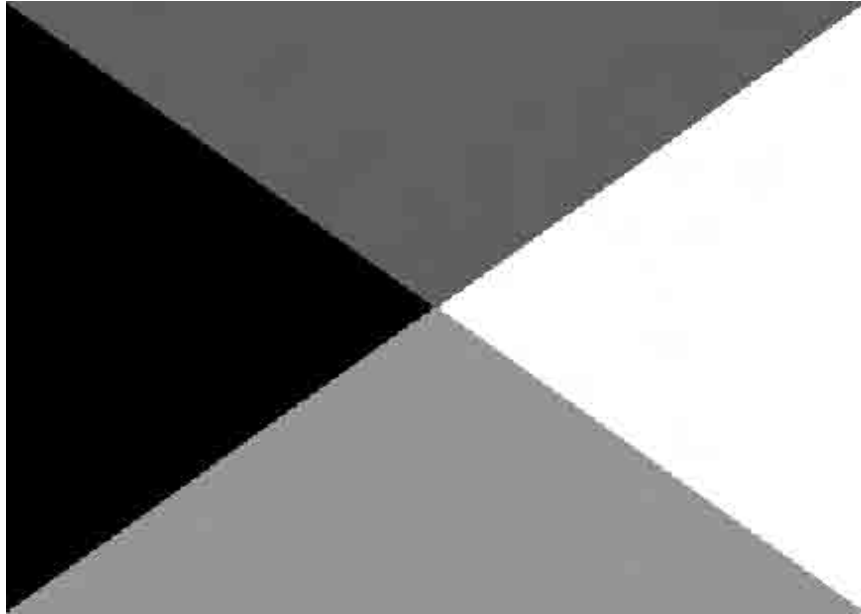


Figura 3.1: Imagem original.

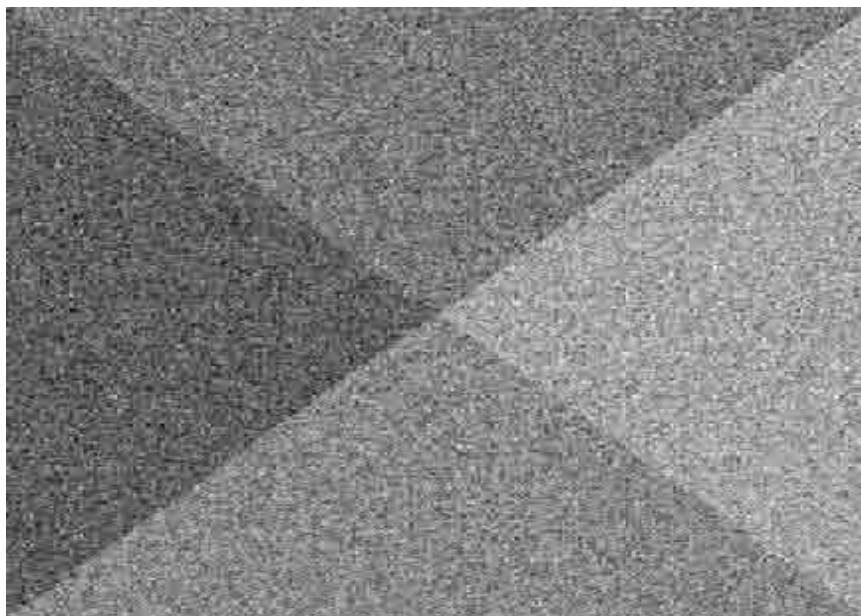


Figura 3.2: Imagem com ruído.



Figura 3.3: Reconstrução pelo método de Euler.



Figura 3.4: Reconstrução pelo método de Runge-Kutta de quarta ordem.



Figura 3.5: Reconstrução pelo método de Euler melhorado.



Figura 3.6: Reconstrução pelo método de Euler implícito.

3.1. IMPLEMENTAÇÃO NUMÉRICA

Agora, apresentamos os gráficos da linha 128 correspondente às imagens anteriores.

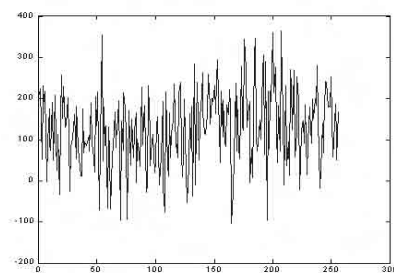
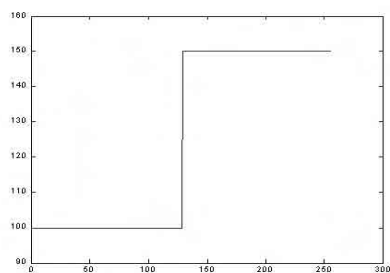


Gráfico da linha 128 das imagens original e com ruído.

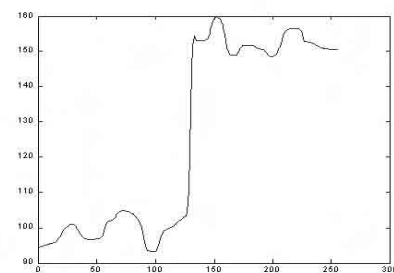
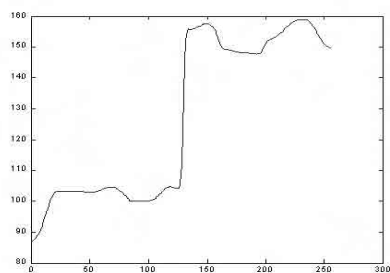


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

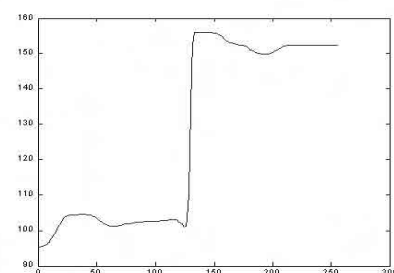
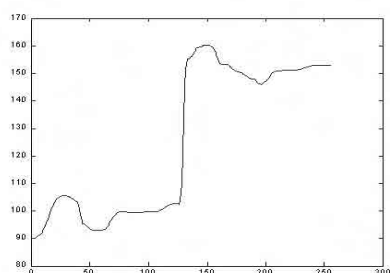


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

A segunda figura, possui $SNR = 0.0014$ e $\Delta t = 0.4$.

Foram realizadas 111 iterações, com os seguintes valores:
 $k = 46505.57385649140$, $\lambda = 1$, $\sigma = 63.33048888080454$ e $a = 89.63911267420528$.

E obtemos os seguintes erros médios:

Euler: $e_m = 4.096510883162409$;

Runge-Kutta de quarta ordem: $e_m = 4.509617490448035$;

Euler melhorado: $e_m = 4.084661623539079$;

Euler implícito: $e_m = 3.672594790448563$.

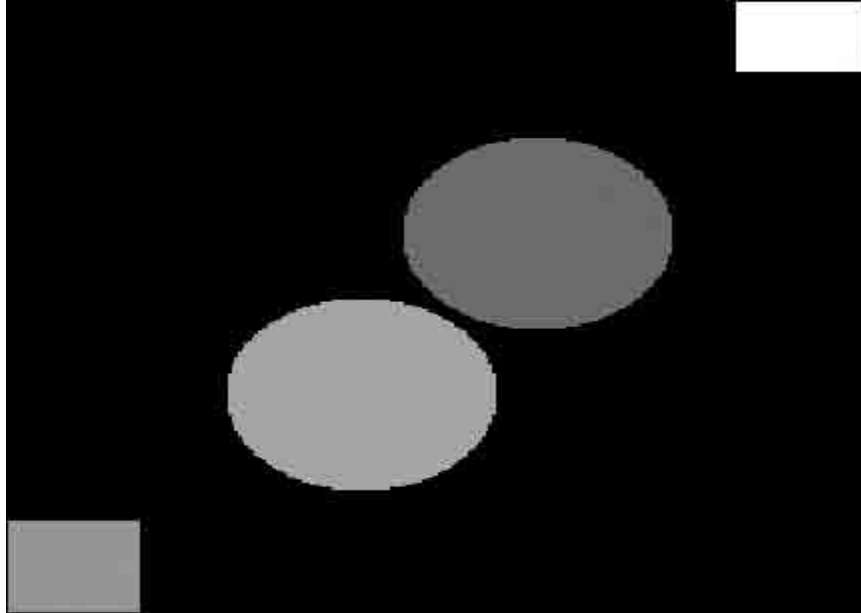


Figura 3.7: Imagem original.

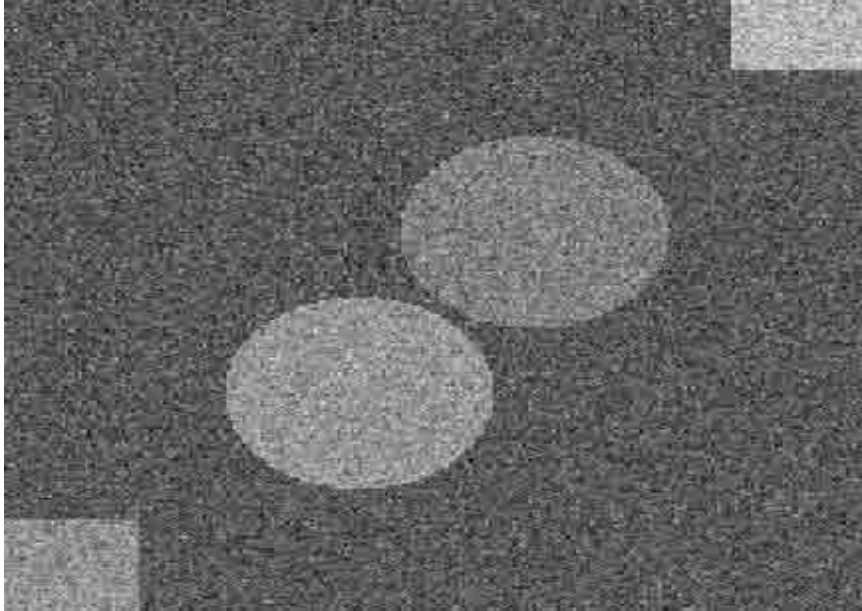


Figura 3.8: Imagem com ruído.

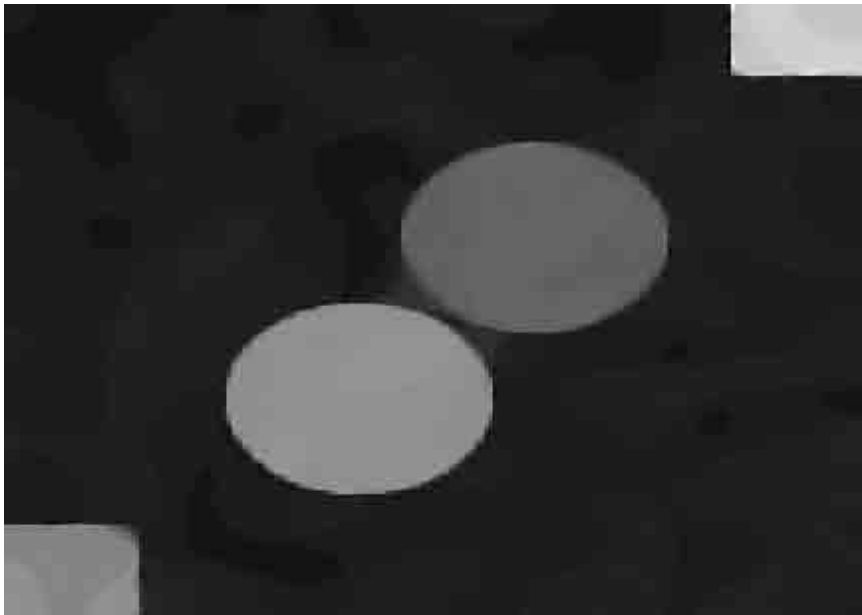


Figura 3.9: Reconstrução pelo método de Euler.

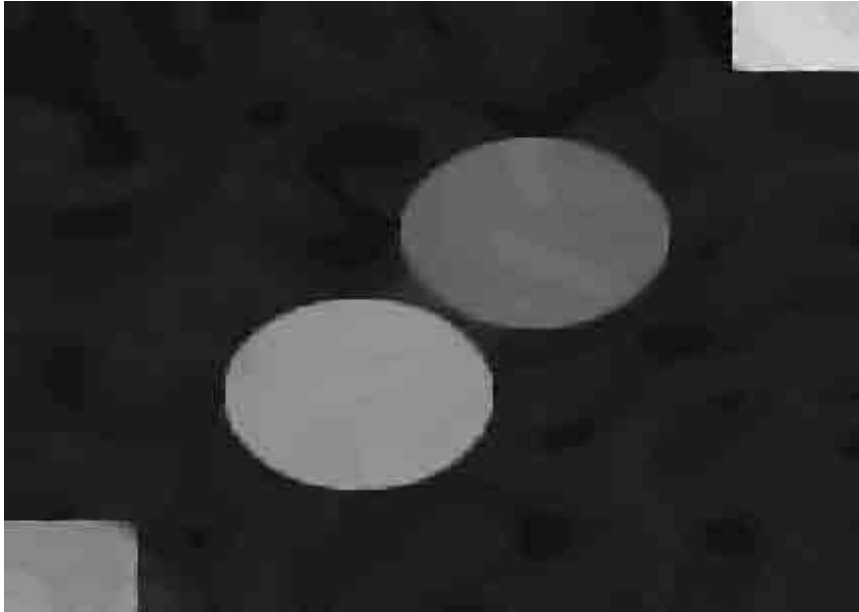


Figura 3.10: Reconstrução pelo método de Runge-Kutta de quarta ordem.

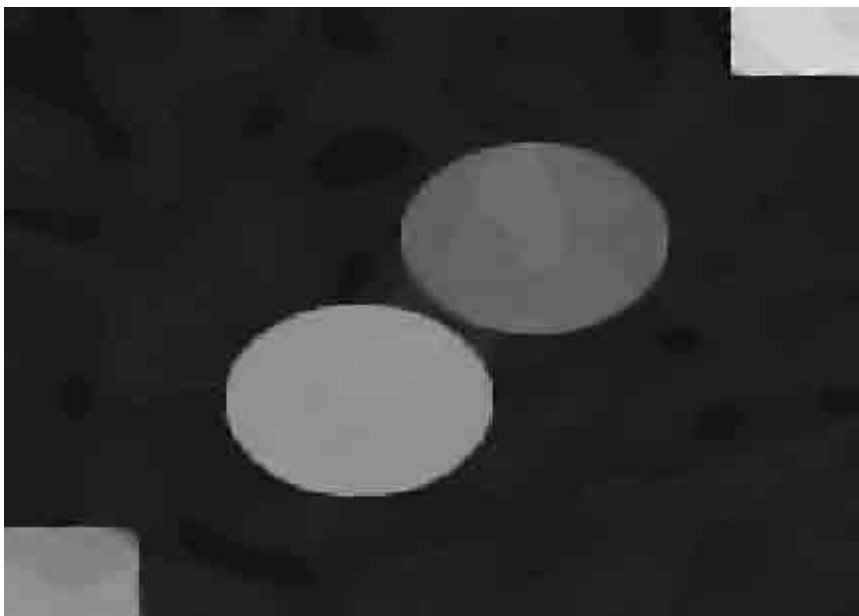


Figura 3.11: Reconstrução pelo método de Euler melhorado.

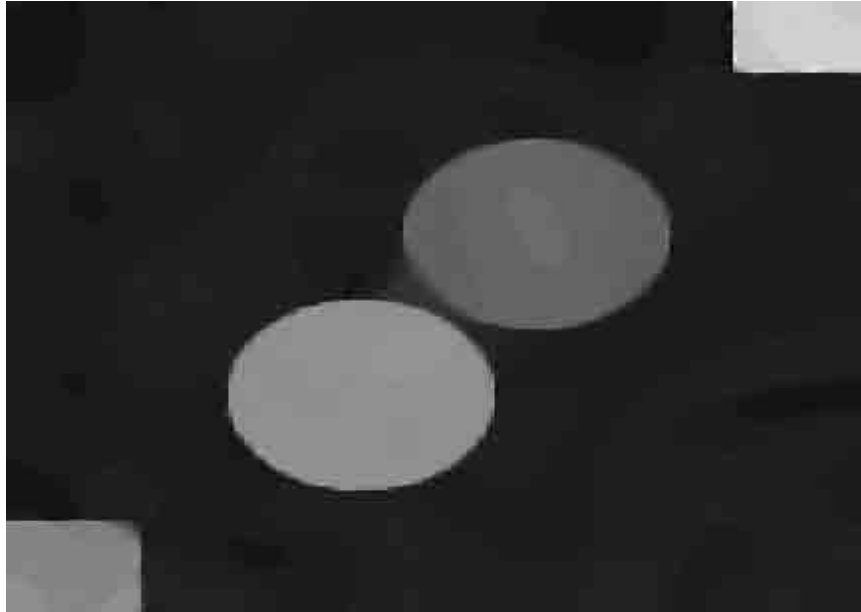


Figura 3.12: Reconstrução pelo método de Euler implícito.

Em seguida, mostramos os gráficos da linha 128 correspondente às imagens anteriores.

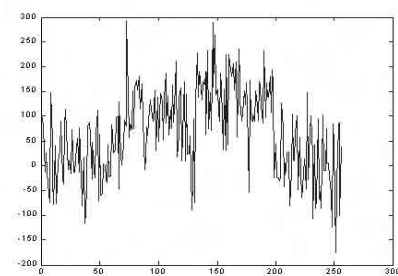
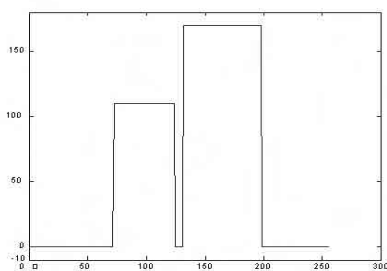


Gráfico da linha 128 das imagens original e com ruído.

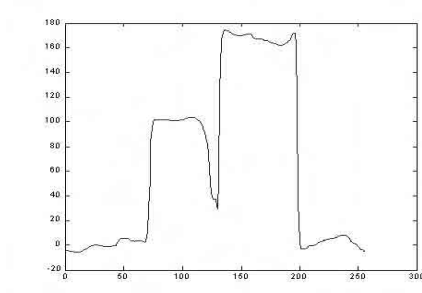
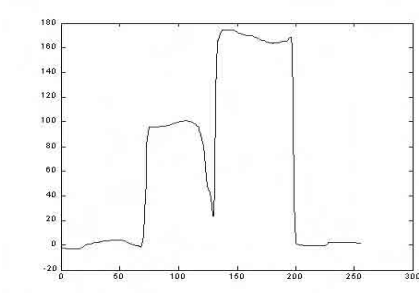


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

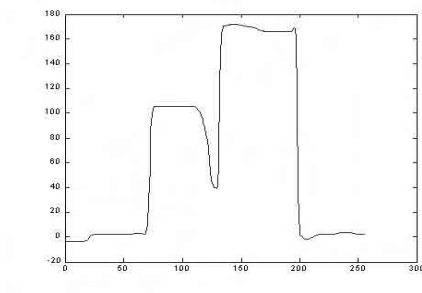
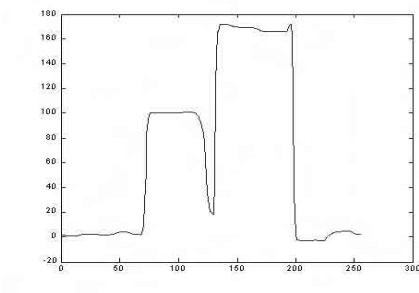


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

Agora, consideremos uma figura da vida real, com $SNR = 12.0426$ e, para esta figura, tomamos $\Delta t = 0.1$.

Foram realizadas, para esta imagem, 25 iterações, com os seguintes valores: $k = 7077.553720200804$, $\lambda = 1$, $\sigma = 10.37141783464018$ e $a = 42.69552226158050$.

Para cada método, obtemos os erros médios:

Euler: $e_m = 4.497775147547864$;

Runge-Kutta de quarta ordem: $e_m = 4.517931662401200$;

Euler melhorado: $e_m = 4.510952182176844$;

Euler implícito: $e_m = 4.429115875183594$.



Figura 3.13: Imagem original.



Figura 3.14: Imagem com ruído.



Figura 3.15: Reconstrução pelo método de Euler.



Figura 3.16: Reconstrução pelo método de Runge-Kutta de quarta ordem.



Figura 3.17: Reconstrução pelo método de Euler melhorado.



Figura 3.18: Reconstrução pelo método de Euler implícito.

E os respectivos gráficos:

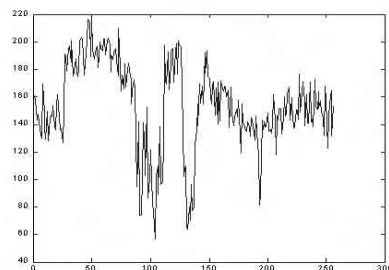
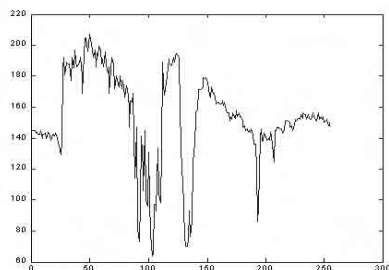


Gráfico da linha 128 das imagens original e com ruído.

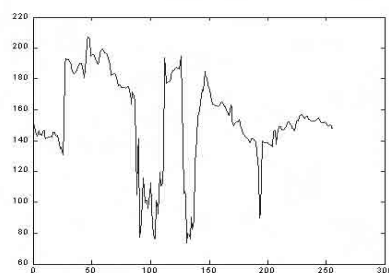
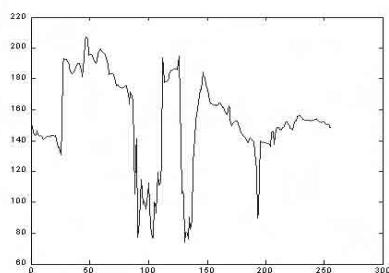


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

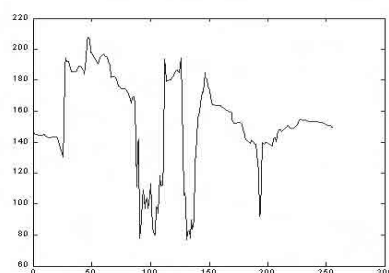
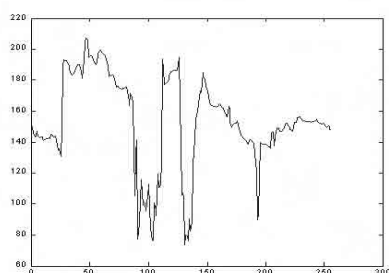


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

A figura a seguir é uma imagem com $SNR = -0.0383$ e $\Delta t = 0.4$.

Foram realizadas 48 iterações, com os seguintes valores:

$k = 5970.839419211368$, $\lambda = 1$, $\sigma = 27.29261430995206$ e $a = 38.45556689192124$.

E obtemos os seguintes erros médios:

Euler: $e_m = 1.679983885058301$;

Runge-Kutta de quarta ordem: $e_m = 2.143869706336979$;

Euler melhorado: $e_m = 1.748939021845636$;

Euler implícito: $e_m = 1.401804079263219$.

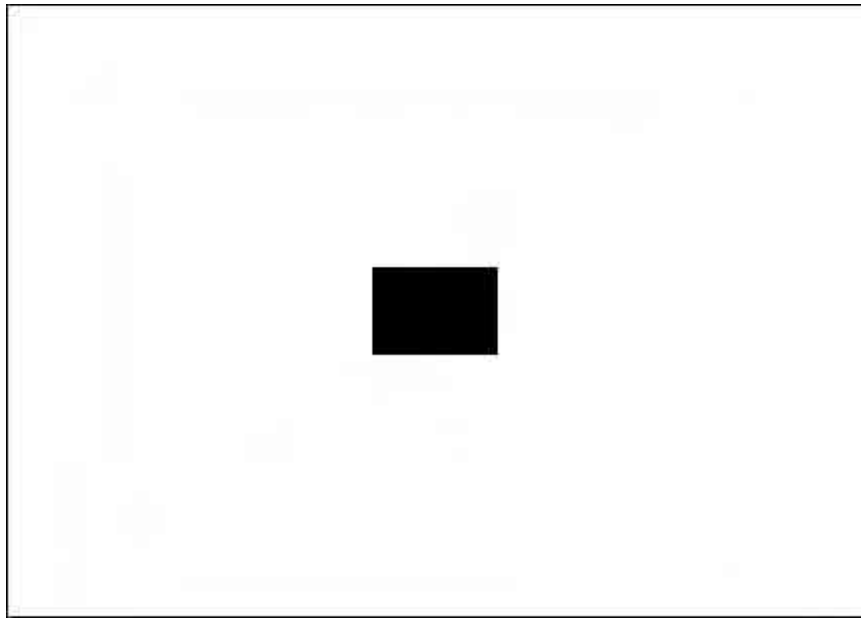


Figura 3.19: Imagem original.

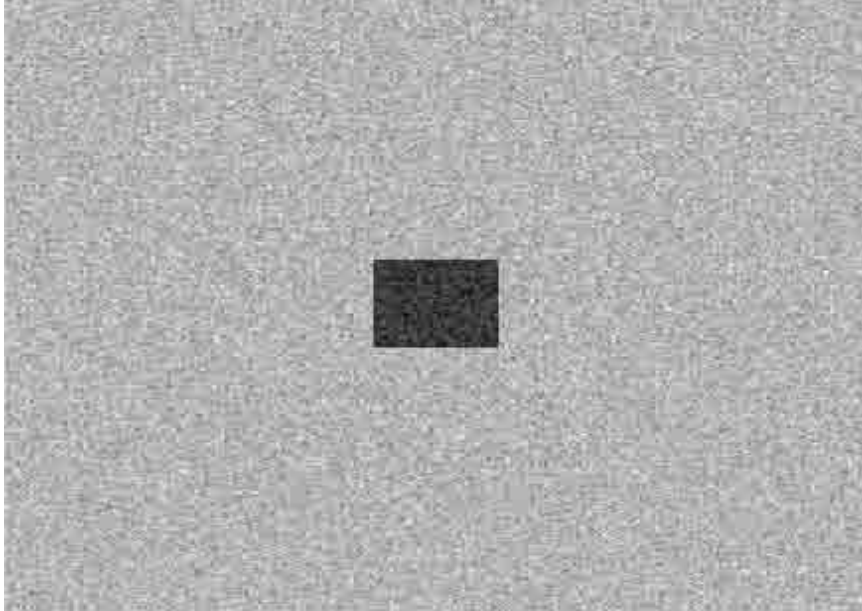


Figura 3.20: Imagem com ruído.

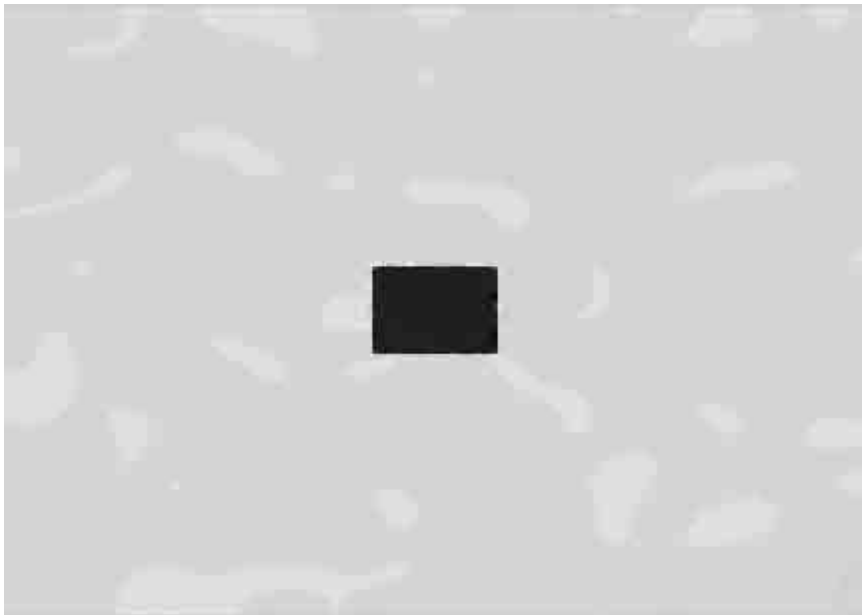


Figura 3.21: Reconstrução pelo método de Euler.

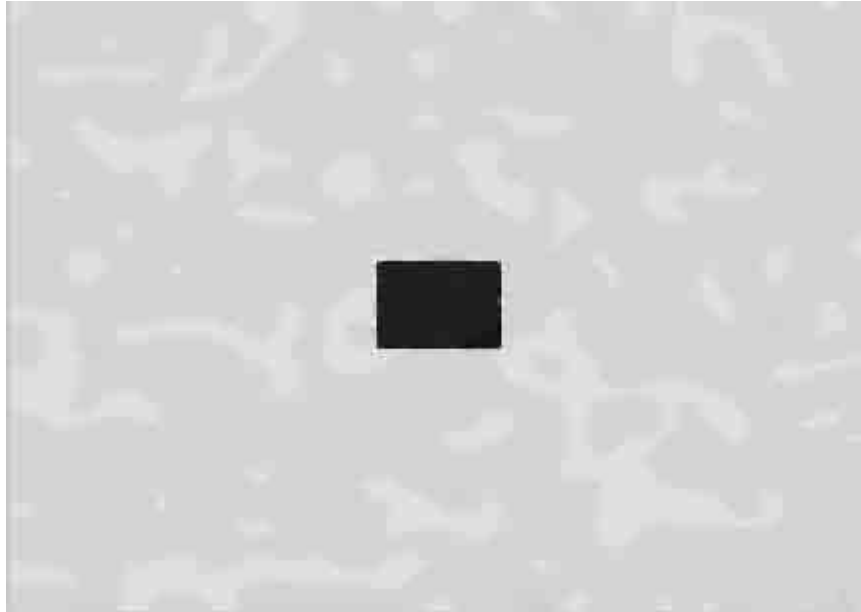


Figura 3.22: Reconstrução pelo método de Runge-Kutta de quarta ordem.

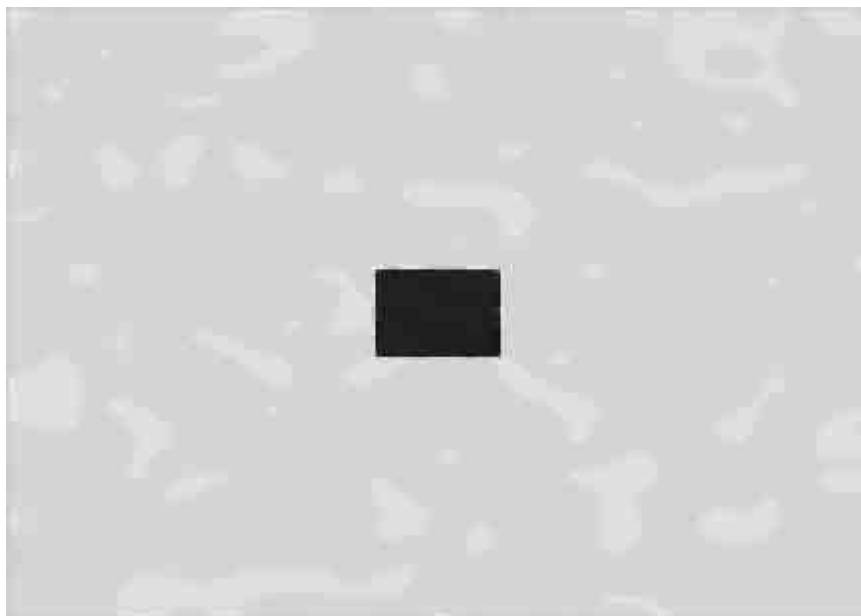


Figura 3.23: Reconstrução pelo método de Euler melhorado.



Figura 3.24: Reconstrução pelo método de Euler implícito.

Com respectivos gráficos:

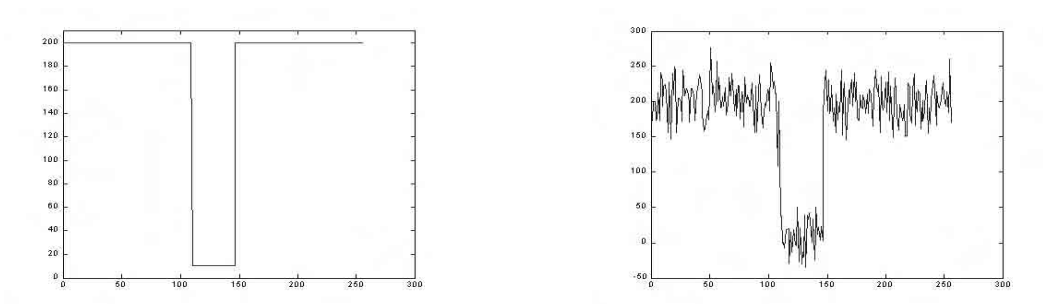


Gráfico da linha 128 das imagens original e com ruído.

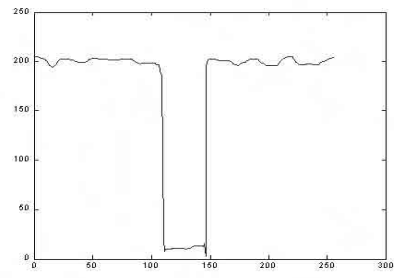
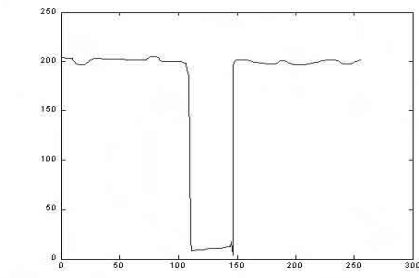


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

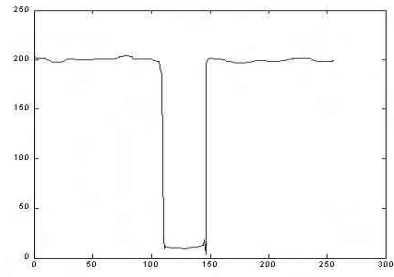
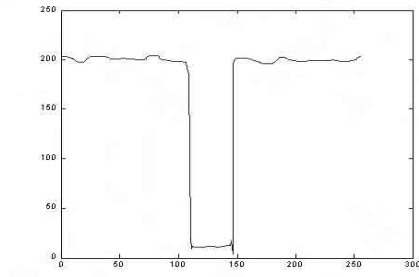


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

Agora, tomamos uma figura que possui $SNR = 6.0220$ e $\Delta t = 0.4$.

Neste caso, foram realizadas 39 iterações, com os seguintes valores: $k = 29937.97957656765$, $\lambda = 1$, $\sigma = 35.08812267991659$ e $a = 78.65666589492307$.

E os erros médios obtidos foram:

Euler: $e_m = 4.527174779043410$;

Runge-Kutta de quarta ordem: $e_m = 4.937737484449915$;

Euler melhorado: $e_m = 4.660082074507577$;

Euler implícito: $e_m = 4.215887545839231$.

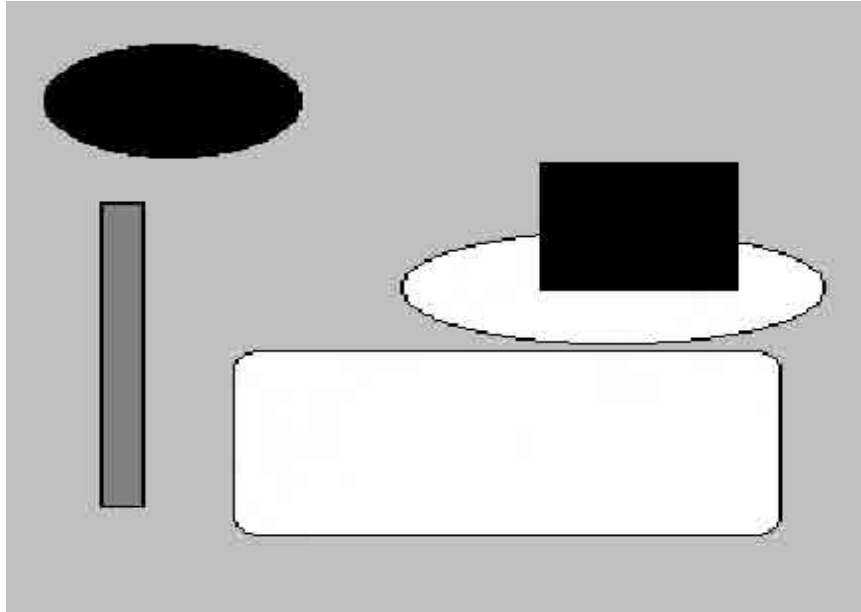


Figura 3.25: Imagem original.

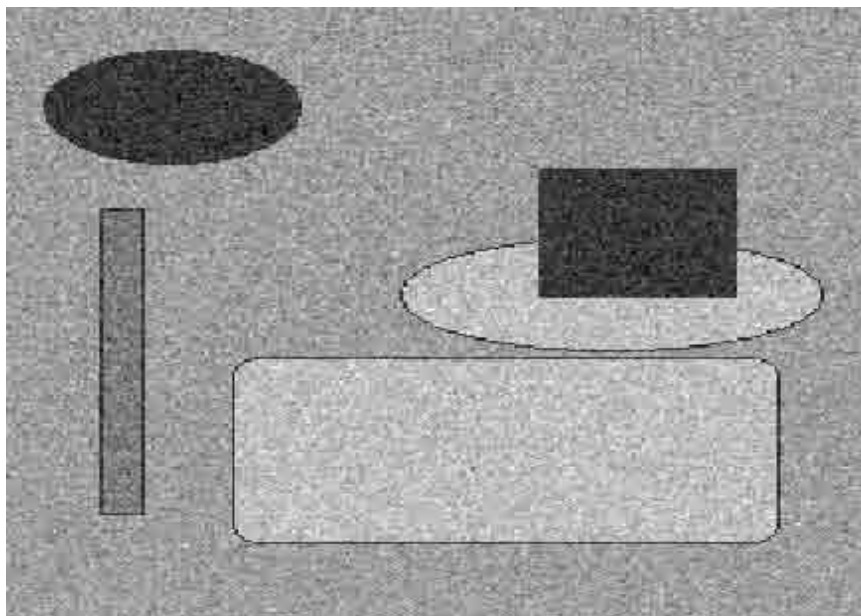


Figura 3.26: Imagem com ruído.

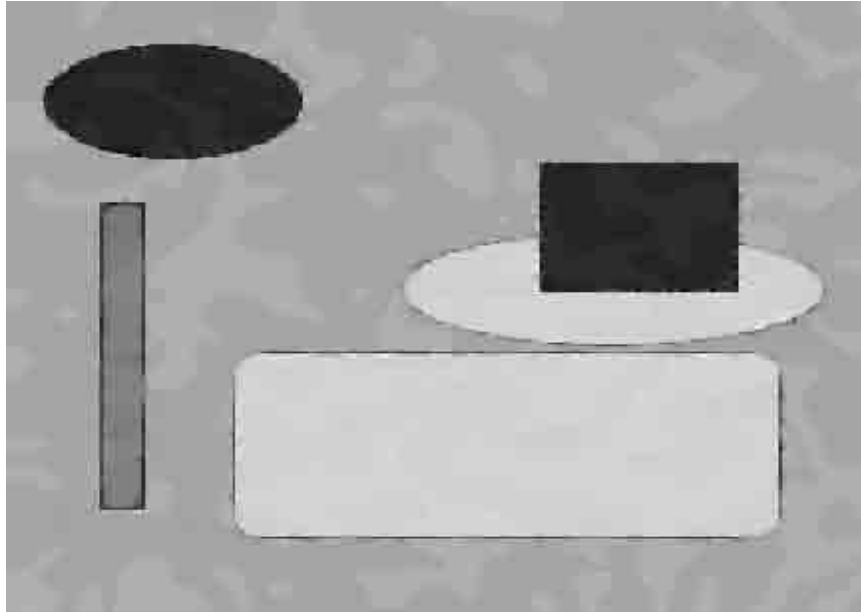


Figura 3.27: Reconstrução pelo método de Euler.

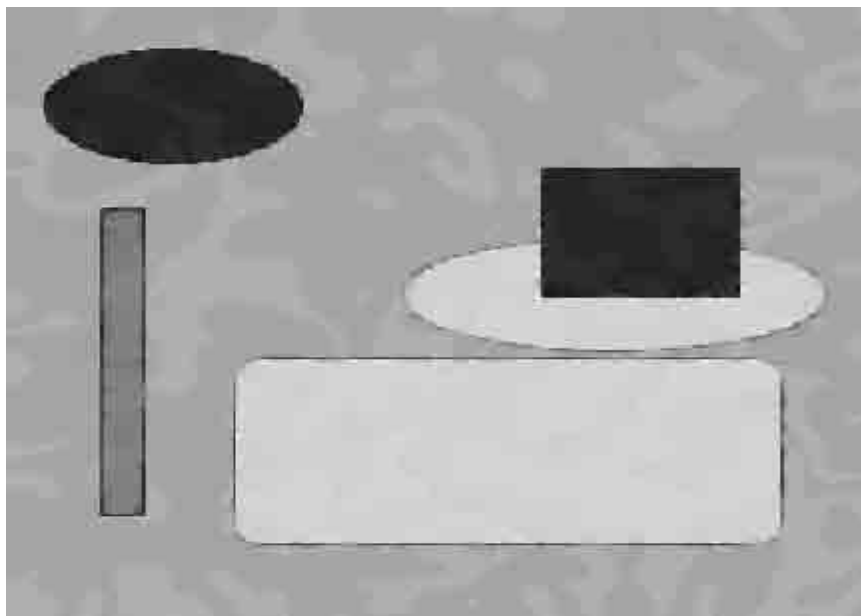


Figura 3.28: Reconstrução pelo método de Runge-Kutta de quarta ordem.

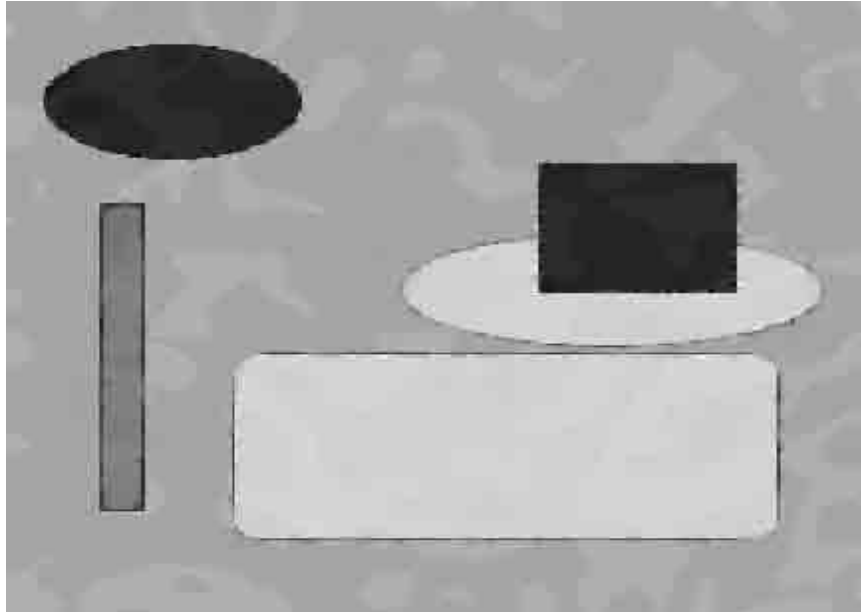


Figura 3.29: Reconstrução pelo método de Euler melhorado.

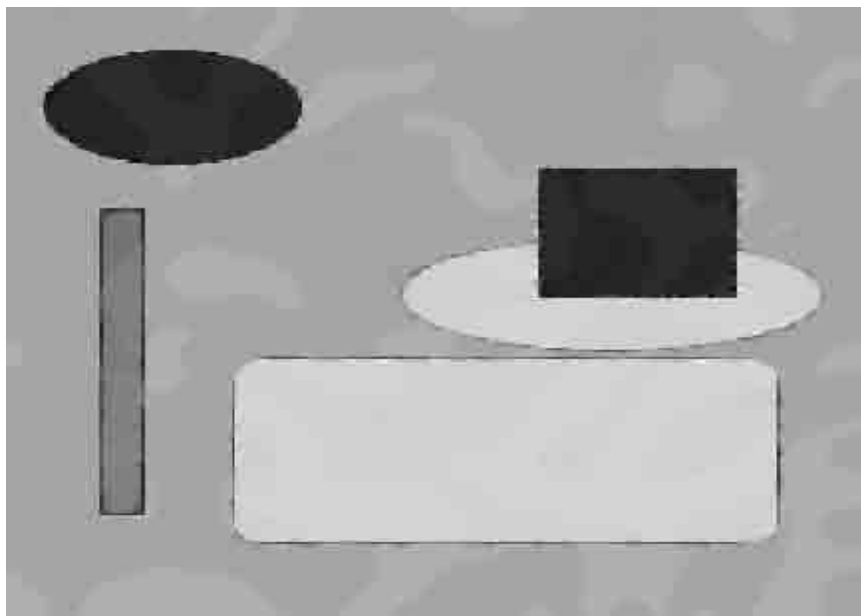


Figura 3.30: Reconstrução pelo método de Euler implícito.

3.1. IMPLEMENTAÇÃO NUMÉRICA

Cujos gráficos estão a seguir.

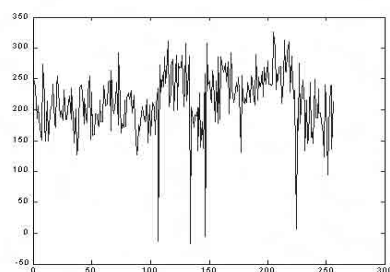
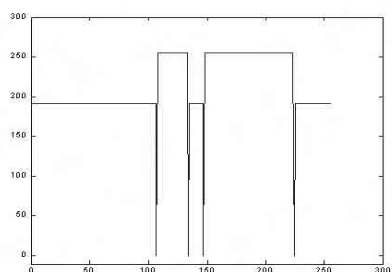


Gráfico da linha 128 das imagens original e com ruído.

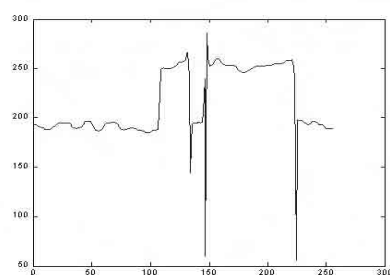
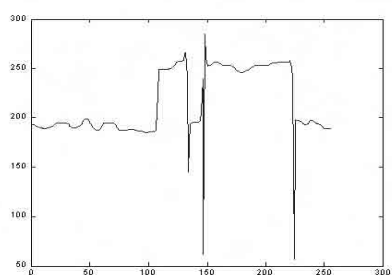


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

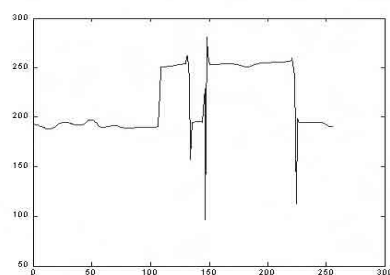
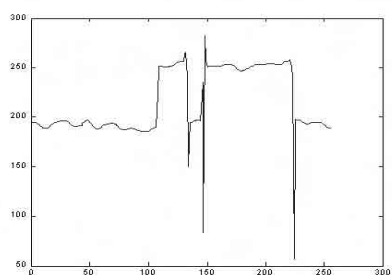


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

Por último, tomamos outra imagem da vida real, cujo valor de SNR é 12.0549 e $\Delta t = 0.1$.

Neste caso, foram realizadas 43 iterações, com os seguintes valores: $k = 24431.39683519813$, $\lambda = 1$, $\sigma = 17.81368831680179$ e $a = 73.58842133105432$.

E os erros médios obtidos foram:

Euler: $e_m = 9.433214689366467$;

Runge-Kutta de quarta ordem: $e_m = 9.445927109400376$;

Euler melhorado: $e_m = 9.435542847617956$;

Euler implícito: $e_m = 9.451112463962772$.



Figura 3.31: Imagem original.

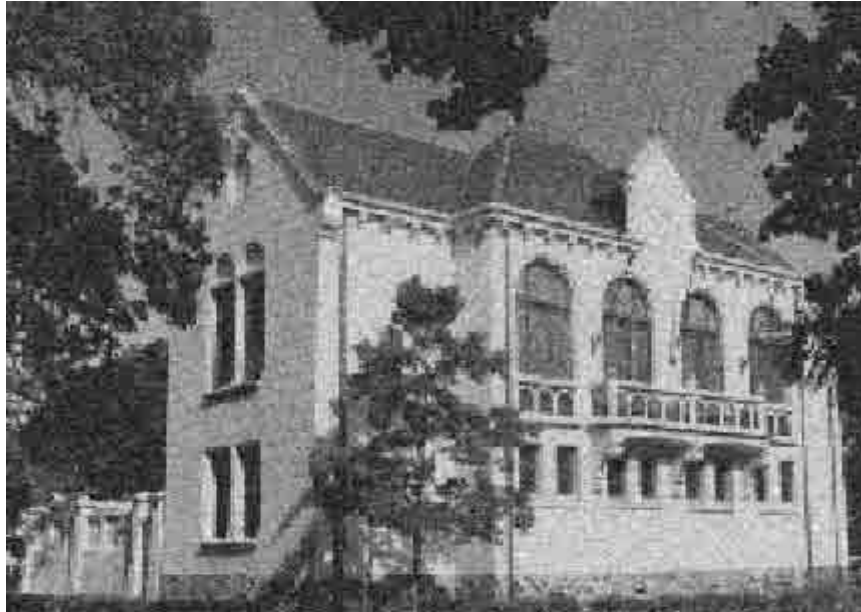


Figura 3.32: Imagem com ruído.



Figura 3.33: Reconstrução pelo método de Euler.



Figura 3.34: Reconstrução pelo método de Runge-Kutta de quarta ordem.



Figura 3.35: Reconstrução pelo método de Euler melhorado.



Figura 3.36: Reconstrução pelo método de Euler implícito.

E respectivos gráficos:

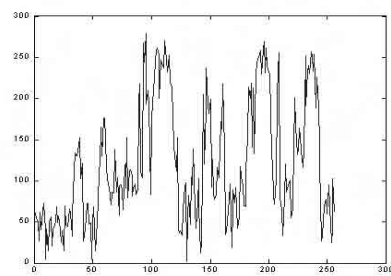
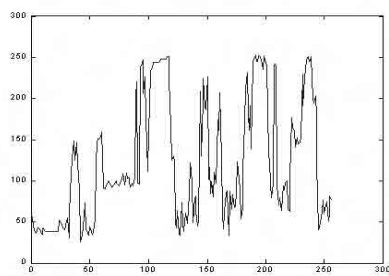


Gráfico da linha 128 das imagens original e com ruído.

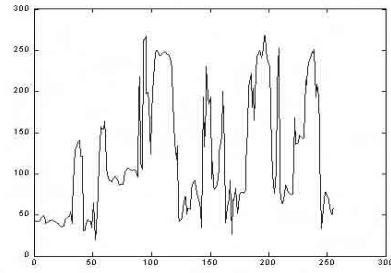
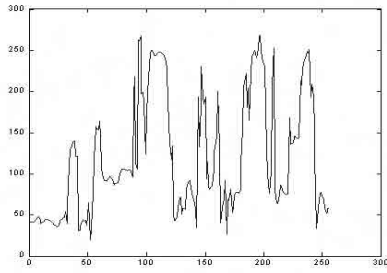


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler e de Runge-Kutta de quarta ordem, respectivamente.

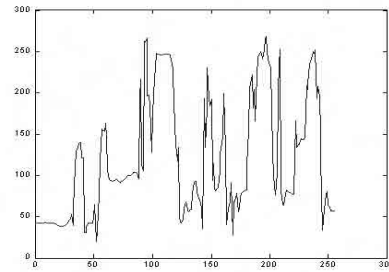
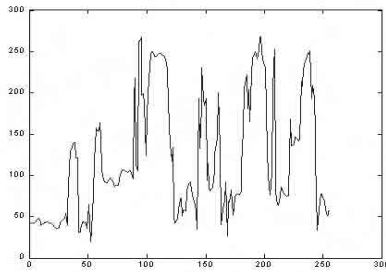


Gráfico da linha 128 das imagens reconstruídas pelos métodos de Euler melhorado e de Euler implícito, respectivamente.

3.2 Considerações Finais

Diante dos resultados apresentados na seção anterior, fazemos, aqui, algumas observações.

Comparando as imagens reconstruídas pela aplicação dos quatro métodos considerados, verificamos que, para as imagens sintéticas, os métodos de Euler e de Euler implícito preservam melhor as tonalidades enquanto que para os métodos de Euler melhorado e de Runge-Kutta de quarta ordem, as imagens apresentam uma maior variação nas tonalidades, o que pode ser observado pelo aparecimento de um número maior de manchas nas figuras. Este fato pode ser visualizado,

também, nos gráficos correspondentes à linha 128. Além disso, o menor erro médio calculado, corresponde ao método de Euler implícito.

Agora, para a Figura 3.13 e Figura 3.31, que correspondem às imagens da vida real, a comparação através da visualização gráfica é mais difícil. Mas, comparando os valores dos erros médios, verificamos que o obtido pelo método de Euler implícito é o menor, no caso da Figura 3.13. Entretanto, o mesmo não ocorre no caso da Figura 3.31.

A partir destes dados e de outros testes que realizamos para diferentes imagens, percebemos que, de um modo geral, o método de Euler implícito apresenta uma pequena melhora quando comparado aos outros métodos testados. Mas, não temos a pretensão de garantir, no momento, que ele forneça imagens melhores, pois dispomos apenas de resultados experimentais. Além disso, verificamos que o tempo de execução do programa que implementa este método é um pouco maior do que o tempo gasto, por exemplo, para a execução do método de Euler.

Este trabalho teve por objetivo verificar o comportamento de diferentes métodos numéricos aplicados à resolução do problema (2.3.4), que representa a eliminação de ruídos em imagens digitais, e propor um resultado sobre a estabilidade do esquema numérico (2.3.13).

Gostaríamos de mencionar a possibilidade de darmos prosseguimento a estes estudos, principalmente no que se refere à parte teórica, para que, juntamente com os resultados experimentais, possamos tirar conclusões mais precisas e, possivelmente, obter resultados melhores.

Referências Bibliográficas

- [1] ALVAREZ, L.; LIONS, P. L.; MOREL, J. M. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, v. 29, n. 3, p. 845-866, jun. 1992.
- [2] ALVAREZ, L.; ESCLARIN, J. Image quantization using reaction-diffusion equations. *SIAM J. APPL. MATH.*, p. 153-175, feb. 1994.
- [3] ALVAREZ, L.; MOREL, J. M. Formalization and computational aspects of image analysis. *Acta Numerica*, p. 1-59, 1994.
- [4] AMES, W. F. *Numerical Methods for Partial Differential Equations*. 3. ed. Academic Press Limited, 1992.
- [5] ATKINSON, K. E. *An Introduction to Numerical Analysis*. John Wiley and Sons, 1988.
- [6] BARCELOS, C. A. Z. Restauração e Análise de Imagens via Equações Diferenciais Parciais. *Tendências em Matemática Aplicada e Computacional*, SBMAC, v.3, n.2, p. 1-13, 2002.
- [7] BARCELOS, C. A. Z.; BOAVENTURA, M.; SILVA JÚNIOR, E. C. A well-balanced flow equation for noise removal and edge detection. *IEEE Transactions on image processing*, v.12, n.7, p. 751-763, jul. 2003.
- [8] BIRKHOFF, G.; ROTA, G. *Ordinary Differential Equations*. 3. ed. John Wiley and Sons, 1978.

- [9] BURDEN R. L.; FAIRES J. D. *Numerical Analysis*. PWS-KENT Publishing Company, 1993.
- [10] BUTCHER, J. C. *The Numerical Analysis of Ordinary Differential Equations*. John Wiley and Sons Ltd, 1987.
- [11] CAMPOS FILHO, F. F. *Algoritmos Numéricos*. Rio de Janeiro: LTC, 2001.
- [12] CIDADE, G.; SILVA NETO, A.; ROBERTY, N. C. Restauração de imagens com aplicações em biologia e engenharia. *Notas em Matemática Aplicada*, SBMAC, v.1, 2003.
- [13] CUNHA, M. C. C. *Métodos Numéricos*. 2. ed. Campinas, SP: Editora da UNICAMP, 2003.
- [14] GREENSPAN, D.; CASULLI, V. *Numerical Analysis for Applied Mathematics, Science, and Engineering*. Addison-Wesley Publishing Company, 1988.
- [15] LAMBERT, J. D. *Computational Methods in Ordinary Differential Equations*. John Wiley and Sons, 1973.
- [16] LAMBERT, J. D. *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*. John Wiley and Sons Ltd, 1991.
- [17] NORDSTRÖM, K. N. Biased anisotropic diffusion: a unified regularization and diffusion approach to edge detection. *Image and Vision Computing*, p. 318-327, 1990.
- [18] NOVAIS, A.; CUNHA, M. C. C. Métodos numéricos para equações diferenciais parciais. *Notas em Matemática Aplicada*, SBMAC, v.4, 2003.
- [19] ORTEGA, J. M. *Numerical Analysis: A Second Course*. Academic Press, 1972.

- [20] PERONA P.; MALIK J. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.12, n.7, p. 629-639, jul. 1990.
- [21] RUGGIERO, M. A. G.; LOPES, V. L. R. *Cálculo Numérico: Aspectos Teóricos e Computacionais*. 2. ed. Pearson Education do Brasil, 1996.
- [22] SILVA JÚNIOR, E. C. *Um novo modelo de difusão não linear com preservação de bordas para segmentação e eliminação de ruídos em imagens e análise comparativa com modelos correlatos*. Dissertação (Mestrado em Matemática Aplicada) - IBILCE, UNESP, São José do Rio Preto - SP, 2002.
- [23] WENDLANDER, E. Modelos matemáticos e métodos numéricos em águas subterrâneas. *Notas em Matemática Aplicada*. SBMAC, v.3, 2003.