



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

**INPE-7992-TDI/746**

**REGISTRO DE IMAGENS E MÉTODOS DE CÁLCULO DO ERRO  
DE REGISTRO**

Sonia Mara de Souza

Dissertação de Mestrado em Computação Aplicada, orientada pelo Dr. Maurício Fabbri,  
aprovada em 22 de abril de 1999.

INPE  
São José dos Campos  
2000

**Publicado por:**

**Coordenação de Ensino, Documentação e  
Programas Especiais - CEP**

**Instituto Nacional de Pesquisas Espaciais - INPE**

**Caixa Postal 515**

**12201-970 - São José dos Campos - SP - Brasil**

**Fone: (012) 345.6911**

**Fax: (012) 345.6919**

**E-Mail: [marciana@sid.inpe.br](mailto:marciana@sid.inpe.br)**

- Solicita-se intercâmbio
- Exchange welcome
- Si sollecita intercambio
- Echange souhaité
- Mann bittet un Austausch
- Solicitamos intercambio
- Проси обмен
- 歡迎交換
- 出校防交換

**Publicação Externa - É permitida sua reprodução para interessados.**

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

**INPE-7992-TDI/746**

**REGISTRO DE IMAGENS E MÉTODOS DE CÁLCULO DO ERRO  
DE REGISTRO**

Sonia Mara de Souza

Dissertação de Mestrado em Computação Aplicada, orientada pelo Dr. Maurício Fabbri,  
aprovada em 22 de abril de 1999.

INPE  
São José dos Campos  
2000

621.376.5

SOUZA, S. M.

Registro de imagens e métodos de cálculo do erro de registro / S. M. Souza - São José dos Campos: INPE , 1999.  
132p. - (INPE-7992-TDI/746).

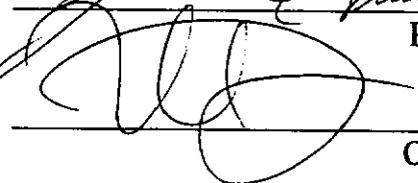
1. Correção geométrica. 2. Detecção de borda. 3. Transformações. 4. Precisão geométrica. 5. Imagem de satélite. I. Título.

Aprovado pela Banca Examinadora em  
cumprimento a requisito exigido para a  
obtenção do Título de Mestre em  
**Computação Aplicada.**

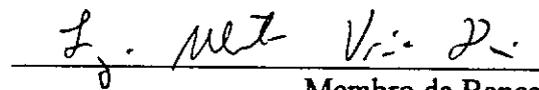
Dr. Jerônimo dos Santos Travelho

  
\_\_\_\_\_  
Presidente

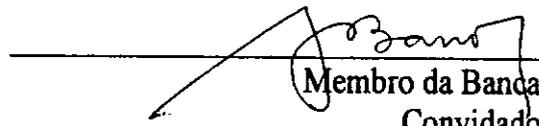
Dr. Maurício Fabbri

  
\_\_\_\_\_  
Orientador

Dr. Luiz Alberto Vieira Dias

  
\_\_\_\_\_  
Membro da Banca

Dr. Ardemiro de Barros Silva

  
\_\_\_\_\_  
Membro da Banca  
Convidado

Candidato: Sonia Mára de Souza

São José dos Campos, 16 de abril de 1999.

Um agradecimento muito especial ao meu Orientador Espiritual que muito ajudou na elaboração deste trabalho. "Nunca te deixarei, jamais te abandonarei" (Hb 13,5).

## **AGRADECIMENTOS**

Agradeço ao pessoal do DPI, principalmente João Diehl e Guaraci Erthal pela colaboração prestada no desenvolvimento deste trabalho. Agradeço de maneira especial Leila Maria Garcia Fonseca pelas valiosas discussões, sem as quais este trabalho não teria sido concluído.

Agradeço ao meu orientador Maurício Fabbri pelo incentivo e colaboração neste trabalho.

Agradeço ao Amauri Montes pelas dúvidas esclarecidas.

Agradeço aos amigos Anibal e Rovedy pela ajuda prestada na parte computacional.

Agradeço a minha amiga Patrícia Barreto que ajudou na formatação deste trabalho.

Agradeço as bibliotecárias Silvia, Suelena e Yolanda pela disponibilidade de sempre servir.

Agradeço aos meus amigos Carlos e Alice Ueda pelo apoio e encorajamento de seguir em frente.

Finalmente agradeço ao meu marido Antonio Marcio e meus filhos Leticia e Luiz Otávio pela compreensão e paciência.

## RESUMO

Neste trabalho foram avaliados e testados dois métodos de registros de imagens do sensor TM (*Landsat*), um manual (tradicional) e o outro automático (algoritmo de Li et al). No método manual a seleção dos pontos de controle foi realizada manualmente através da tela do computador. No método automático esta escolha é feita pelo algoritmo através da semelhança das feições adquiridas das imagens. Para analisar a precisão das imagens registradas foram calculados o Erro de Registro através de dois métodos. Um deles, de uso tradicional, é a raiz quadrada do erro médio quadrático (RMSE), que é a medida dos erros residuais entre alguns pontos de teste escolhidos na imagem. O outro é o Método da Variância, que não é de uso comum, e trabalha com dois tipos de erros: erro geométrico e o erro de medida. Ele calcula o erro total e subtrai o erro de medida gerado pelo processo que está sendo usado para calcular o erro.

# **REGISTRATION OF IMAGES AND METHODS OF MEASUREMENT OF REGISTRATION ACCURACY**

## **ABSTRACT**

In this work were appraised and tested two methods of registrations of images of sensor TM (*Landsat*), a manual (traditional) and the other automatic (algorithm of Li et al). In the manual method the selection of the control points was accomplished manually through the screen of the computer. In the automatic method this choice is made by the algorithm through the likeness of the acquired features of the images. To analyze the precision of the registered images they were calculated the accuracy of registration through two methods. One of them, of traditional use, is the root mean square error (RMSE), that is the measure of registration accuracy among some test points chosen in the image. The other is the Method of the Variance, that is not of common use, and it works with two types of errors: geometric error and the measurement error. It calculates the total error and it subtracts the measurement error generated by the process that is being used to measure the accuracy.

# SUMÁRIO

<b>LISTA DE FIGURAS</b>	<b>15</b>
<b>LISTA DE TABELAS</b>	<b>19</b>
<b>Capítulo 1- INTRODUÇÃO</b> .....	<b>21</b>
1.1- Proposta de Trabalho.....	23
<b>Capítulo 2- CONCEITOS BÁSICOS</b> .....	<b>27</b>
2.1- Imagens em Sensoriamento.....	27
2.2- Imagem Digital.....	27
2.2.1- Caracterização de Imagens .....	29
2.2.1.1- Resolução e Bandas .....	30
2.3- Detecção de Borda .....	31
2.3.1- Borda .....	33
<b>Capítulo 3- CONCEITOS MATEMÁTICOS</b> .....	<b>37</b>
3.1- Transformação Linear .....	37

3.2-	Transformação Afim .....	40
3.3-	Transformações Espaciais .....	45
<b>Capítulo 4-</b>	<b>MÉTODOS DE REGISTRO DE IMAGENS .....</b>	<b>49</b>
4.1-	Registro de Imagens .....	49
4.1.1-	REGISTRO MANUAL.....	52
4.1.2-	REGISTRO AUTOMÁTICO .....	54
4.1.2.1-	Registro Baseado Em Área .....	54
4.1.2.2-	Registro Baseado Em Feições .....	57
4.2-	Cálculo do Erro de Registro .....	59
4.2.1-	Primeiro Método: RMSE .....	59
4.2.2-	Segundo Método: Variância .....	61
<b>Capítulo 5-</b>	<b>REGISTRO AUTOMÁTICO: O ALGORITMO DE LI ET AL [4].....</b>	<b>65</b>
5.1-	Descrição do Algoritmo .....	65
5.1.1-	Realce de Bordas .....	66
5.1.2-	Detecção de Bordas .....	69

5.1.3-	Codificação e Extração de Atributos .....	71
5.1.4-	Casamento .....	75
5.1.5-	Estimação dos Parâmetros da Função de Transformação .....	78
5.1.6-	Teste de Consistência .....	81
<b>Capítulo 6-</b>	<b>RESULTADOS OBTIDOS .....</b>	<b>83</b>
6.1-	Registro usando o Método Manual .....	83
6.1.1-	Primeiro Registro: Caraguatatuba .....	84
6.1.2-	Segundo Registro: Pindamonhangaba .....	85
6.1.3-	Terceiro Registro: Natividade .....	87
6.1.4-	Quarto Registro: Mato Grosso .....	87
6.2-	Registro usando o Método de Feições .....	94
6.2.1-	Primeiro Registro: Caraguatatuba .....	94
6.2.2-	Segundo Registro: Pindamonhangaba .....	95
6.2.3-	Terceiro Registro: Natividade .....	95
6.2.4-	Quarto Registro: Mato Grosso .....	99

6.2.5- Demais Ilustrações .....	100
<b>Capítulo 7- CONCLUSÕES E COMENTÁRIOS.....</b>	<b>111</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>117</b>
<b>Apêndice A-PROGRAMAS EM C .....</b>	<b>121</b>
A.1- Cálculo da Correlação .....	121
A.2- Cálculo da Variância .....	129

## LISTA DE FIGURAS

2.1	Exemplo de detecção de borda. ....	32
2.2	Seção transversal de uma árvore. ....	33
2.3	Típos de bordas. ....	34
2.4	Modelos de bordas. ....	35
3.1	Transformação linear. ....	38
3.2	Transformação afim. ....	41
3.3	Uma transformação afim pode inverter figuras. ....	42
3.4	Transformação afim. ....	43
3.5	Duas folhas determinam uma transformação afim. ....	43
3.6	Transformação espacial. ....	45
3.7	Superfície $u(x_i, y_i)$ . ....	47
4.1	Efeitos de distorções geométricas. ....	50
4.2	Registro manual. ....	52
4.3	Registro baseado em área. ....	55
4.4	Medida da precisão de registro. ....	64
5.1	Diagrama de blocos. ....	65
5.2	Exemplo de detecção de borda. ....	67
5.3	LoG em uma dimensão. ....	68

5.4	LoG em duas dimensões. ....	68
5.5	Exemplo de detecção de borda. ....	69
5.6	Imagem (a), convolução com o LoG (b) e os cruzamentos (c). ....	70
5.7	Janela 3 x 3. ....	70
5.8	Representação de um contorno em código de cadeia. ....	72
5.9	Exemplo de um contorno. ....	73
5.10	Exemplo de cálculo de perímetro. ....	74
5.11	Exemplo da função mod em um contorno. ....	77
5.12	Mapeamento de um ponto (X,Y) em um ponto ( $\hat{X}, \hat{Y}$ ). ....	80
6.1	Caraguá1. ....	84
6.2	Caraguá2. ....	85
6.3	Imagem registrada - caragua - banda3. ....	86
6.4	Pinda1. ....	87
6.5	Pinda2. ....	88
6.6	Imagem registrada - Pinda - banda 3 ....	88
6.7	Natividade1. ....	89
6.8	Natividade2. ....	89
6.9	Imagem registrada - Natividade - banda 3 ....	90
6.10	Matol. ....	90
6.11	Matol2. ....	91

<b>6.12</b>	Imagem registrada - Mato - banda 3 .....	91
<b>6.13</b>	Caraguá - feições encontradas - banda3.....	96
<b>6.14</b>	Caraguá - feições encontradas - banda5.....	96
<b>6.15</b>	Caraguá - feições encontradas - banda7.....	97
<b>6.16</b>	Mosaico - Caraguá - banda3.....	97
<b>6.17</b>	Mosaico - Caraguá - banda5.....	98
<b>6.18</b>	Mosaico - Caraguá - banda7.....	98
<b>6.19</b>	Pinda - feições encontradas - banda3.....	99
<b>6.20</b>	Mosaico - Pinda - banda3.....	100
<b>6.21</b>	Natividade - feições encontradas - banda3.....	101
<b>6.22</b>	Mosaico - Natividade - banda3.....	101
<b>6.23</b>	Mato - feições encontradas - banda7.....	102
<b>6.24</b>	Mosaico - Mato - banda7.....	103
<b>6.25</b>	Caraguá - Ponto4.....	104
<b>6.26</b>	Caraguá - Ponto5.....	104
<b>6.27</b>	Caraguá - Ponto7.....	105
<b>6.28</b>	Caraguá - Ponto9.....	105
<b>6.29</b>	Contornos encontrados de Caraguá1 - banda3.....	106
<b>6.30</b>	Contornos encontrados de Caraguá2 - banda3.....	106

## LISTA DE TABELAS

2.1	Resolução dos sistemas sensores .....	31
6.2	Pontos de controle e pontos de teste.....	86
6.3	Coordenadas dos pontos de teste.....	92
6.4	Parâmetros da transformação - Método manual .....	93
6.5	Caraguá - pontos aproximados - Método manual .....	93
6.6	Pinda - pontos aproximados - Método manual .....	93
6.7	Natividade - pontos aproximados - Método manual .....	93
6.8	Mato - pontos aproximados - Método manual .....	93
6.9	Número de contornos casados e valores limiares .....	94
6.10	Caraguá - pontos aproximados - Método automático .....	107
6.11	Pinda - pontos aproximados - Método automático .....	107
6.12	Natividade - pontos aproximados - Método automático .....	107
6.13	Mato - pontos aproximados - Método automático .....	107
6.14	Parâmetros da transformação - Método automático .....	107
6.15	Comparação entre o Método manual e o automático .....	108
6.16	Valores obtidos para o Método da variância .....	108
6.17	Comparação entre o Método RMSE e a variância .....	109
7.18	Comparação entre os Métodos usando transf. afim .....	115

# CAPÍTULO 1

## INTRODUÇÃO

Registro de duas imagens, obtidas de uma mesma cena, pode ser entendido como a operação que faz com que uma imagem se case, ou se sobreponha perfeitamente a outra, de modo que cada par de pontos obtidos das imagens assim sobrepostas corresponda ao mesmo ponto do objeto físico na cena imageada [1]. O processo de registro de imagens é fundamental em vários problemas práticos nas áreas de sensoriamento remoto, visão por computador, reconhecimento de padrões e análise de imagens médicas. Algumas aplicações são descritas a seguir [2].

Em sensoriamento remoto, há um grande número de sensores de recursos naturais com diferentes características geométricas e radiométricas. A combinação de imagens adquiridas por estes sensores pode efetivamente melhorar o processo de extração de informação, sendo por isto uma técnica bastante utilizada. Por exemplo, a combinação de imagens dos satélites SPOT e Landsat cujos sensores são *Haute Resolution Visible* (HRV) e *Thematic Mapper* (TM) respectivamente, tem sido muito usada no monitoramento do crescimento urbano.

Um outro exemplo é a combinação de imagens óticas com imagens de radares. As imagens de radar são pouco afetadas por coberturas de nuvens, e além disso contem informações que podem ser complementares na análise da região em estudo. Por exemplo, dados de radares de abertura sintética (*Synthetic Aperture Radar* -SAR) obtidos a partir dos satélites *Shuttle Imaging Radar-C* (SIR-C) e *Japanese Earth Resources Satellite-1* (JERS-1) combinados com imagens TM têm sido usados no mapeamento de inundações e vegetação na área de Manaus. Sensores

SAR operam independentemente da iluminação solar e cobertura de nuvens, enquanto que as imagens TM fornecem informações adicionais na porção do espectro óptico. Como estes sensores estão em diferentes plataformas e em diferentes órbitas, o registro das imagens é um processo obrigatório antes da combinação e análise dos dados.

No campo da análise de imagens médicas, é procedimento comum a integração de informações obtidas por diferentes equipamentos tais como a tomografia computadorizada (CT), tomografia por emissão de pósitron (PET) e imageamento por ressonância magnética (MRI) para detecção e localização de doenças. Cada um destes equipamentos gera imagens com diferentes características geométricas, as quais também devem ser registradas antes de serem analisadas.

Na análise multi-temporal de imagens, o objetivo é detectar mudanças que tenham ocorrido em um determinado intervalo de tempo. Um método simples de detectar estas modificações em um par de imagens é sobrepô-las e identificar as diferenças entre elas. Como estas imagens foram tomadas sob condições e épocas diferentes, elas devem ser registradas antes do processo comparativo. Uma aplicação típica da análise temporal de imagens é a detecção e monitoramento de mudanças e crescimento urbano, e monitoramento de desmatamento.

Em visão por computador, o registro é necessário quando se quer determinar o movimento de um objeto, e também obter informação tridimensional de uma cena a partir de um par estereoscópico de imagens [3]. Nesta categoria se incluem, também, o problema de mosaico de imagens e medição da direção do vento. Mosaico de imagens consiste em compor um conjunto de imagens adquiridas em diferentes pontos de vista para construir uma imagem maior dando uma visão global da cena.

O reconhecimento de um padrão de referência em uma imagem, como por exemplo o reconhecimento de caractere em texto, verificação de assinatura em documento e impressões digitais, também está relacionado com o problema de registro. Outros exemplos seriam: localização da posição e orientação de feições conhecidas tais como pistas de aeroporto, lotes de estacionamento, tanques de guerra, etc. Esta diversidade de problemas e suas aplicações têm sido a causa para o desenvolvimento de inúmeros métodos de registro de imagens. Como o desempenho de cada metodologia é dependente da aplicação, das características do sensor, e da natureza e composição da região imageada, um único esquema de registro de imagens não funciona satisfatoriamente para as diversas situações.

## 1.1 Proposta de Trabalho

Dentro da Divisão de Eletrônica Aeroespacial (DEA) do INPE, o interesse por Registro de Imagens surgiu por causa do Projeto *CCD Imaging Instrument Experiment (CIMEX)*. Este projeto consiste de uma câmera de imageamento, baseado em tecnologia CCD (*Charge Coupled Device*), com duas bandas espectrais na mesma faixa do sensor TM do satélite Landsat, ou seja:

$$B1 : 1.55 \text{ a } 1.7\mu\text{m}$$

$$B2 : 2.08 \text{ a } 2.35\mu\text{m}$$

Este experimento voará a bordo do onibus espacial *Shuttle* (Nasa), dentro de um compartimento utilizado para pequenas cargas HH (*hitchhiker*). A parte óptica é constituída de duas objetivas colocadas paralelamente. O alinhamento dessas objetivas será feito mecanicamente até uma certa precisão. O processo de registro de imagem terá a finalidade

de refinar este alinhamento, tentando minimizar este erro.

Diante de vários métodos de registro de imagens existentes, a proposta deste trabalho é testar e validar alguns métodos de registro de imagens e calcular o erro de registro. Foram escolhidos dois métodos de registro, um manual e o outro automático. No método manual a escolha dos pontos de controle é feita visualmente através da tela do computador. O método automático é baseado em feições, o qual extrai e casa as feições das imagens. Para o cálculo do erro de registro foram utilizados dois métodos. Um deles é a raiz quadrada do erro médio quadrático ( Root Mean Square Error - RMSE) que é uma medida tradicional da precisão de registro, o outro é o Método da Variância ( uma técnica que ainda não tinha sido utilizada ) que trabalha com dois tipos de erros, o erro geométrico que é o que queremos determinar e o erro de medida gerado pelo processo que está sendo usado para calcular o erro, o qual foi implementado e testado.

Este trabalho está organizado da seguinte forma:

No Capítulo 2 descreve-se alguns conceitos sobre imagens obtidas de satélites, suas resoluções e bandas espectrais utilizadas pelos satélites mais comuns. Também comenta-se sobre detecção de borda e tipos de bordas existentes.

No Capítulo 3 descreve-se alguns conceitos matemáticos importantes no processo de registro de imagens. Inicia-se descrevendo o que é uma transformação linear até chegar numa transformação espacial.

No Capítulo 4 descreve-se a necessidade de se fazer o registro, as distorções geométricas mais

comuns presentes nas imagens de satélites e as metodologias existentes de registro de imagens.

Apresenta-se também dois métodos para calcular o erro de registro.

No Capítulo 5 mostra-se resumidamente o algoritmo utilizado para se fazer o registro automático.

No Capítulo 6 apresenta-se os resultados obtidos dos métodos de registro de imagens utilizados, acompanhados dos valores dos erros de registro encontrados.

No Capítulo 7 apresentam-se as conclusões do trabalho.

No Apêndice A mostra-se o programa fonte em linguagem C do Método da Variância.

## CAPÍTULO 2

### CONCEITOS BÁSICOS

#### 2.1 Imagens em Sensoriamento

Obtidas por satélite, fotografias aéreas ou “scaners” aerotransportados, os dados representam formas de captura indireta da informação espacial. Armazenados como matrizes, cada elemento da imagem (denominado “*pixel*”) tem um valor proporcional à reflectância do solo para a área imageada.

Pela natureza do processo de aquisição de imagens, os objetos geográficos estão contidos na imagem e para individualizá-los, é necessário recorrer a técnicas de foto-interpretação e de classificação automática.

Características importantes de imagens de satélite são: o número de bandas do espectro eletromagnético imageadas (resolução espectral), a área da superfície terrestre observada instantaneamente por cada detector (resolução espacial) e o intervalo entre duas passagens do satélite pelo mesmo ponto (resolução temporal).

#### 2.2 Imagem Digital

Em sensoriamento remoto orbital, um grande número de dados é usado para representar uma imagem, que podem ser manipulados no formato digital, afim de se extrair informações destas imagens. A cada ponto imageado pelos sensores, corresponde a uma área mínima

denominada “*pixel*” (*picture cell*), que deve estar geograficamente identificado, e para o qual são registrados valores digitais relacionados a intensidade de energia refletida em faixas (bandas) bem definidas do espectro eletromagnético. Uma imagem digital pode ser definida por uma função bidimensional, que fornece a intensidade de luz refletida ou emitida por uma cena, na forma  $I(x, y)$ , onde os valores de  $I$  representam a intensidade da imagem em cada coordenada  $(x, y)$ . Essa intensidade é representada por um valor inteiro, não-negativo e finito, chamado nível de cinza.

Para se realizar o processamento digital de imagens de dados de sensoriamento remoto, é necessário que a imagem esteja no formato digital. Existem basicamente duas maneiras de se obter uma imagem digital:

- adquirir a imagem de sensoriamento remoto no formato analógico (por exemplo, fotografia aérea), e então digitalizá-la;
- adquirir a imagem de sensoriamento remoto já no formato digital, tais como os dados em fita CCT (*Computer Compatible Tape*), gravados por satélites como *Landsat* e *Spot*.

O processo de digitalização de uma imagem não-digital (“imagem contínua”), corresponde a uma discretização (ou amostragem) da cena em observação, através da superposição de uma malha hipotética, e uma atribuição de valores inteiros (os níveis de cinza) a cada ponto dessa malha (processo chamado de quantização). Em satélites como o *Landsat* e *Spot*, o sinal elétrico detectado em cada um de seus canais é convertido, ainda a bordo do satélite, por um

sistema analógico/digital, e a saída enviada para as estações de recepção via telemetria. As imagens destes satélites são amostradas com um número grande de pontos (as imagens do sensor “*Thematic Mapper*” do satélite *Landsat* possuem mais de 6000 amostras por linha). Além disso, tais imagens têm a característica de serem multi-espectrais, no sentido de constituírem uma coleção de imagens de uma mesma cena, num mesmo instante, obtida por vários sensores com respostas espectrais diferentes.

### 2.2.1 Caracterização de Imagens

Pode-se representar uma imagem por uma matriz de dados, onde as linhas e colunas definem as coordenadas espaciais do *pixel*.

Radiância é o fluxo radiante que provêm de uma fonte, numa determinada direção, por unidade de área. Na realidade a medida da radiância representada em cada *pixel* pelo seu nível de cinza, não é apenas a da radiação refletida pela superfície contida na cena do *pixel*, mas também a radiação devida ao espalhamento atmosférico.

A quantificação da radiância contínua de uma cena é representada pelos níveis de cinza discretos na imagem digital, e é representada por um número de bits por *pixel* para produzir um intervalo de radiância. Os sensores da nova geração obtêm normalmente imagens em 8 ou 10 bits (equivalente a 256 ou 1024 níveis digitais). Na realidade o nível de cinza é representado pela radiância média de uma área relativamente pequena em uma cena. Esta área é determinada pela altitude do sistema sensor a bordo do satélite e outros parâmetros como o IFOV (*Instantaneous Field of View*), que é o ângulo formado pela projeção geométrica de um único elemento detector

sobre a superfície da Terra.

No caso das imagens multi-espectrais a representação digital é mais complexa porque, para cada coordenada  $(x, y)$ , haverá um conjunto de valores de nível de cinza. Representa-se, então, cada *pixel* por um vetor com tantas dimensões quantas forem as bandas espectrais.

**Banda espectral** é o intervalo entre dois comprimentos de onda, no espectro eletromagnético.

### 2.2.1.1 Resolução e Bandas

Resolução é uma medida da habilidade que um sistema sensor possui de distinguir entre respostas que são semelhantes espectralmente ou próximas espacialmente. A resolução pode ser classificada em espacial, espectral e radiométrica.

**Resolução espacial:** mede a menor separação angular ou linear entre dois objetos. Por exemplo, uma resolução de 20 metros implica que objetos distanciados entre si a menos que 20 metros, em geral não serão discriminados pelo sistema.

**Resolução espectral:** é uma medida da largura das faixas espectrais do sistema sensor. Por exemplo, um sensor que opera na faixa de 0.4 a 0.45 $\mu m$  tem uma resolução espectral menor do que o sensor que opera na faixa de 0.4 a 0.5  $\mu m$ .

**Resolução radiométrica:** está associada à sensibilidade do sistema sensor em distinguir dois níveis de intensidade do sinal de retorno. Por exemplo, uma resolução de 10 bits (1024 níveis digitais) é melhor que uma de 8 bits.

A Tabela 2.1 apresenta as características de resolução dos sistemas sensores *Thematic Mapper* (TM), *Haute Resolution Visible* (HRV) e *Advanced Very Resolution Radiometer* (AVHRR), a bordo dos satélites *Landsat*, *Spot* e *NOAA*, respectivamente. As diferentes bandas espectrais dos sensores tem aplicações distintas em estudos de sensoriamento remoto.

TABELA 2.1: RESOLUÇÃO DOS SISTEMAS SENSORES

RESOLUÇÃO	TM	HRV	AVHRR
Frequência da aquisição de imagens	16 dias	26 dias	2 vezes ao dia
Resolução espacial	30m 120m (banda 6)	20m (banda 1 a 3) 10m (Pan)	1.1km (nominal)
Resolução radiométrica	8 bits	8 bits(1 – 3) 6 bits (Pan)	8 bits
Resolução espectral bandas espectrais ( $\mu m$ )	banda 1: 0.45 – 0.52 banda 2: 0.52 – 0.60 banda 3: 0.63 – 0.69 banda 4: 0.76 – 0.90 banda 5: 1.55 – 1.75 banda 6: 10.74 – 12.5 banda 7: 2.08 – 2.35	banda 1: 0.50 – 0.59 banda 2: 0.61 – 0.68 banda 3: 0.79 – 0.89 pan: 0.51 – 0.73	banda 1: 0.58 – 0.68 banda 2: 0.725 – 1.1 banda 3: 3.55 – 3.93 banda 4: 10.3 – 11.3 banda 5: 11.5 – 12.5

### 2.3 Detecção de Borda

Detecção de borda é uma das operações mais usadas em análise de imagens, sendo uma ferramenta essencial para processamento [4]. A razão para isso é que as bordas formam o contorno de um objeto. Uma borda é a fronteira entre um objeto e o “background”, e indica a fronteira entre objetos sobrepostos. Isto significa que se uma borda tem uma identificação precisa, então todos os objetos podem ser localizados e propriedades básicas tal como perímetro,

área e formato podem ser calculados. A Figura 2.1 mostra um exemplo de detecção de borda. Existem dois objetos sobrepostos na imagem original (a), que possuem um fundo de cor cinza uniforme. A versão realçada da mesma imagem (b) tem linhas escuras contornando os três objetos. Notar que não dá para definir quais partes da imagem são os objetos. Somente as fronteiras entre as regiões são identificadas. No entanto sabendo que as bolhas na imagem são as regiões, pode-se determinar que a bolha número três cobre parte da bolha número dois.

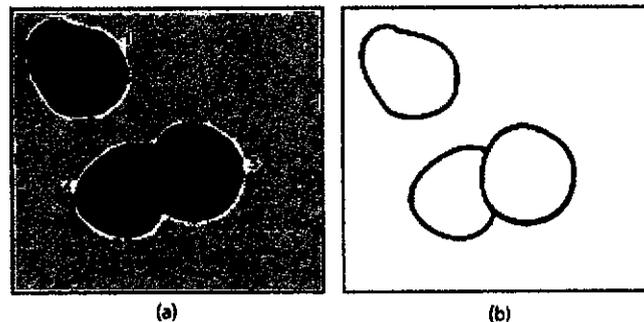


Fig. 2.1- Exemplo de detecção de borda. FONTE: [4] ( p. 2).

Consideremos a Figura 2.2(a), a qual é uma fotografia de uma seção de uma árvore. Os crescimentos dos anéis são os objetos de interesse nesta imagem. Cada anel representa um ano de vida da árvore, então o número de anéis corresponde a idade da árvore. Usando detecção de borda , como mostra a Figura 2.2(b), consegue-se separar a imagem em anéis (parte escura) e o resto da imagem é o fundo (parte branca).

Tecnicamente, detecção de borda (*edge detection*) é o processo de localizar os *pixels*, e realçar a borda (*edge enhancement*) é o processo de aumentar o contraste entre as bordas e o fundo da imagem, afim de que as bordas se tornem mais visíveis. Na prática estes termos são permutáveis, uma vez que alguns detectores de borda setam o valor do *pixel* em um determinado valor de cinza, para que eles possam ser facilmente visualizados. Rastrear a borda (*edge tracing*) é o

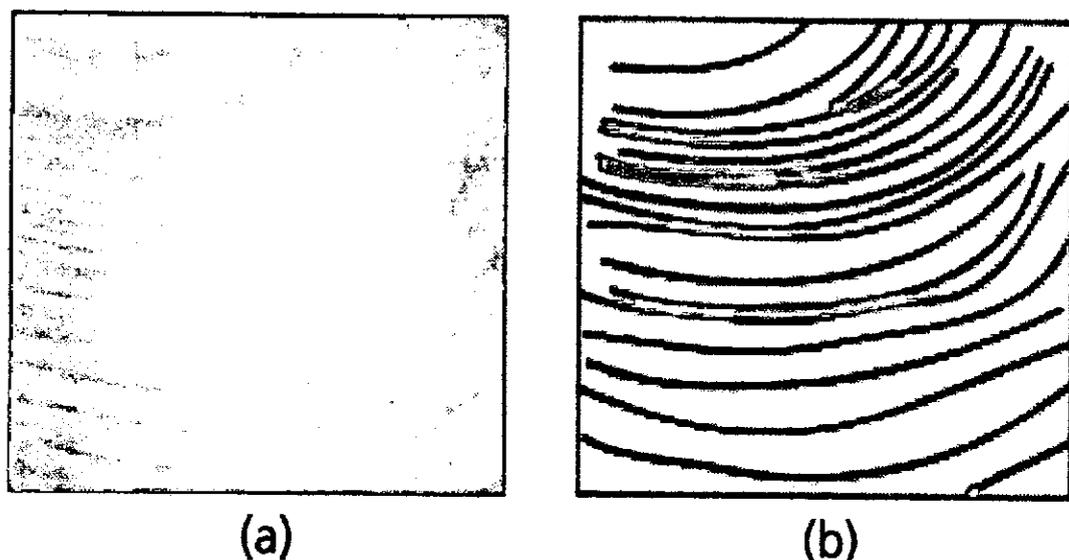


Fig. 2.2- Seção transversal de uma árvore. FONTE: [4] (p.3).

processo de seguir as bordas, usualmente reunindo os *pixels* numa lista. Isto é feito numa certa direção, sentido horário ou anti-horário ao redor dos objetos. Código de cadeia é um exemplo de um método de rastrear a borda . O resultado é uma representação bem delienada dos objetos, que pode ser usada para calcular o formato ou classificar o objeto.

### 2.3.1 Borda

A Figura 2.3(a) mostra um exemplo de uma borda ideal. Neste exemplo uma borda é simplesmente a mudança do nível de cinza da imagem em uma localização específica. Quanto maior for a mudança no nível de cinza, mais fácil será detectar a borda. Devido à digitalização da imagem, nem todas as bordas acontecem na fronteira do *pixel*. A mudança do nível de cinza pode corresponder a alguns *pixels* (Figura 2.3(b-d)). A posição da borda será considerada o centro da rampa.

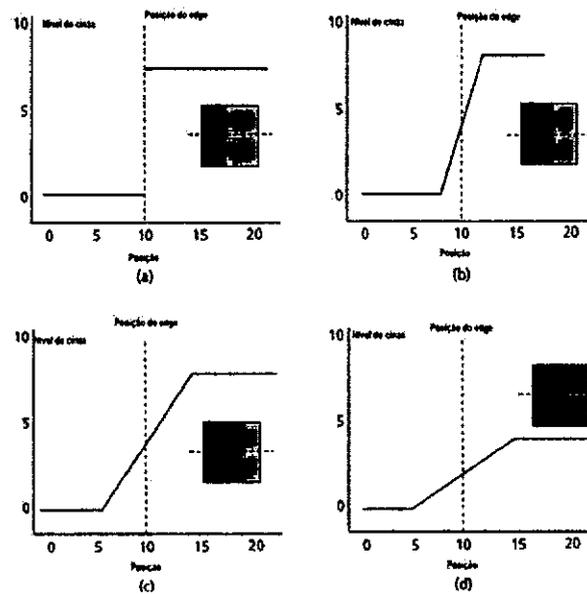


Fig. 2.3- Tipos de bordas. FONTE: [4] (p.4).

Um outro fator que dificulta a exata localização de uma borda é o ruído. Dois *pixels* que possuem o mesmo nível de cinza na cena, não terão necessariamente o mesmo nível na imagem, isto devido à fatores como intensidade da luz, tipo das lentes e da câmera, temperatura, efeitos atmosféricos e outros. Ruído é um efeito aleatório e é caracterizado somente estatisticamente. O resultado do ruído numa imagem é produzir variação aleatória de *pixel* para *pixel*.

Uma borda ideal como mostra a Figura 2.3(a) nunca acontecerá na prática, porque:

- a) Os objetos raramente possuem uma forma bem delineada;
- b) É difícil uma borda ocorrer exatamente na margem de um *pixel*;
- c) Devido ao problema de ruído.

A Figura 2.4(a) mostra o problema b) citado anteriormente. Temos uma borda ideal e um conjunto de *pixels* envolvidos. Notar que as bordas acontecem no extremo esquerdo dos *pixels*

brancos. Se a câmera mover meio *pixel* para a esquerda, a borda se move para a direita (Figura 2.4(c)), e agora parte dos *pixels* ao longo da borda são brancos e parte são pretos. Desde que o valor do nível de cinza da imagem é dado por:

$$v = \frac{(v_w a_w + v_b a_b)}{a_w + a_b}$$

onde  $v_w$  e  $v_b$  são os níveis de cinza da imagem das regiões brancas e pretas, e  $a_w$  e  $a_b$  são as áreas das partes brancas e pretas do pixel, este movimento da câmera refletirá numa média do nível de cinza.

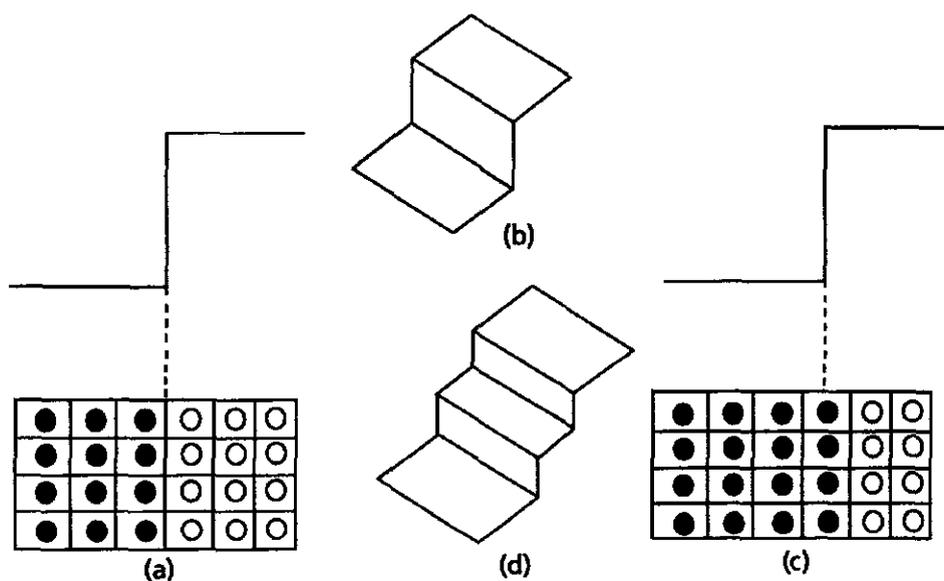


Fig. 2.4- Modelos de bordas. FONTE: [4] (p.5).

### CAPÍTULO 3

#### CONCEITOS MATEMÁTICOS

#### 3.1 Transformação Linear

Considere a transformação  $\mathfrak{R}^2 \rightarrow \mathfrak{R}^2$  na forma  $U_1 = g_1(X_1, X_2)$  e  $U_2 = g_2(X_1, X_2)$ . No caso especial em que  $g_1$  e  $g_2$  são funções lineares homogêneas, temos que[5] :

$$\begin{aligned}U_1 &= a_{11}X_1 + a_{12}X_2, \\U_2 &= a_{21}X_1 + a_{22}X_2.\end{aligned}\tag{3.1}$$

A inversa da transformação é obtida resolvendo a equação (3.1) para  $X_1$  e  $X_2$  em termos de  $U_1$  e  $U_2$ . Isto pode ser feito se o determinante da transformação da equação (3.1),  $a = a_{11}a_{22} - a_{12}a_{21}$ , for diferente de zero. Assumindo que este é o caso, escrevemos a transformação inversa como:

$$\begin{aligned}X_1 &= b_{11}U_1 + b_{12}U_2, \\X_2 &= b_{21}U_1 + b_{22}U_2.\end{aligned}\tag{3.2}$$

onde:

$$\begin{aligned}b_{11} &= a_{22}/a & b_{12} &= -a_{12}/a \\b_{21} &= -a_{21}/a & b_{22} &= a_{11}/a\end{aligned}$$

Se  $b$  indica o determinante da equação (3.2), então:

$$b = b_{11}b_{22} - b_{12}b_{21} = 1/a.$$

Vamos considerar a transformação linear equação (3.1) e sua inversa equação (3.2) do ponto de

vista geométrico. Podemos considerar a equação (3.1) o mapeamento de um ponto  $(X_1, X_2)$  do plano  $x$  em um ponto  $(U_1, U_2)$  do plano  $u$  (Figura 3.1). Do mesmo modo a transformação inversa equação (3.2) mapeia  $(U_1, U_2)$  em  $(X_1, X_2)$ .

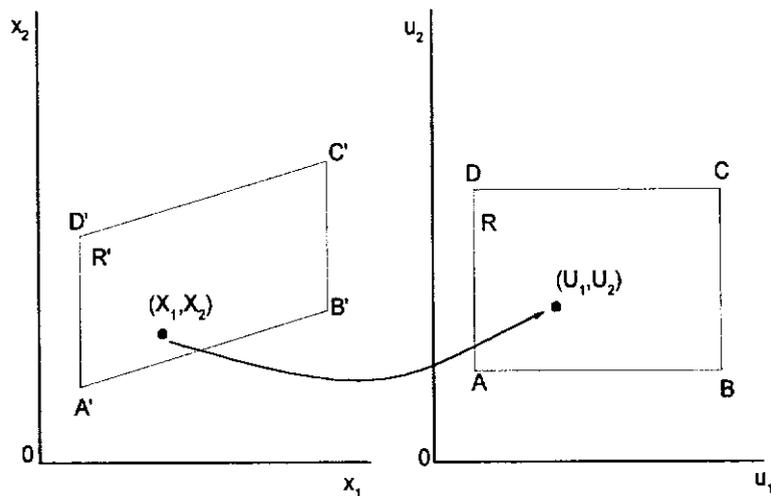


Fig. 3.1- Transformação linear. FONTE:[5] ( p. 238).

Consideremos o retângulo  $R$  do plano  $u$  com lados paralelos aos eixos. Sejam seus vértices dados por  $A(u_1, u_2)$ ,  $B(u_1 + \Delta u_1, u_2)$ ,  $C(u_1 + \Delta u_1, u_2 + \Delta u_2)$ ,  $D(u_1, u_2 + \Delta u_2)$ . A imagem desses pontos depois da transformação equação (3.2) será:

$$A' : (b_{11}u_1 + b_{12}u_2, b_{21}u_1 + b_{22}u_2),$$

$$B' : (b_{11}(u_1 + \Delta u_1) + b_{12}u_2, b_{21}(u_1 + \Delta u_1) + b_{22}u_2),$$

$$C' : (b_{11}(u_1 + \Delta u_1) + b_{12}(u_2 + \Delta u_2), b_{21}(u_1 + \Delta u_1) + b_{22}(u_2 + \Delta u_2)),$$

$$D' : (b_{11}u_1 + b_{12}(u_2 + \Delta u_2), b_{21}u_1 + b_{22}(u_2 + \Delta u_2)).$$

Pode ser facilmente mostrado que o interior do retângulo  $R$  está mapeado no interior do

quadrilátero  $R'$ , unindo os pontos  $A', B', C'$  e  $D'$ , e este quadrilátero é um paralelogramo, cuja área  $R'$  está relacionada à área  $R$  pela equação:

$$AreaR' = |b| AreaR,$$

onde  $|b|$  é o valor absoluto do determinante da transformação da equação (3.2). Além disso, se  $(U_1, U_2)$  cai em  $R$ , sua imagem  $(X_1, X_2)$  cairá em  $R'$ , deste modo:

$$(U_1, U_2) \in R \iff (X_1, X_2) \in R'.$$

**Generalizando para o caso de  $n$  variáveis:** supondo  $X_1, \dots, X_n$  um ponto do  $\mathbb{R}^N$ . Se

$U_1 = g_1(X_1, \dots, X_n), \dots, U_n = g_n(X_1, \dots, X_n)$  são transformações lineares, temos:

$$\begin{aligned} U_1 &= a_{11}X_1 + \dots + a_{1n}X_n \\ &\dots\dots\dots \\ &\dots\dots\dots \\ U_n &= a_{n1}X_1 + \dots + a_{nn}X_n \end{aligned} \tag{3.3}$$

Seja  $a$  o determinante da transformação:

$$a = \begin{vmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{nn} \end{vmatrix}$$



A matriz  $A$  também pode ser escrita na forma:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} r_1 \cos \theta_1 & -r_2 \operatorname{sen} \theta_2 \\ r_1 \operatorname{sen} \theta_1 & r_2 \cos \theta_2 \end{pmatrix}$$

onde  $(r_1, \theta_1)$  são as coordenadas polares do ponto  $(a, c)$  e  $(r_2, (\theta_2 + \pi/2))$  são as coordenadas polares do ponto  $(b, d)$ . A transformação linear:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

em  $\mathbb{R}^2$  mapeia um paralelogramo com o vértice na origem em outro paralelogramo com vértice na origem como ilustrado na Figura 3.2. Notar pela Figura 3.3 que o paralelogramo pode ser invertido pela transformação. Uma transformação afim geral  $w(x) = Ax + t$  em  $\mathbb{R}^2$  consiste

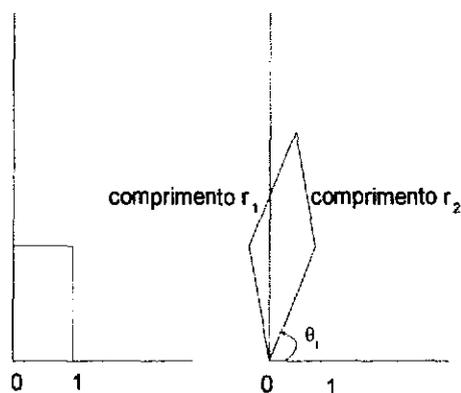


Fig. 3.2- Transformação afim.FONTE: [6] (p.51).

de uma transformação linear  $A$  a qual deforma o espaço relativo a origem, seguida de uma translação ou deslocamento especificada pelo vetor  $t$  (Figura 3.4).

Vamos mostrar como encontrar uma transformação afim que transforma uma folha grande numa

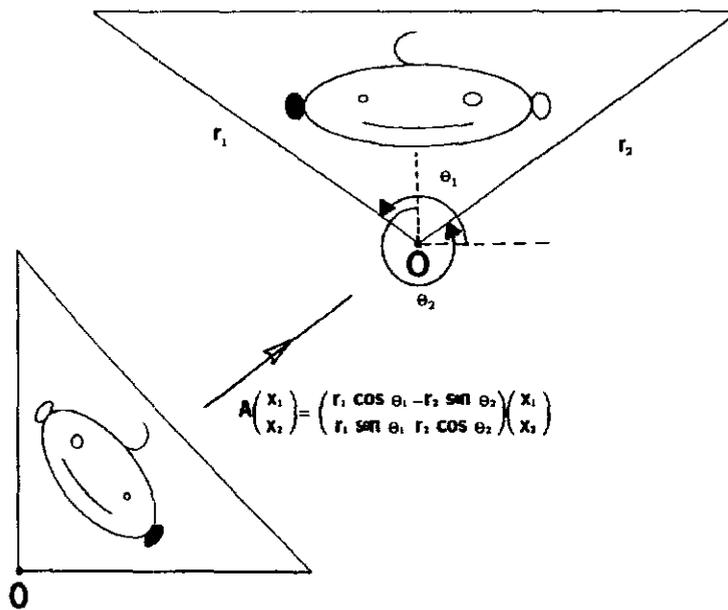


Fig. 3.3- Uma transformação afim pode inverter figuras. FONTE: [6] (p. 51).

folha pequena Figura(3.5). Desejamos encontrar os números  $a, b, c, d, e,$  e  $f$  tal que:

$$w(\text{Folha grande}) \cong \text{Folha pequena}$$

Vamos introduzir os eixos das coordenadas  $x$  e  $y$ , como mostra a Figura(3.5). Marcar três pontos na folha grande e determinar suas coordenadas  $(x_1, x_2), (y_1, y_2),$  e  $(z_1, z_2)$ . Marcar os pontos correspondentes na pequena folha e determinar suas coordenadas  $(\tilde{x}_1, \tilde{x}_2), (\tilde{y}_1, \tilde{y}_2),$  e  $(\tilde{z}_1, \tilde{z}_2)$  respectivamente. Então  $a, b, e$  são obtidas resolvendo as três equações lineares:

$$x_1 a + x_2 b + e = \tilde{x}_1,$$

$$y_1 a + y_2 b + e = \tilde{y}_1,$$

$$z_1 a + z_2 b + e = \tilde{z}_1;$$

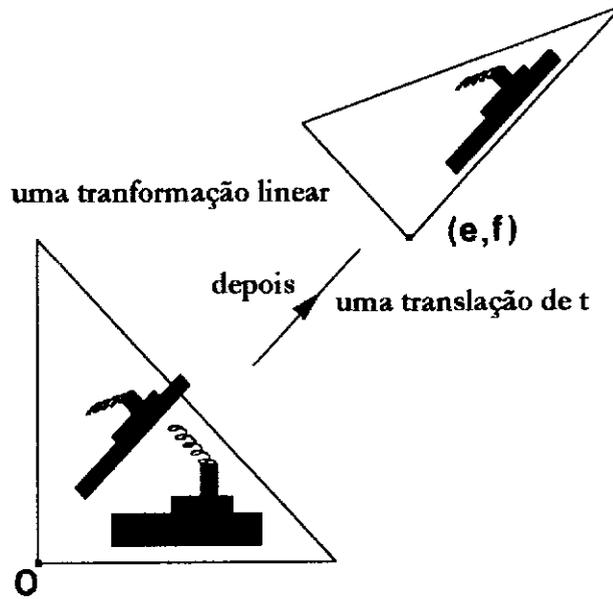


Fig. 3.4- Transformação afim. FONTE:[6] (p. 52).

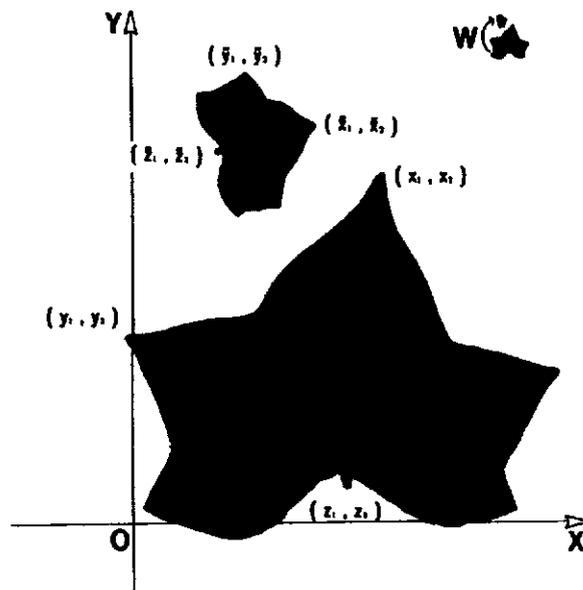


Fig. 3.5- Duas folhas determinam uma transformação afim. FONTE: [6] (p. 52).

enquanto  $c, d,$  e  $f$  satisfazem:

$$x_1c + x_2d + f = \tilde{x}_2,$$

$$y_1c + y_2d + f = \tilde{y}_2,$$

$$z_1c + z_2d + f = \tilde{z}_2.$$

Uma transformação  $w : \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$  é chamada de similaridade se ela é uma transformação afim tendo uma das seguintes formas:

$$w \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} r \cos \theta & -r \operatorname{sen} \theta \\ r \operatorname{sen} \theta & r \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad (3.6)$$

$$w \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} r \cos \theta & r \operatorname{sen} \theta \\ r \operatorname{sen} \theta & -r \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad (3.7)$$

para alguma translação  $(e, f) \in \mathfrak{R}^2$ , um número real  $r \neq 0$ , e um ângulo  $\theta$ ,  $0 \leq \theta < 2\pi$ , onde  $\theta$  é chamado de ângulo de rotação e  $r$  é o fator de escala. Exemplos importantes são:

$$R_\theta \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \cos \theta & -\operatorname{sen} \theta \\ \operatorname{sen} \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

que é uma rotação, e a transformação linear:

$$R \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

que é uma reflexão. Notamos que a transformação de similaridade mantêm o ângulo relativo entre dois objetos; por isto pode também ser caracterizada como uma transformação rígida.

### 3.3 Transformações Espaciais

Uma transformação espacial é uma função de mapeamento que estabelece a correspondência espacial entre todos os pontos de uma imagem de entrada e a imagem de saída. No processo de registro de imagens, esta função modela matematicamente a distorção da imagem de entrada (imagem de ajuste) em relação à imagem de saída (imagem de referência), ver Figura 3.6.

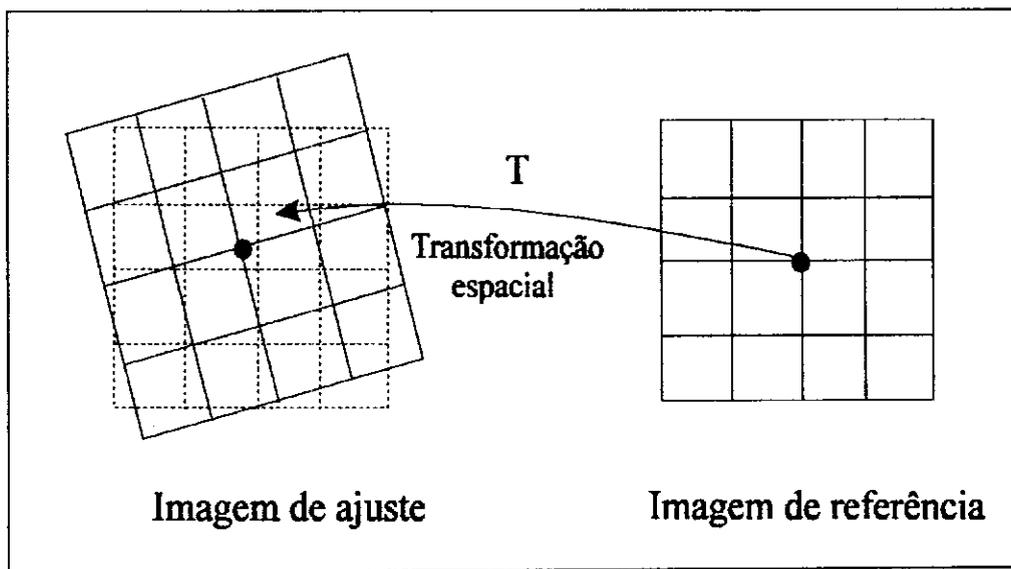


Fig. 3.6- Transformação espacial. FONTE: [2] ( p. 30).

Nos métodos tradicionais de registro, as funções de mapeamento são determinadas usando os pontos de controle selecionados no processo de casamento. Este esquema é bastante utilizado em situações nas quais um grande número de pontos de controle são, a princípio, obtidos com boa precisão. Dados  $n$  pontos de controle nas duas imagens de uma mesma região geográfica, a função de mapeamento pode ser formulada da seguinte maneira:

$$X_i = u(x_i, y_i), \quad (3.8)$$

$$Y_i = \nu(x_i, y_i)$$

onde  $(x_i, y_i)$  e  $(X_i, Y_i)$  correspondem as coordenadas dos pontos na imagem de referência e na imagem de ajuste, respectivamente. As funções  $u$  e  $\nu$  especificam a transformação espacial e, mapeiam as coordenadas dos pontos da imagem de referência para as coordenadas dos correspondentes pontos na imagem de ajuste. Este tipo de mapeamento é conhecido por mapeamento inverso, porque mapeia a imagem de saída sobre a imagem de entrada [8].

O problema de determinar as funções  $u$  e  $\nu$  pode ser convenientemente abordado como um problema de ajuste de superfície [8]. A derivação das funções de mapeamento é equivalente a determinar duas superfícies suaves: uma que passa pelos pontos  $(x_i, y_i, X_i)$ , e outra que passa por  $(x_i, y_i, Y_i)$ , para  $1 \leq i \leq n$ . A Figura 3.7 mostra a superfície para  $u(x_i, y_i)$  com os pontos de controle representados nos pontos da grade. Pode-se observar que na ausência de distorções ( $x_i = X_i$  e  $y_i = Y_i$ ) as superfícies são do tipo rampas. A introdução de distorções faz com que as superfícies se modifiquem de acordo com a complexidade do modelo de distorção.

As funções de mapeamento usadas para alinhar duas imagens podem ser globais ou locais. Uma transformação global é dada por uma única equação que registra otimamente todos os *pixels* nas duas imagens. Na transformação local há várias equações, onde cada uma delas é usada para uma determinada região da imagem. As transformações locais, geralmente, são mais precisas mas exigem maior custo computacional. As transformações globais mais frequentemente usadas, para modelar as distorções espaciais entre duas imagens, são as transformações afins e polinomiais. A transformação de similaridade é suficiente para registrar duas imagens com distorção rígida,

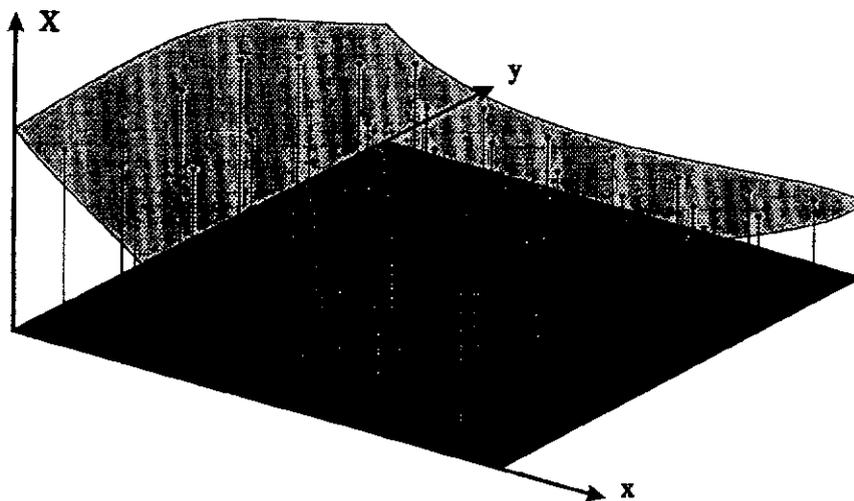


Fig. 3.7- Superfície  $u(x_i, y_i)$ . FONTE: [9] ( p. 6).

e é composta dos operadores cartesianos de escala , translação e rotação como segue abaixo:

$$\begin{aligned} X_i &= e + r(x_i \cos \theta - y_i \text{sen} \theta), \\ Y_i &= f + r(x_i \text{sen} \theta + y_i \cos \theta) \end{aligned} \quad (3.9)$$

No caso das imagens possuírem distorções mais significativas, ou de não se ter muitas informações sobre a transformação necessária para alinhar as imagens, transformações afins ou polinomiais são mais freqüentemente usadas. Elas podem ser representadas como duas funções de mapeamento espacial da forma:

$$X = \sum_{i=0}^m \sum_{j=0}^i a_{ij} x^i y^{j-1}, \quad (3.10)$$

$$Y = \sum_{i=0}^m \sum_{j=0}^i b_{ij} x^i y^{j-1}. \quad (3.11)$$

onde  $a_{ij}$  e  $b_{ij}$  são os coeficientes dos polinômios a serem determinados e,  $m$  é o grau do polinômio. O modelo polinomial é suficientemente geral para poder representar vários tipos de transformações geométricas. Assim o modelo de ordem 1 pode representar as transformações afins.. O modelo de ordem 2 é apropriado para representar, por exemplo, as perspectivas. Para muitas aplicações os polinômios de segunda ou terceira ordem são suficientes [2] . Polinômios de ordem maior do que três devem ser usados com muita precaução devido ao comportamento imprevisível das funções polinomiais de alto grau [7] . Ou seja, os polinômios de alto grau geram resultados com boa precisão na vizinhança dos pontos de controle, mas por outro lado podem gerar distorções em regiões da imagem fora desta vizinhança. Na prática é aconselhado usar a menor ordem  $m$ , suficiente para modelar as distorções geométricas da imagem, e usar um número grande de pontos de controle com máxima precisão, descartando aqueles pares duvidosos.

## CAPÍTULO 4

### MÉTODOS DE REGISTRO DE IMAGENS

#### 4.1 Registro de Imagens

A necessidade de fazer o registro de imagens existe principalmente quando se deseja combinar duas imagens diferentes de uma mesma área. Por exemplo, para imagens de satélite de diferentes épocas (multi-temporais), onde deseja comparar mudanças ocorridas em uma determinada área. Outro exemplo comum de necessidade do registro é na integração de imagens de diferentes sensores (HRV e TM, por exemplo), ou na confecção de mosaicos a partir de imagens adjacentes de uma área.

Quando uma imagem é adquirida por sensores a bordo de satélites ou aeronaves ela contém distorções radiométricas e geométricas. As distorções radiométricas afetam os valores dos níveis de cinza da imagem, e são provocadas pelo efeito atmosférico, e erros de medição dos instrumentos [7]. As distorções geométricas mudam a geometria da imagem de modo que a posição, o tamanho e a forma do *pixel* são alterados durante o processo de aquisição de imagens. São várias as fontes de erros geométricos tais como a rotação da terra durante o processo de aquisição, a curvatura da terra, variações da velocidade e altitude da plataforma, mudanças na elevação topográfica, etc [7]. Alguns destes efeitos são mostrados na Figura 4.1. Distorções sistemáticas como aquelas provocadas pela rotação da terra e órbita não-polar do satélite, podem ser corrigidas usando modelos orbitais e dados de calibração. Outras distorções imprevisíveis devem ser removidas com a ajuda de pontos de controle e modelos matemáticos de distorção,

que é o procedimento usado no processo de registro de imagens como será visto mais adiante.

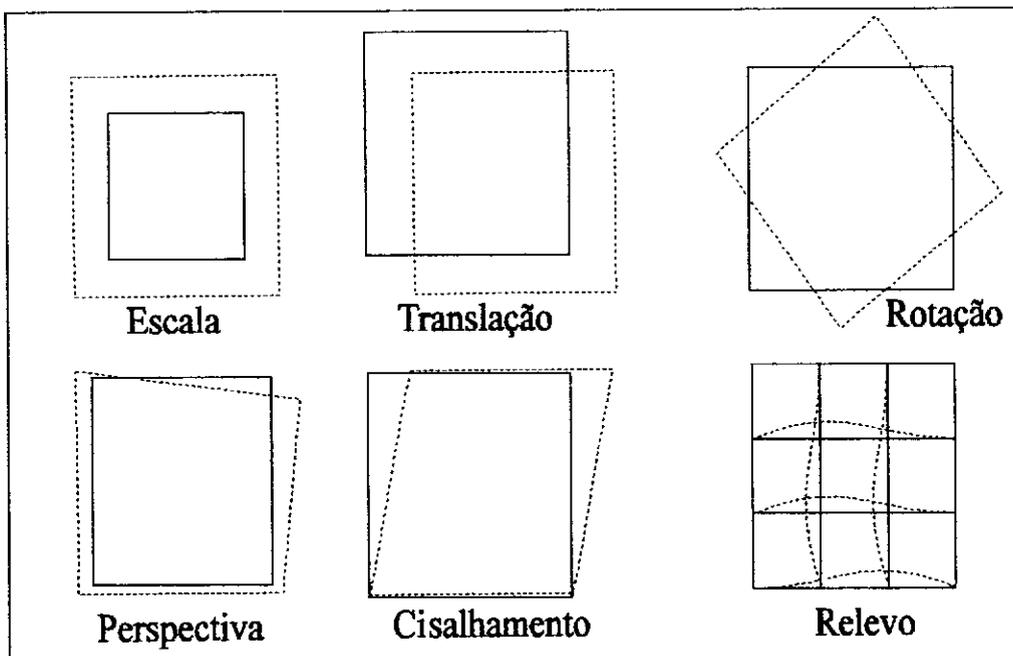


Fig. 4.1- Efeitos de distorções geométricas. FONTE: [8] ( p.62).

Tradicionalmente, o processo de registro automático ou semi-automático de imagens é realizado em quatro fases:

- **Identificação de feições:** identifica um conjunto de feições (objetos) relevantes nas duas imagens, tais como intersecções de linhas, contornos de regiões, regiões, pontos de borda, etc. Estas feições devem estar presentes e, de preferência, bem delineadas em ambas as imagens.
- **Casamento das feições:** estabelece a correspondência entre as feições. Cada feição na imagem de ajuste é casada com a correspondente feição na imagem de referência. As feições, assim casadas, são identificadas por coordenadas espaciais que identificam

as suas posições nas imagens. Estas coordenadas espaciais são usualmente chamadas de pontos de controle.

- **Transformação Espacial:** determina as funções de mapeamento. Estas funções modelam a transformação espacial, que será aplicada à imagem de ajuste, usando a informação sobre os pontos de controle obtidos na fase anterior.
- **Interpolação:** reamostra a imagem de ajuste sobre a grade de pontos da imagem de referência.

De um modo geral, os algoritmos de registro são diferentes no sentido de que eles combinam técnicas diferentes para identificação e casamento de feições, e funções de mapeamento.

A tarefa mais difícil, no processo de registro de imagens, é a identificação e a obtenção da correspondência entre o conjunto de feições para obter os pontos de controle. Esta tarefa é crucial para a precisão do registro das imagens. Conhecidos os pontos de controle, aplica-se o processo de interpolação e as técnicas mais utilizadas são do Vizinheiro Mais Próximo (VMP), Bilinear e Convolução Cúbica [10].

### 4.1.1 REGISTRO MANUAL

O método de registro manual é o mais utilizado nas diversas aplicações da área de sensoriamento remoto. Neste método as fases de identificação de feições nas imagens, e o casamento de feições são realizadas simultaneamente, e com assistência de um intérprete. As imagens são visualizadas no monitor do computador e o usuário escolhe as feições correspondentes que aparecem nas duas imagens. Possíveis candidatos incluem lagos, rios, cruzamento de ruas, pontes, ou qualquer outro objeto que seja claramente visível nas duas imagens. A cada objeto identificado é designado uma coordenada na imagem, e os pares de coordenadas correspondentes são referidos como *pontos de controle*. A Figura 4.2 ilustra o processo de registro manual mostrando três pares de feições, onde os quadrados pretos indicam as posições, na tela, dos *pontos de controle*.

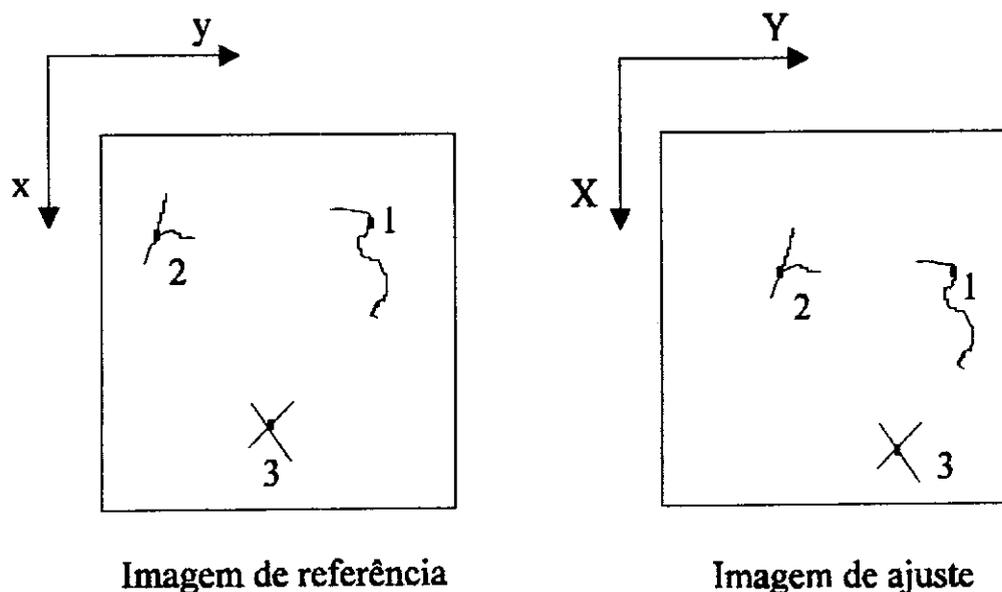


Fig. 4.2- Registro manual. FONTE: [7] (p.51).

Neste trabalho trabalhou-se com imagens do sensor *TM* (512 x 544) e os pares de pontos de controle foram obtidos utilizando o SPRING (Sistema de Processamento de Informações Georeferenciadas)[11, 12], pois ele possui um bom recurso de zoom, possibilitando uma melhor precisão na marcação dos pontos.

Depois de obtidos os pares de pontos, aplicou-se uma função que mapeia as coordenadas da imagem de ajuste (imagem que se deseja registrar) com as coordenadas da imagem de referência (imagem tomada como referência). Esta função é uma transformação de similaridade (equação 3.9 do Capítulo 3). Em todas as imagens foram escolhidos 10 pares de pontos, sendo que 6 serão usados pela função de transformação e 4 serão usados como pontos de teste. O número de pontos de controle para se fazer o registro deve ser maior ou igual ao número de coeficientes do polinômio usado para fazer a transformação. No entanto devido a imprecisão na localização destes pontos é melhor ter alguns pontos extras, como exemplo, duas vezes o número de pontos de controle que coeficientes. Desse modo o efeito de poucos pontos ruins é reduzido [10]. Como para este caso utiliza-se uma transformação de similaridade (4 parâmetros), somente dois pares de pontos de controle seriam suficientes para se obter uma transformação exata, mas adotando a sugestão de [10] em utilizar alguns pontos extras, foram selecionados 6 pontos de controle. Escolhemos somente 4 pontos de teste porque a imagem é pequena e também por causa da dificuldade de obtê-los devido as imagens serem multi-temporais. É mais conveniente usar poucos pontos corretos do que muitos duvidosos.

Depois da transformação a imagem precisa ser reamostrada porque as coordenadas da imagem processada (linha e coluna) não coincidem com aquelas da imagem original. A técnica de reamostragem utilizada é o método de interpolação bilinear, no qual o valor do *pixel* da imagem

corrigida é determinado pelos 4 *pixels* vizinhos.

Para se obter um registro de boa precisão um grande número de *pontos de controle*, bem distribuídos em toda a extensão da imagem devem ser selecionados. Isto é uma tarefa muito tediosa e repetitiva, e muitas vezes o processo não é suficiente quando a quantidade de dados é muito grande. Então, há a necessidade de técnicas de registro automático ou semi-automático que requeiram pouca ou nenhuma supervisão humana.

## 4.1.2 REGISTRO AUTOMÁTICO

### 4.1.2.1 Registro Baseado Em Área

Nos métodos baseados em área, uma pequena janela de pontos na imagem de referência é estatisticamente comparada com janelas de mesmo tamanho na imagem de ajuste. Este processo é ilustrado na Figura 4.3.

Considere a imagem de ajuste  $S$  com  $M$  linhas e  $N$  colunas, e  $n$  janelas  $W_z$ ,  $z = 1, \dots, n$ , com  $K$  linhas e  $L$  colunas extraídas da imagem de referência  $R$  e centradas no ponto  $(a_z, b_z)$ . Seja  $S_{ij}$  uma subimagem de  $S$  com sua coordenada direita e superior representada por  $(i, j)$ , onde:

$$S_{ij}(l, m) = S(i + l, j + m) \quad (4.1)$$

para  $0 \leq l \leq k - 1, 0 \leq m \leq L - 1, 0 \leq i \leq M - K, 0 \leq j \leq N - L$ . Cada janela  $W_z$  é comparada com todas as subimagens  $S_{ij}$  na imagem  $S$ . Depois de encontrar a subimagem  $S_{ij}$

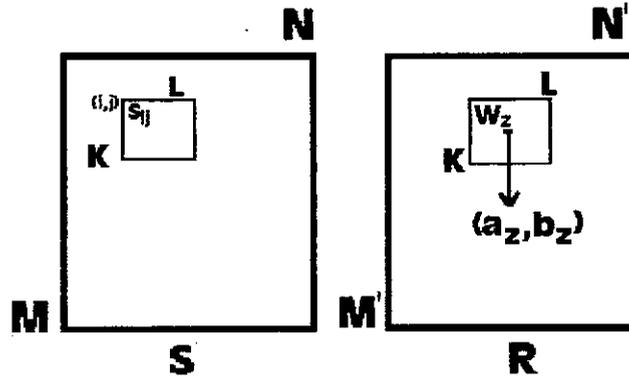


Fig. 4.3- Registro baseado em área. FONTE: [2] (p.33).

que melhor casa com  $W_z$ , seus centros  $(a_z, b_z)$  e  $(i + (K - 1)/2, j + (L - 1)/2)$  são tomados como pontos de controle. Então, esses pontos de controle podem ser usados para calcular os parâmetros da função de mapeamento.

O processo de comparação usa medidas de similaridade entre duas janelas. As mais comuns são a correlação normalizada, coeficiente de correlação, e detecção de similaridade seqüencial [13]. Para a janela  $W_z$  e a subimagem  $S_{ij}$ , definida acima, a correlação normalizada pode ser matematicamente representada por:

$$R(i, j) = \frac{\sum_{l=0}^{K-1} \sum_{m=0}^{L-1} W_z(l, m) S_{ij}(l, m)}{\sqrt{\sum_{l=0}^{K-1} \sum_{m=0}^{L-1} W_z^2(l, m) \sum_{l=0}^{K-1} \sum_{m=0}^{L-1} S_{ij}^2(l, m)}} \quad (4.2)$$

o melhor casamento ocorre quando o valor de  $R(i, j)$  é máximo. O coeficiente de correlação

mede a similaridade entre duas janelas sobre uma escala absoluta de  $[-1, 1]$ ,

$$R(i, j) = \frac{\sum_{l=0}^{K-1} \sum_{m=0}^{L-1} (W_z(l, m) - \mu_z)(S_{ij}(l, m) - \mu_s)}{\sqrt{\sum_{l=0}^{K-1} \sum_{m=0}^{L-1} (W_z(l, m) - \mu_z)^2 \sum_{l=0}^{K-1} \sum_{m=0}^{L-1} (S_{ij}(l, m) - \mu_s)^2}} \quad (4.3)$$

onde  $\mu_z$  e  $\mu_s$  são as médias (valor de intensidade média) da janela  $W_z$  e da subimagem  $S_{ij}$ , respectivamente.

Por muito tempo, o método de correlação foi limitado ao problema de registro onde as imagens eram desalinhadas por uma pequena diferença de rotação. Uma outra técnica [14] propõe que a rotação seja corrigida primeiro, usando um método de estimação da direção da iluminação, e determinando os pontos de controle através da correlação no domínio do espaço.

Os métodos de correlação podem ser eficientemente implementados no domínio da frequência usando a Transformada Rápida de Fourier. Além disso, algumas propriedades da Transformada de Fourier podem ser usadas para adquirir invariância a translação, rotação e escala. Por exemplo, se duas imagens se diferenciam apenas por um deslocamento, a diferença de fase da transformada de Fourier das duas imagens pode ser usada para estimar este deslocamento. Alguns trabalhos, usando técnicas de Fourier, foram desenvolvidos para correção tanto do efeito de translação, como o de rotação e escala [15]. Estas técnicas são preferíveis às técnicas de correlação no caso de as duas imagens, a serem registradas, terem sido adquiridas sob diferentes condições de iluminação [2].

Os métodos de correlação não são muito adequados para o registro de imagens obtidas por diferentes sensores. As imagens podem possuir diferentes características espectrais e neste caso

o método de correlação gera resultados não confiáveis. Isto é, se houver uma significativa diferença entre as imagens na vizinhança do ponto de controle que se deseja localizar, o método baseado em área pode falhar. Neste contexto, os métodos baseados em feições, os quais extraem e casam as feições comuns nas duas imagens, são tidos como mais apropriados para o registro de imagens de diferentes sensores.

#### **4.1.2.2 Registro Baseado Em Feições**

No método de registro baseado em feições duas tarefas são envolvidas no processo de identificação dos pontos de controle: extração de feições e casamento de feições. Na fase de extração de feições a imagem é representada, em uma forma compacta, por um conjunto de feições e o processo de casamento é realizado neste mesmo espaço de representação. A imagem, neste caso, pode ser representada no domínio do espaço ou no domínio da transformada.

No domínio do espaço, as feições mais comumente usadas são bordas, regiões, linhas, intersecções de linhas, centróides das regiões, pontos de descontinuidades na curvatura, etc. Os contornos das regiões e bordas têm sido freqüentemente usados como feições. Estas são extraídas por técnicas de segmentação e detecção de bordas tais como o operador de Canny, operador Laplaciano da Gaussiana (LoG) (que será visto no Capítulo 5) e crescimento de região [2]. Uma região numa imagem é um conjunto de pontos ligados, ou seja, de qualquer ponto da região pode-se chegar a qualquer outro ponto por um caminho inteiramente contido na região. É possível fazer o crescimento de regiões segundo a varredura ou por busca em todas as direções. Ao contrário do crescimento de regiões segundo a varredura, onde várias regiões são crescidas simultaneamente, no crescimento por busca em todas as direções uma única região é crescida

por vez. O método é semelhante ao da perseguição de um contorno em todas as direções [16, 17].

Geralmente os algoritmos de casamento de feições usam atributos de forma (perímetro, momentos invariantes à translação, alongamento, cor, textura, etc), e relações definidas pelo arranjo espacial. Normalmente se procura usar propriedades invariantes à rotação, escala, e à variação de níveis de cinza da imagem no processo de casamento de feições. Cada objeto na primeira imagem é comparado com os objetos candidatos em potencial ao casamento na segunda imagem, usando os seus atributos. Um par de objetos que possui atributos similares são aceitos como casados.

No cálculo da função de mapeamento, os centróides de regiões fechadas, pontos salientes ao longo de um contorno, ou posições de máxima curvatura podem ser usados como pontos de controle. Como exemplo, Li[18] usam os centróides dos contornos fechados e pontos salientes dos contornos abertos dos objetos nas imagens.

No domínio da transformada, a imagem pode ser decomposta e representada como um conjunto de coeficientes. Por exemplo, quando a Transformada de Fourier é usada para representar a imagem no domínio da frequência, a informação de borda na imagem pode ser adquirida considerando o conteúdo das altas frequências no espectro. Entretanto, o problema é que a Transformada de Fourier dá uma informação global da distribuição das frequências, mas não tem informação de localização espacial necessária para realizar o casamento. Transformações espaciais tais como a Transformada de Wavelet são mais adequadas à esta aplicação [19, 20].

Independentemente da técnica utilizada, para que um registro de boa precisão seja garantido é fundamental que as feições escolhidas como pontos de controle satisfaçam as seguintes condições [2] :

1. estejam presentes em ambas as imagens;
2. sejam bem distribuídas em toda a imagem;
3. estejam localizadas em áreas de alto contraste, e
4. sejam únicas nas suas áreas de vizinhança.

## **4.2 Cálculo do Erro de Registro**

### **4.2.1 Primeiro Método: RMSE**

Uma medida tradicional para verificar a precisão do registro das imagens é a raiz quadrada do erro médio quadrático (*Root Mean Square Error*-RMSE) entre os pontos de controle casados depois da transformação. Acontece que este valor não é confiável, uma vez que ele é aplicado somente nos pares de pontos de controle que fazem parte da equação de mapeamento, e não em qualquer ponto da imagem. Uma outra alternativa melhor [10] , adotada neste trabalho, é escolher pontos de teste em ambas as imagens, e então aplicar este cálculo (RMSE). Estes pontos não serão usados pela transformação, mas os seus resíduos serão usados para calcular a precisão do registro, o que aumenta a sua confiabilidade.

Para calcular este erro precisamos das coordenadas de pares de pontos da imagem, antes e depois

do registro. Para determinar as coordenadas antes do registro, utilizamos o SPRING. Isto já foi feito visualmente, quando na determinação dos pontos de controle, onde foram selecionados 10 pontos e escolhemos 4 como pontos de teste. Em todas as imagens foram escolhidos somente 4 pontos de teste, porque a imagem é pequena e também por causa da dificuldade de obtê-los. Devido as imagens serem multi-temporais existe grandes mudanças o que torna difícil encontrar o mesmo ponto em ambas imagens. O critério para escolha destes pontos é que eles estejam espalhados pela imagem e sejam únicos na sua área de vizinhança. Na prática, exemplos numéricos mostram isso, que se aumentarmos muito o número de pontos de controle, o erro RMSE vai saturar, com isso é mais conveniente escolhermos alguns pontos de controle para fazer o registro e deixarmos alguns pontos de teste para calcular a precisão do registro.

Para determinar as coordenadas depois do registro, utilizamos os parâmetros gerados pela transformação, isto é, escala, rotação e translações. Aplicando estes parâmetros nos pontos de teste, calculamos suas novas coordenadas usando a seguinte equação:

$$\hat{X}_i = r[x_i \cos \theta + y_i \text{sen} \theta] + \Delta x \quad (4.4)$$

$$\hat{Y}_i = r[-x_i \text{sen} \theta + y_i \cos \theta] + \Delta y$$

onde  $x_i$  e  $y_i$  são as coordenadas da imagem de ajuste,  $r$ ,  $\theta$ ,  $\Delta x$  e  $\Delta y$  ( $\Delta x = e$  e  $\Delta y = f$ , ver eq.3.9 do cap.3) são os parâmetros gerados pelo algoritmo, e  $\hat{X}_i$  e  $\hat{Y}_i$  são as novas coordenadas dos pontos de teste. De posse dos valores das coordenadas desses pontos de teste, antes e depois do registro, é possível calcular o erro de registro (RMSE) através da equação:

$$RMSE = \left\{ \sum_{i=1}^n \left( (\hat{X}_i - X_i)^2 + (\hat{Y}_i - Y_i)^2 \right) / n \right\}^{1/2} \quad (4.5)$$

onde  $\hat{X}_i$  e  $\hat{Y}_i$  são as novas coordenadas dos pontos de teste calculados anteriormente, e  $X$  e  $Y$  são as coordenadas da imagem de referência. Os resultados obtidos estão mostrados no Capítulo 6.

É importante salientar que o RMSE é um valor médio, é um dado estatístico que na prática deve ser arredondado. Se assumimos que este erro possui uma distribuição gaussiana, então não devemos ter um erro de registro maior que  $3\sigma$  na imagem inteira. Como exemplo, se o RMSE = 0.5 pixel, não encontraremos um erro maior do que 1.5 pixel em toda a imagem.

#### 4.2.2 Segundo Método: Variância

Um outro método de calcular o erro de registro de imagens proposto por [10] é o Método da Variância. Muitas vezes o erro de registro que desejamos calcular é da mesma ordem do erro de medida (gerado pelo processo que está sendo usado para medir o erro), e é então necessário separar estes dois tipos de erros. Seja  $o$  o erro observado ou medido em um ponto. Assumamos que este erro  $o$  é a soma do erro geométrico  $g$  e do erro de medida  $m$ , ou seja:

$$o = g + m \quad (4.6)$$

Se temos vários pontos, podemos calcular a variância de  $o$ :

$$\sigma^2(o) = \sigma^2(g + m) = \sigma^2(g) + \sigma^2(m) \quad (4.7)$$

onde  $\sigma^2(g)$  = erro geométrico,  $\sigma^2(m)$  = erro de medida e  $\sigma^2(o)$  é o erro medido ou observado, e onde se supõe que os erros são estatisticamente independentes. O que desejamos conhecer é

$\sigma^2(g)$ , que é a medida da qualidade geométrica da imagem. Assumindo que este erro geométrico varia lentamente sobre a imagem e que é constante dentro de uma pequena região, podemos estimar o erro de medida fazendo várias medidas em uma região. Se o erro geométrico é constante em tal região, então a sua varância é zero, ou seja:

se  $g = cte \implies \sigma^2(g) = 0$ , sendo assim  $\sigma^2(o) = \sigma^2(m)$ , dentro de cada região.

Para aplicar este método precisamos ter a imagem de referência e a imagem registrada. O procedimento é descrito a seguir:

1. Primeiramente selecionamos conjuntos de pontos em áreas pequenas (janelas) da imagem de referência ( $R$ ) e localizamos esses mesmos pontos na imagem registrada ( $C$ ). Seja  $r$  o número de regiões e  $n_i$  o número de pontos na região  $i$ . Seja  $(X_{ij}, Y_{ij})$  as coordenadas do ponto  $j$  na região  $i$  de  $R$  e  $(X'_{ij}, Y'_{ij})$  as coordenadas da imagem registrada. Uma janela (região) de  $R$  é correlacionada com uma área em  $C$ .

2. Considerando primeiro as coordenadas em  $X$ , calculamos:

$$o_{ij} = X_{ij} - X'_{ij} \quad (4.8)$$

$$\sigma^2(o_i) = \frac{\sum_{j=1}^{n_i} (o_{ij} - \bar{o}_i)^2}{n_i - 1} \quad (4.9)$$

onde  $\bar{o}_i$  é a média de  $o_{ij}$  na região  $i$ . O erro geométrico é assumido constante na região então  $\sigma^2(m_i) = \sigma^2(o_i)$ , com isto temos o valor do erro de medida.

3. Combinar as variâncias do erro de medida de cada região como:

$$\hat{\sigma}^2(m) = \frac{\sum_{i=1}^r (n_i - 1) \sigma^2(m_i)}{\sum_{i=1}^r (n_i - 1)} \quad (4.10)$$

o que dá uma estimativa global da variância devido ao processo de medida.

4. Num segundo passo localizar um conjunto de pontos isolados em ambas as imagens  $(X_s, Y_s)$  e  $(X'_s, Y'_s)$  sendo  $s = 1, 2, \dots, n_s$ . Calcular o erro e a variância do erro observado (medido):

$$o_s = X_s - X'_s \quad (4.11)$$

$$\sigma^2(o_s) = \frac{\sum_{s=1}^{n_s} (o_s - \hat{o}_s)^2}{n_s - 1} \quad (4.12)$$

5. Agora estimar o erro geométrico verdadeiro:

$$\sigma^2(g) = \sigma^2(o_s) - \hat{\sigma}^2(m) \quad (4.13)$$

este é o resultado que se deseja. O cálculo deve ser repetido para a outra coordenada  $Y$ , assim teremos o erro geométrico total como:

$$\sigma^2(g_{total}) = \sigma^2(g_X) + \sigma^2(g_Y) \quad (4.14)$$

A Figura 4.4 a seguir mostra as janelas nas quais são feitas várias medidas e num segundo passo a obtenção dos pontos isolados.

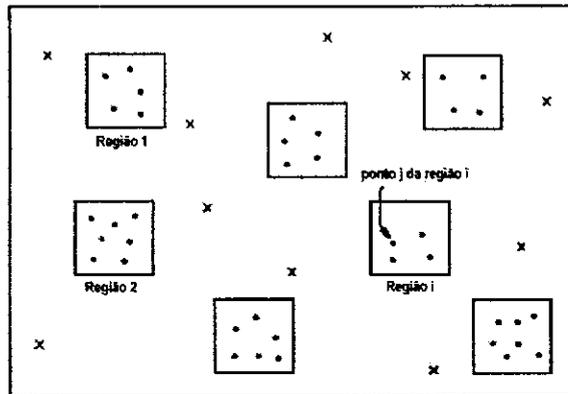


Fig. 4.4- Medida da precisão de registro. FONTE: [10] (p. 155).

Este método foi implementado e testado e os resultados obtidos estão mostrados no Capítulo 6. O algoritmo foi implementado em linguagem *C* usando o Borland 5.0 e rodou bem num microcomputador Pentium 133 Mhz. O tempo de processamento é em média de 4 minutos para cada imagem (512 x 544).

## CAPÍTULO 5

### REGISTRO AUTOMÁTICO: O ALGORITMO DE LI ET AL [4]

#### 5.1 Descrição do Algoritmo

Este algoritmo é baseado no método de feições. Ele extrai os contornos dos objetos presentes na imagem e os separa em dois grupos: contornos fechados e contornos abertos. No processo de casamento dos contornos fechados usa a correlação como medida de similaridade e também outros atributos extraídos dos contornos. Para o casamento dos contornos abertos usa pontos salientes ao longo dos contornos. A Figura 5.1 mostra o diagrama de blocos do algoritmo proposto.

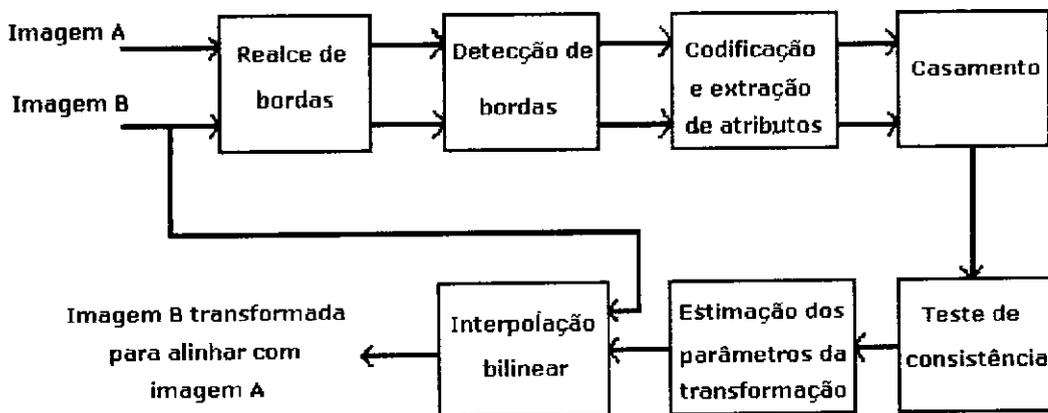


Fig. 5.1- Diagrama de blocos.

### 5.1.1 Realce de Bordas

Primeiramente as imagens passam por um processo de convolução através do operador Laplaciano da Gaussiana (LoG) e as bordas são detectadas nos pontos de cruzamento por zero (mudança de sinal) [21]. A Figura 5.2 mostra um exemplo de detecção de borda para uma função contínua. Notar que na segunda derivada obtemos o ponto de cruzamento por zero. As derivadas da função foram obtidas utilizando as seguintes equações:

$$\begin{aligned} I'(x) &= \frac{dI(x)}{dx} \\ I''(x) &= \frac{d^2 I(x)}{dx^2} \end{aligned} \quad (5.1)$$

A equação 5.2 mostra a convolução de uma função contínua com o LoG [22]:

$$I(x, y) * LoG(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(\alpha, \beta) LoG(x - \alpha, y - \beta) d\alpha d\beta \quad (5.2)$$

onde

$$LoG(x, y) = \nabla^2 G(x, y) = \frac{1}{2\pi\sigma^4} \left( 2 - \left( \frac{x^2 + y^2}{\sigma^2} \right) \right) \cdot e^{-(x^2 + y^2)/2\sigma^2}$$

e a Figura 5.3 mostra o gráfico da função LoG em uma dimensão e em duas dimensões (Figura 5.4). Para o caso de uma função digital o LoG tem que ser discretizado. A Figura 5.5 mostra

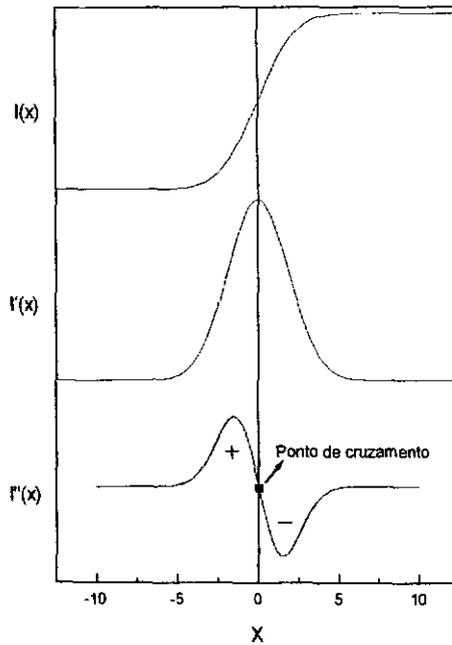


Fig. 5.2- Exemplo de detecção de borda.

um exemplo de detecção de borda para uma função digital obtida através da equação 5.3.

$$I(x, y) * LoG(x, y) = \sum_n \sum_m I(n, m) LoG(x - n, y - m) \quad (5.3)$$

A idéia de se usar um filtro LoG é primeiro suavizar a imagem com um filtro gaussiano e então encontrar as bordas da imagem suavizada. Estes filtros são frequentemente usados em imagens que possuem ruído como nas imagens digitais obtidas de satélites [10]. O LoG é um detector que consegue determinar as mudanças nos níveis de cinza de uma imagem. Estas mudanças correspondem aos contornos dos objetos que se deseja extrair. A idéia básica para o uso do LoG é que a gaussiana possui a característica de ser suave e localizada tanto no domínio da frequência quanto do espaço. Utilizando o LoG os cruzamentos serão detectados e localizados precisamente nos cruzamentos por zero do laplaciano [23]. A Figura 5.6 mostra como trabalha

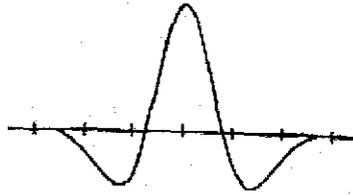


Fig. 5.3- LoG em uma dimensão. FONTE: [23] (p. 55).

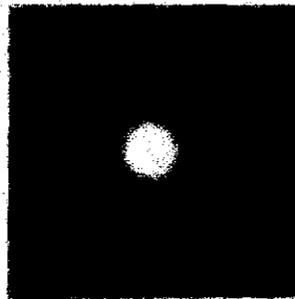


Fig. 5.4- LoG em duas dimensões. FONTE: [23] (p.55).

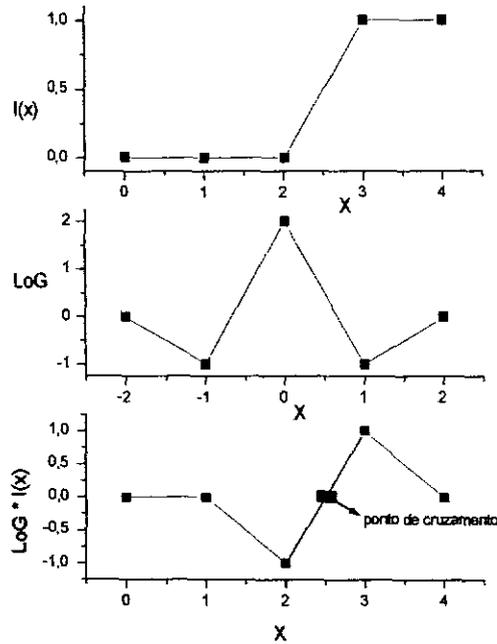


Fig. 5.5- Exemplo de detecção de borda.

um operador LoG em uma imagem.

### 5.1.2 Detecção de Bordas

Numa segunda etapa os cruzamentos são detectados ao longo das direções vertical e horizontal.

A intensidade de cada cruzamento  $S(x, y)$  é calculada considerando as inclinações do LoG da imagem nas direções  $x$  e  $y$  (indicados por  $k_x$  e  $k_y$ ) definido como:

$$S(x, y) = \begin{cases} \sqrt{k_x^2 + k_y^2} & \text{se } (x, y) \text{ é um ponto de cruzamento} \\ 0 & \text{caso contrário} \end{cases} \quad (5.4)$$

ou seja  $S(x, y)$  nada mais é do que a magnitude do gradiente. Seja a Figura 5.7 um exemplo de uma janela 3x3 centrada no ponto  $(x, y)$ , o gradiente  $\nabla f(x, y) = (k_x, k_y)$  é obtido usando as

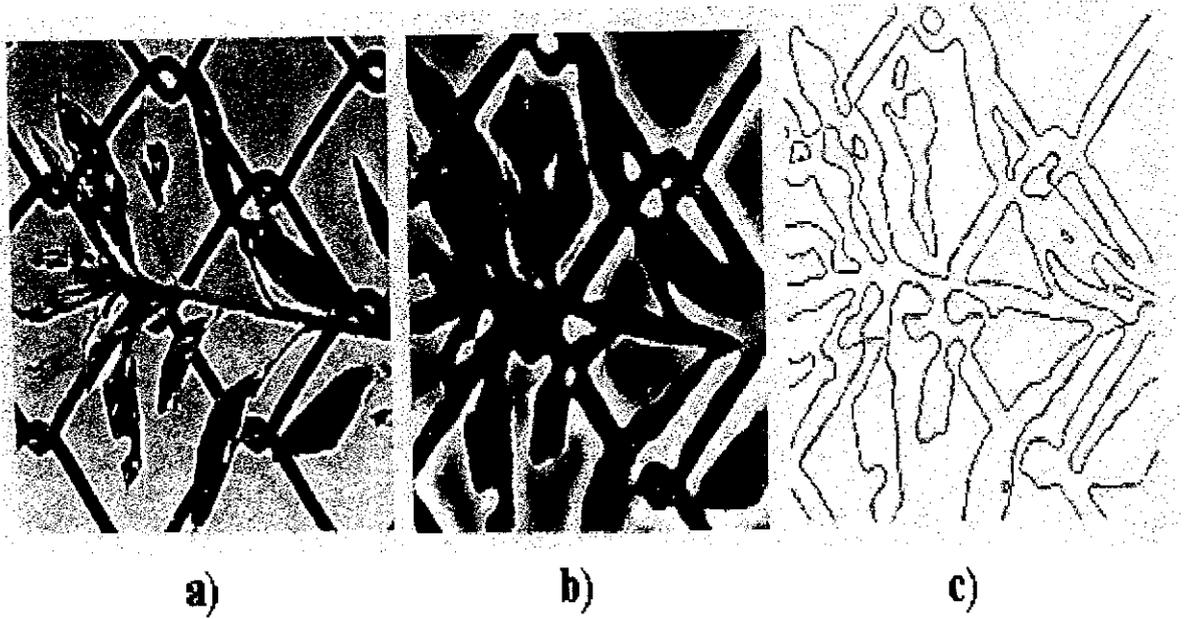


Fig. 5.6- Imagem (a), convolução com o LoG (b) e os cruzamentos (c). FONTE: [23] (p. 58).

$(x_{-1}, y_1)$	$(x, y_1)$	$(x_1, y_1)$
$(x_{-1}, y)$	$(x, y)$	$(x_1, y)$
$(x_{-1}, y_{-1})$	$(x, y_{-1})$	$(x_1, y_{-1})$

Fig. 5.7- Janela 3 x 3.

seguintes equações [24] :

$$k_x = \{f(x_1, y_1) + 2f(x_1, y) + f(x_1, y_{-1})\} - \{f(x_{-1}, y_1) + 2f(x_{-1}, y) + f(x_{-1}, y_{-1})\}$$

$$k_y = \{f(x_{-1}, y_1) + 2f(x, y_1) + f(x_1, y_1)\} - \{f(x_{-1}, y_{-1}) + 2f(x, y_{-1}) + f(x_1, y_{-1})\}$$

A direção e a magnitude do gradiente no ponto  $(x, y)$  são dados respectivamente por:

$$\nu(x, y) = \tan^{-1}(k_x/k_y),$$

$$S(x, y) = \sqrt{k_x^2 + k_y^2}$$

O algoritmo seleciona uma feição quando as condições abaixo são satisfeitas:

- [1]  $S(x, y)$  em cada ponto do contorno tem valor maior que  $T_1$ ;
- [2] Ao menos um ponto do contorno tem um valor  $S(x, y)$  maior do que  $T_2$ .

Os valores  $T_1$  e  $T_2$  são os limiares estabelecidos para extrair os contornos e  $T_1 < T_2$ . A pesquisa de contorno é iniciada quando um valor de limiar maior do que  $T_2$  é encontrado. Esta busca é conduzida em ambas direções da feição e os pixels com valores maiores do que  $T_1$  são aceitos como pontos do contorno até que esta condição não seja mais encontrada.

### 5.1.3 Codificação e Extração de Atributos

As feições são então divididas em duas categorias: contornos fechados e contornos abertos, e são representados por um código de cadeia [25] . Uma curva digital pode ser representada por

uma sequência inteira  $\{a_i \in (0, 1, 2, \dots, 7)\}$  dependendo da posição do pixel atual em relação ao pixel anterior. A Figura 5.8 mostra um exemplo de código de cadeia.

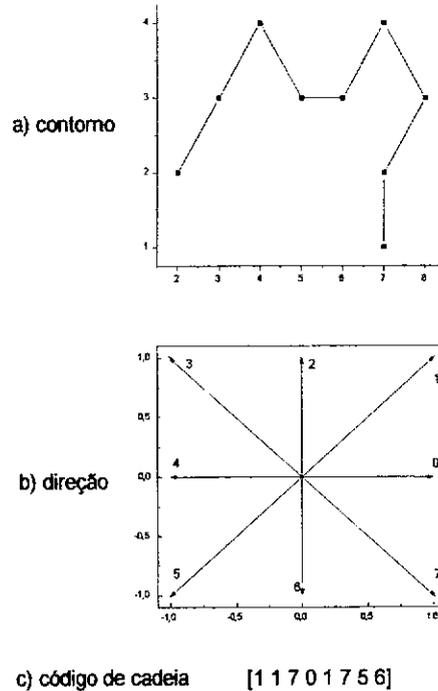


Fig. 5.8- Representação de um contorno em código de cadeia. FONTE: [24] (p. 119).

Para cada contorno são calculados cinco atributos: o perímetro, as distâncias maiores e menores da borda para o centróide, e o primeiro e o segundo momentos invariantes. O centróide de um contorno pode ser calculado pela seguinte equação:

$$x_c = \frac{1}{n} \sum_{i=1}^n x_i \quad y_c = \frac{1}{n} \sum_{i=1}^n y_i \quad (5.5)$$

onde  $x_i$  e  $y_i$  são coordenadas dos pontos do contorno [26]. As distâncias maiores ( $d_1$ ) e menores ( $d_2$ ) até o centróide podem ser obtidas encontrando o valor máximo e mínimo da seguinte

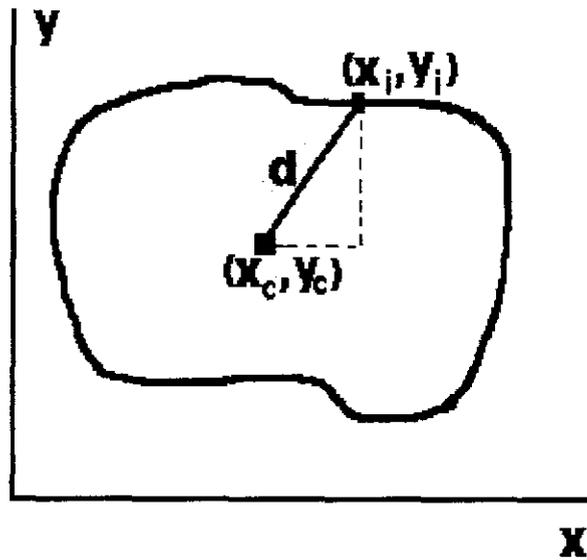


Fig. 5.9- Exemplo de um contorno.

equação:

$$d_1 = \max \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \tag{5.6}$$

$$d_2 = \min \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$$

onde  $(x_i, y_i)$  é um ponto qualquer do contorno como mostra a Figura 5.9.

O perímetro de uma região é o comprimento de seu contorno e pode ser determinado pela seguinte equação:

$$p = n_H \times H + n_V \times V + n_d \sqrt{H^2 + V^2}$$

onde:  $n_H$  é o número de componentes na horizontal,  $n_V$  é o número de componentes na vertical,  $n_d$  é o número de componentes na diagonal,  $H$  e  $V$  são os espaçamentos (resolução) na horizontal

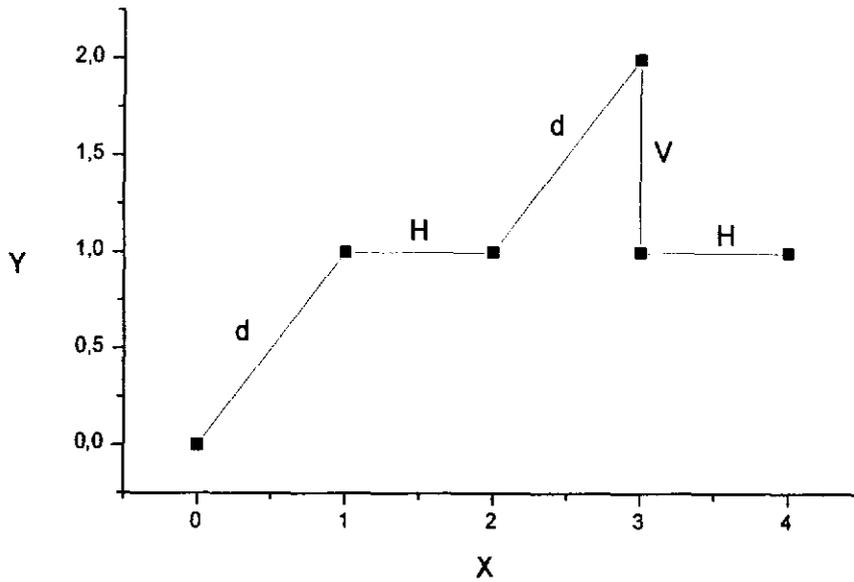


Fig. 5.10- Exemplo de cálculo de perímetro.

e vertical respectivamente. Como exemplo, se tivermos um contorno com espaçamento unitário nas direções horizontal e vertical ( $H = V = 1$ ), o seu comprimento será dado pelo número de componentes horizontais e verticais mais a soma dos componentes diagonais como mostra a Figura 5.10, onde  $p = 2H + V + 2d = 2 + 1 + 2\sqrt{2} = 5.8$ . O perímetro ainda pode ser aproximado pelo número de pontos do contorno se fizermos ( $H = V = d = 1$ )[16,22].

O primeiro e o segundo momentos invariantes são definidos do seguinte modo [22, 27]:

$$h_1 = \eta_{2,0} + \eta_{0,2}$$

$$h_2 = (\eta_{2,0} - \eta_{0,2})^2 + 4\eta_{1,1}^2$$

$$\text{onde } \eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^\alpha} \quad \text{com } \alpha = \frac{p+q}{2} + 1 \quad \text{e } \mu_{p,q} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} (n - \bar{x})^p (m - \bar{y})^q$$

para o caso de contornos estas equações se tornam:

$$\begin{aligned}
 h_1 &= \frac{1}{n^2} \sum_{i=1}^n [(x_i - x_c)^2 + (y_i - y_c)^2] \\
 h_2 &= \frac{1}{n^4} \left[ \sum_{i=1}^n (x_i - x_c)^2 - \sum_{i=1}^n (y_i - y_c)^2 \right]^2 + \frac{4}{n^4} \left[ \sum_{i=1}^n (x_i - x_c) + (y_i - y_c) \right]^2
 \end{aligned} \tag{5.7}$$

Estes momentos são chamados de invariantes porque não dependem da posição, tamanho e orientação, e foram definidos originalmente para imagens em 2D [28].

#### 5.1.4 Casamento

O processo de casamento começa com os contornos fechados. Cada contorno fechado na primeira imagem é comparado com cada contorno fechado na segunda imagem. Os pares são aceitos como candidatos ao casamento se a diferença relativa para cada atributo estiver abaixo de um limiar especificado. Neste trabalho adotou-se os limiares recomendados por Li [18]. Um contorno  $A$  na primeira imagem e um contorno  $B$  na segunda imagem serão considerados casados se:

$C_{AB} \geq C_{AB'}$ , onde  $B'$  inclui todos os contornos com formatos similares ao contorno  $A$  e

$C_{AB} > T_3$ , onde  $T_3$  é um valor de limiar que elimina casamentos com correlação fraca.

onde  $C_{AB}$  é a função de similaridade, definida como:

$$C_{AB} = \max\{D_{kl}\}_{l \in M}$$

onde  $M$  especifica a extensão da pesquisa e  $D_{kl}$  é a medida de correlação do código de cadeia normalizado entre dois segmentos de  $n$  pontos ( um iniciando no índice  $k$  do contorno  $A$  e o outro começando no índice  $l$  do contorno  $B$ ,  $N_A$  é o número de pontos do contorno  $A$  e  $N_B$  é o número de pontos do contorno  $B$ ), definida como:

$$D_{kl} = \frac{1}{n} \sum_{j=0}^{n-1} \cos \frac{\pi}{4} (a'_{k+j} - b'_{l+j})$$

onde

$$a'_{k+i} = a_{(k+i) \bmod N_A} - \frac{1}{n} \sum_{j=0}^{n-1} a_{(k+j) \bmod N_A}, 0 \leq i < n,$$

$$b'_{l+i} = a_{(l+i) \bmod N_B} - \frac{1}{n} \sum_{j=0}^{n-1} b_{(l+j) \bmod N_B}, 0 \leq i < n.$$

A função mod acomoda os casos para os contornos fechados pois o índice  $k$  pode começar em qualquer parte de um contorno como mostra a Figura 5.11. Se  $k$  começa no final do contorno (  $a_5$  ) a função mod faz com que o próximo ponto a ser visitado seja o  $a_0$ . A função cosseno assegura que  $D_{kl} \leq 1$ , quando  $D_{kl} = 1$  o casamento é perfeito e  $D_{kl}$  é normalizada com respeito ao valor médio.

Os centróides destes contornos fechados são tomados como pontos de controle e são usados para calcular uma estimativa inicial dos parâmetros da transformação, que servirão como guia para o casamento dos contornos abertos. Para o casamento dos contornos abertos são usados segmentos salientes ao longo dos contornos, tal como cantos (esquinas), que podem ser detectados na representação do código de cadeia.

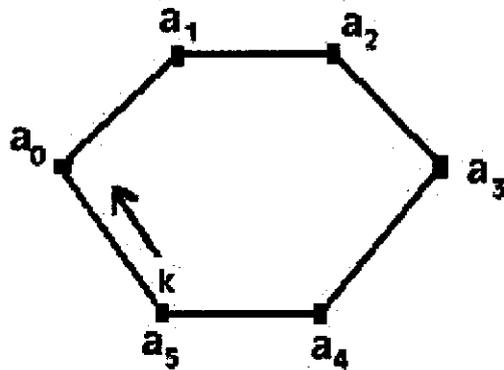


Fig. 5.11- Exemplo da função mod em um contorno.

No domínio contínuo a curvatura num dado ponto é comumente definida como a taxa de variação da tangente à curva naquele ponto. Seja  $\{x(s), y(s)\}$  a representação paramétrica de uma curva, o ângulo tangente será:

$$\phi(s) = \tan^{-1} \left( \frac{\frac{dy(s)}{ds}}{\frac{dx(s)}{ds}} \right)$$

A curvatura definida como  $k(s) = \frac{d\phi}{ds}$  será dada por [27] :

$$k(s) = \frac{dx}{ds} \cdot \frac{d^2y}{ds^2} - \frac{d^2x}{ds^2} \cdot \frac{dy}{ds}$$

No caso discreto a curvatura pode ser definida, num ponto  $P$ , como a diferença entre os ângulos correspondentes aos códigos de cadeia adjacentes a  $P$  [16, 22] . No caso deste algoritmo define-

se uma medida de curvatura no ponto  $i$  dada por:

$$c_i = \max_{1 \leq j \leq 3\sigma} \{ \max\{|a_{i-j} - a_{i+j}|, |a_{i-j} - a_{i+j-1}|\} \}$$

onde  $\sigma$  é o desvio padrão do operador LoG usado para extrair os contornos. O parâmetro  $\sigma$  controla a quantidade de filtragem. Ele determina a escala na qual a operação é executada[27].

O ponto  $i$  será considerado um ponto saliente (ponto de controle) se:

- $c_i$  for maior ou igual a um valor de limiar especificado,
- $c_i \geq c_k$  para todo  $k \in [i - p, i + p]$ , onde  $p$  é uma constante que determina a distância mínima entre pontos salientes.

### 5.1.5 Estimação dos Parâmetros da Função de Transformação

O relacionamento entre um ponto  $(X, Y)$  de uma imagem e o correspondente ponto  $(\hat{X}, \hat{Y})$  da outra imagem pode ser expresso por uma transformação de similaridade da seguinte forma:

$$\begin{pmatrix} \hat{X} \\ \hat{Y} \end{pmatrix} = r \begin{pmatrix} \cos \theta & \text{sen} \theta \\ -\text{sen} \theta & \cos \theta \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \quad (5.8)$$

$$\hat{X} = r(X \cos \theta + Y \text{sen} \theta) + \Delta x$$

$$\hat{Y} = r(-X \text{sen} \theta + Y \cos \theta) + \Delta y$$

onde:  $r$  corresponde ao fator de escala,  $\theta$  é o ângulo de rotação, e  $\Delta x$  e  $\Delta y$  são as translações ao longo das 2 direções ortogonais ( $\Delta x = e, \Delta y = f$ , ver equação 3.6 do Capítulo 3).

Os parâmetros da transformação são estimados de um conjunto de pontos casados  $\{(X_i, Y_i), (\hat{X}_i, \hat{Y}_i), i = 1, \dots, n\}$ . Um método de resolver a equação 5.8, adotado por este algoritmo é fazendo as seguintes substituições:

$$u = r \cos \theta$$

$$v = r \sin \theta$$

obtendo assim a equação:

$$\begin{pmatrix} \hat{X} \\ \hat{Y} \end{pmatrix} = \begin{pmatrix} u & v \\ -v & u \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \quad (5.9)$$

Os parâmetros  $u, v, \Delta x$  e  $\Delta y$  podem ser calculados pelo método dos mínimos quadrados, baseado em todos os pontos casados. Sejam  $(X_m, Y_m), m = 1, 2, \dots, n$  as coordenadas de  $n$  pontos de controle da imagem de referência (saída) e  $(\hat{X}_m, \hat{Y}_m), m = 1, 2, \dots, n$  as coordenadas da imagem de entrada. Seja o mapeamento de um ponto  $(X, Y)$  da imagem de saída em um ponto  $(\hat{X}, \hat{Y})$  da imagem de entrada dado por (ver Figura 5.12):

$$Q(X, Y) = q_0 + q_1 X + q_2 Y + \dots$$

$$R(X, Y) = r_0 + r_1 X + r_2 Y + \dots$$

então  $\hat{X} = Q(X, Y)$  e  $\hat{Y} = R(X, Y)$ . Os coeficientes do polinômio podem ser calculados pelo método dos mínimos quadrados, onde  $\mathbf{Q}$  é o vetor dos coeficientes do polinômio,  $\mathbf{K}$  é o vetor das coordenadas de entrada  $\hat{X}$  e  $\mathbf{M}$  é a matriz de termos do polinômio formado das coordenadas

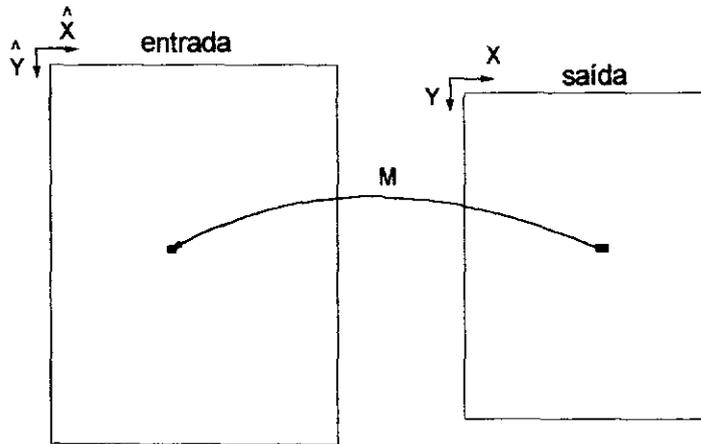


Fig. 5.12- Mapeamento de um ponto  $(X, Y)$  em um ponto  $(\hat{X}, \hat{Y})$ .

$(X, Y)$ , ou seja:

$$\mathbf{Q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} \quad \mathbf{K} = \begin{bmatrix} k_1 \\ k_2 \\ \cdot \\ \cdot \\ \cdot \\ k_n \end{bmatrix} \quad \mathbf{M} = \begin{bmatrix} 1 & X_1 & Y_1 & X_1^2 & X_1 Y_1 & Y_1^2 & \cdots \\ 1 & X_2 & Y_2 & X_2^2 & X_2 Y_2 & Y_2^2 & \cdots \\ 1 & X_3 & Y_3 & X_3^2 & X_3 Y_3 & Y_3^2 & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots \\ 1 & X_n & Y_n & X_n^2 & X_n Y_n & Y_n^2 & \cdots \end{bmatrix}$$

e  $\mathbf{MQ} = \mathbf{K}$ . A solução dos mínimos quadrados é dada por:

$$\mathbf{M}^T \mathbf{M} \mathbf{Q} = \mathbf{M}^T \mathbf{K}$$

$$\mathbf{Q} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{K}$$

Um cálculo similar deve ser feito para a outra coordenada substituindo o vetor  $\mathbf{K}$  pelas coordenadas de entrada  $\hat{Y}$ , assim obtendo o valor de  $\mathbf{R}$  [10, 29].

Os parâmetros de escala  $r$  e de rotação  $\theta$  podem ser determinados de  $u$  e  $v$ , ou seja:

$$r = \sqrt{u^2 + v^2}$$

$$\theta = \arctan\left(\frac{u}{v}\right) \text{ para } -\frac{\pi}{2} < \theta < \frac{\pi}{2}$$

Transformações mais gerais tal como a polinomial ( de grau superior a 1) podem ser mais adequadas se as imagens possuem distorção não-linear.

A medida da precisão do registro entre os pontos casados depois da transformação pode ser calculada através do erro RMSE ( *Root Mean Square Error* ) e é definida como:

$$RMSE = \left( \frac{\sum_{i=1}^m \left[ \left( uX_i + vY_i + \Delta x - \hat{X}_i \right)^2 + \left( uY_i - vX_i + \Delta y - \hat{Y}_i \right)^2 \right]}{m} \right)^{1/2} \quad (5.10)$$

onde  $m$  é o número dos pontos casados. É importante mencionar que  $m$  deve ser razoavelmente grande para tornar o registro significativo ( $m \geq 2$ ).

### 5.1.6 Teste de Consistência

No processo de casamento alguns falsos casamentos são inevitáveis. Um teste de consistência é necessário para assegurar um registro correto. Seja  $(\overline{A_1 A_2})$  a distância entre os pontos  $A_1$  e  $A_2$ . Para 2 conjuntos de  $m$  pontos casados  $\{A_i\}$  e  $\{B_i\}$  as taxas de  $\overline{A_i A_j} / \overline{B_i B_j}$  são calculadas para todas as possíveis combinações  $m(m-1)/2$ . O histograma resultante de escala deveria formar um *cluster* correspondendo a diferença de escalas entre as imagens. Os pares que contribuem com o *cluster* serão aceitos como casamentos corretos, caso contrário, serão descartados.

Este algoritmo foi processado numa estação Sun e o tempo de processamento está em torno de 40 segundos para cada imagem (tamanho da imagem 512 x 544). Os resultados obtidos estão mostrados no Capítulo 6.

## CAPÍTULO 6

### RESULTADOS OBTIDOS

#### 6.1 Registro usando o Método Manual

Foram realizados quatro registros de imagens. As imagens utilizadas são do sensor TM (*Landsat*) e tomadas em tempos diferentes (imagens temporais), mas da mesma área. Foram adquiridas através do INPE-ATUS (Atendimento ao Usuário) via CD-ROM.

Base e Ponto se referem a posição cartográfica no mapa. Bandas são as bandas espectrais obtidas do sensor TM que estão gravadas no CD-ROM.

**Primeira Imagem:** Vale-Paraíba, Serra e Litoral Norte

Base: 218; Ponto: 076; Bandas: 1,2,3,4,5,7; Data: 25/07/96

**Segunda Imagem:** Vale-Paraíba, Serra e Litoral Norte

Base: 218; Ponto: 076; Bandas: 1,2,3,4,5,6,7; Data: 26/06/97

**Terceira Imagem:** Mato Grosso

Base: 224; Ponto: 74A; Bandas: 1,2,3,4,5,6,7 Data: 18/08/95

**Quarta Imagem:** Mato Grosso

Base: 224; Ponto: 74A; Bandas: 1,2,3,4,5,6,7 Data: 06/07/97

O nível de correção geométrica das imagens é 4, indicando que neste nível nenhuma correção de sistema é realizada sobre a imagem. Tamanho de todas as imagens: 512 x 544

### 6.1.1 Primeiro Registro: Caraguatatuba

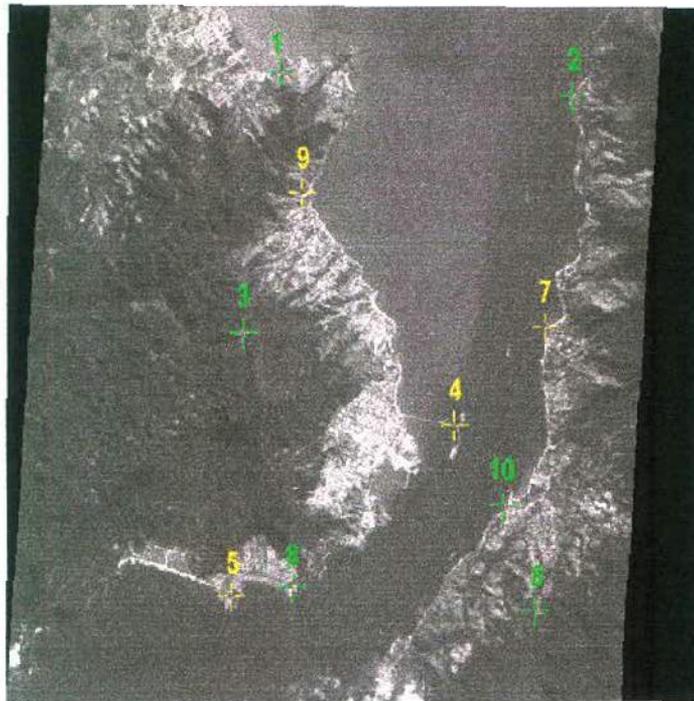


Fig. 6.1- Caraguá1.

Imagem de referência → Caraguá1.grb - (25/07/96)

Imagem de ajuste → Caraguá2.grb - (26/06/97)

As Figuras 6.1 e 6.2 mostram essas imagens.

Em todas as imagens foram adquiridos 10 pontos de controle, dos quais 6 pontos serão usados

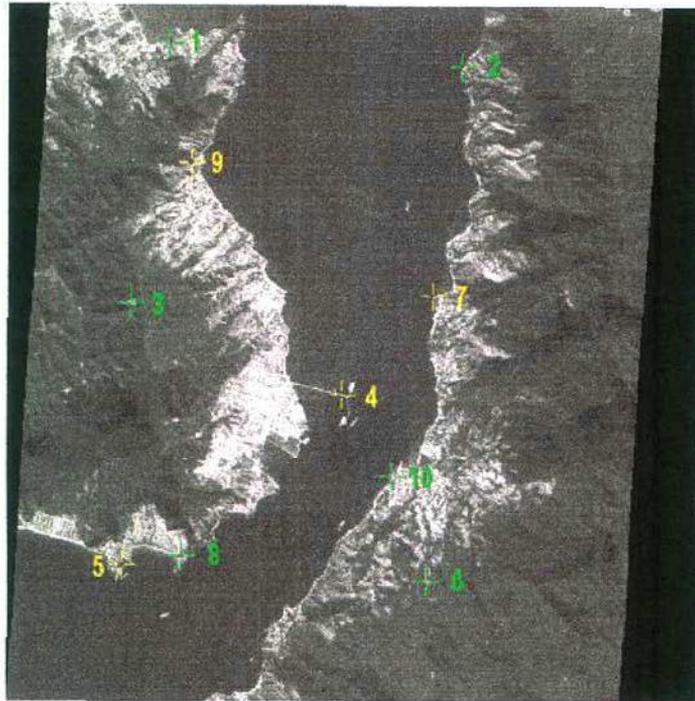


Fig. 6.2- Caraguá2.

pela equação de mapeamento para fazer o registro, e os outros 4 pontos serão usados como Pontos de Teste para poder calcular o erro de registro. A Tabela 6.2 mostra os pontos de controle selecionados para registro (cor verde na imagem) e os pontos escolhidos como pontos de teste (cor amarelo na imagem) para todas as imagens. A Figura 6.3 mostra a imagem registrada.

### 6.1.2 Segundo Registro: Pindamonhangaba

Imagem de referência → Pinda1.grb - (25/07/96)

Imagem de ajuste → Pinda2.grb - (26/06/97)

As Figuras 6.4 e 6.5 mostram essas imagens. A Figura 6.6 mostra a imagem registrada.

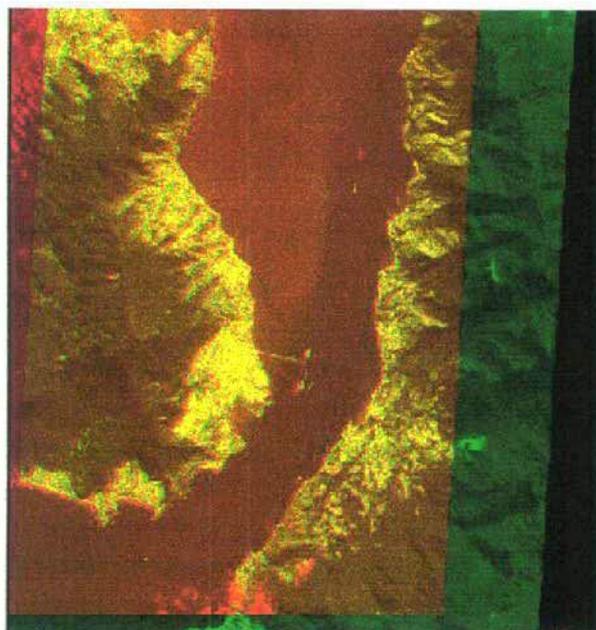


Fig. 6.3- Imagem registrada - caragua - banda3.

TABELA 6.2: PONTOS DE CONTROLE E PONTOS DE TESTE

<b>Imagem</b>	<b>PC (verde)</b>	<b>PT (amarelo)</b>
Caraguá	1, 2, 3, 6, 8, 10	4, 5, 9, 7
Pinda	1, 3, 4, 6, 9, 10	2, 5, 7, 8
Natividade	2, 3, 6, 8, 9, 10	1, 4, 5, 7
Mato	1, 2, 4, 5, 9, 10	3, 6, 7, 8

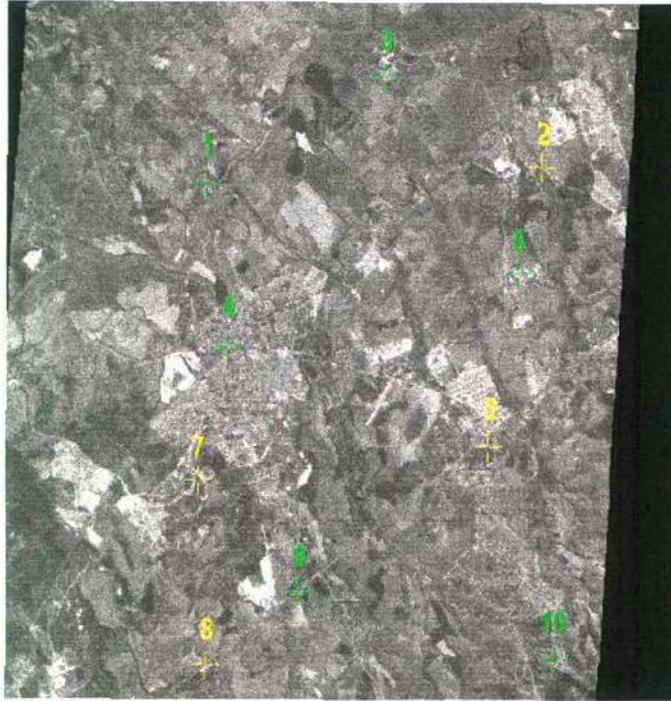


Fig. 6.4- Pinda1.

### 6.1.3 Terceiro Registro: Natividade

Imagem de referência → Natividade1.grb - (25/07/96)

Imagem de ajuste → Natividade2.grb - (26/06/97)

As Figuras 6.7 e 6.8 mostram essas imagens. A Figura 6.9 mostra a imagem registrada.

### 6.1.4 Quarto Registro: Mato Grosso

Imagem de referência → Mato1.grb - (18/08/95)

Imagem de ajuste → Mato2.grb - (06/07/97)

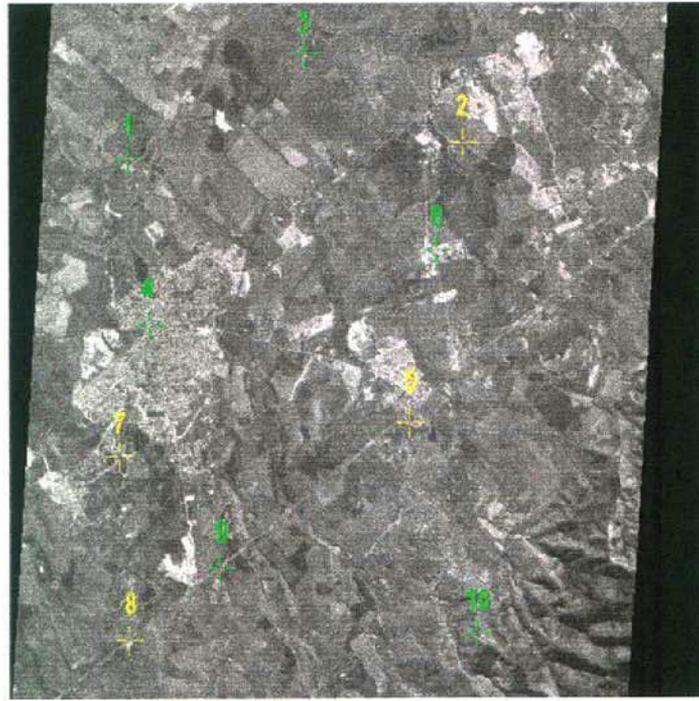


Fig. 6.5- Pinda2.

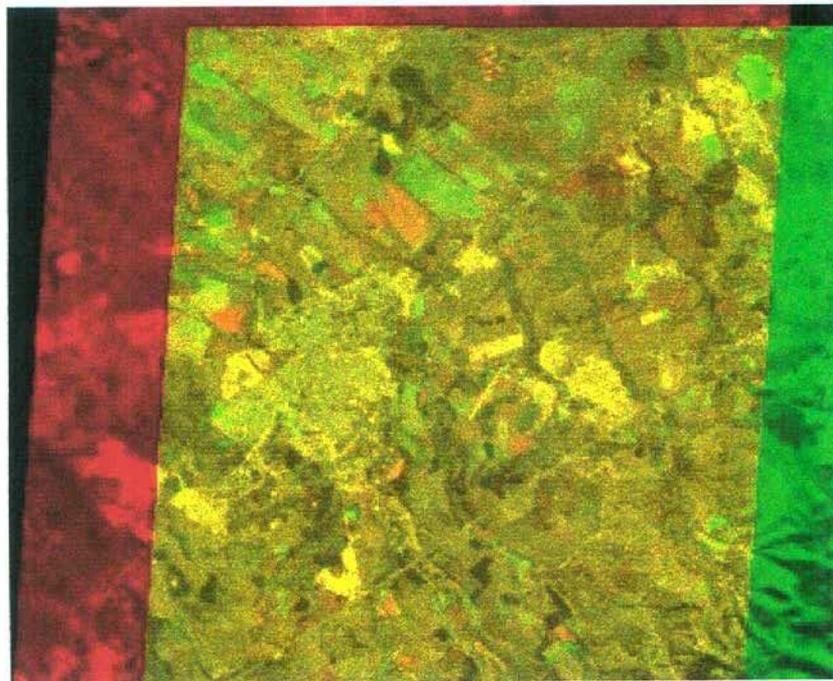


Fig. 6.6- Imagem registrada - Pinda - banda 3

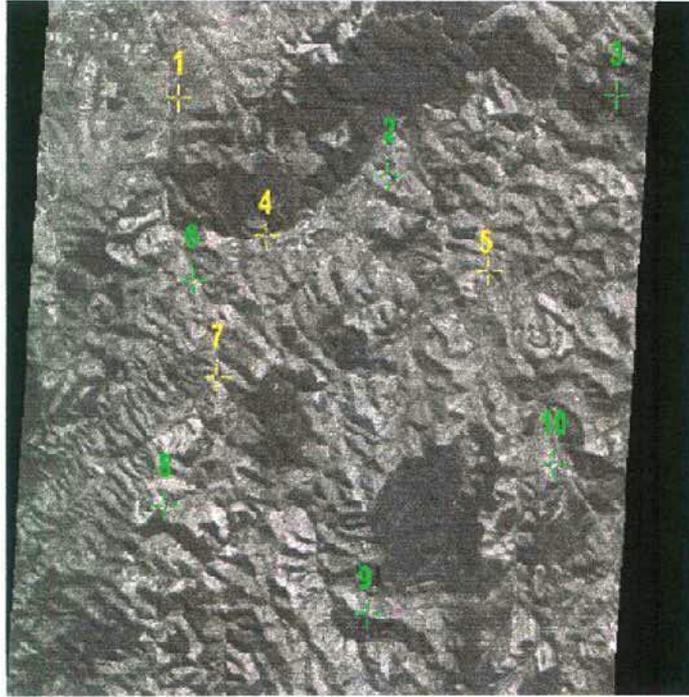


Fig. 6.7- Natividade1.

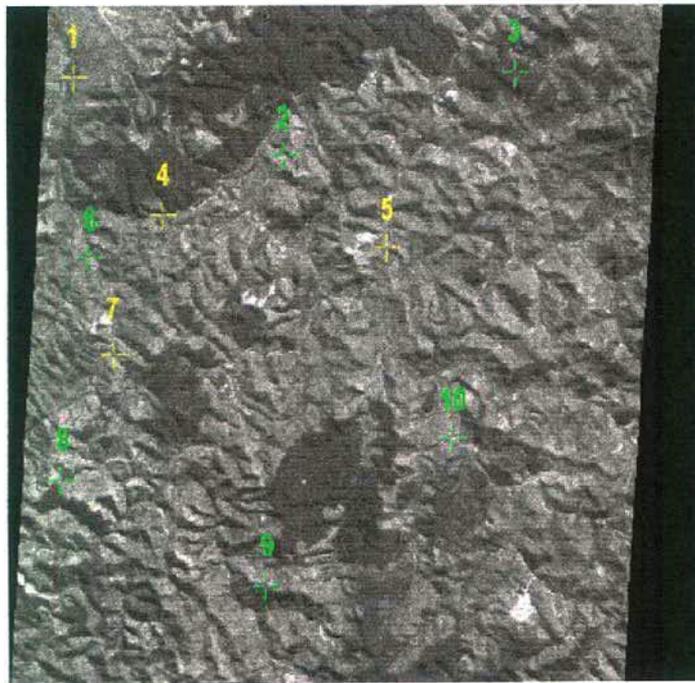


Fig. 6.8- Natividade2.

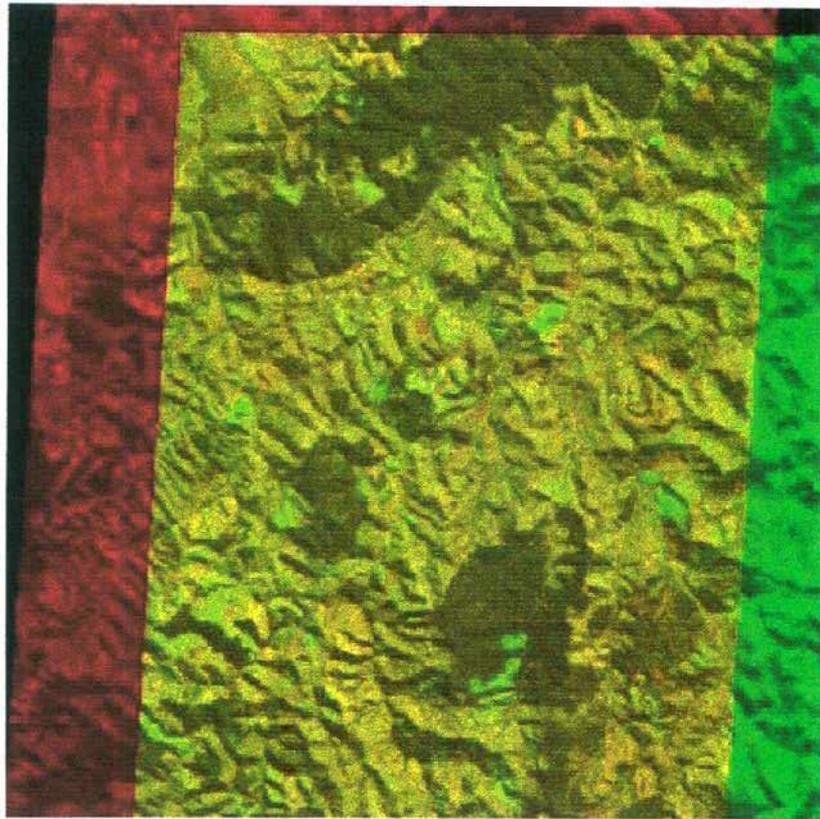


Fig. 6.9- Imagem registrada - Natividade - banda 3

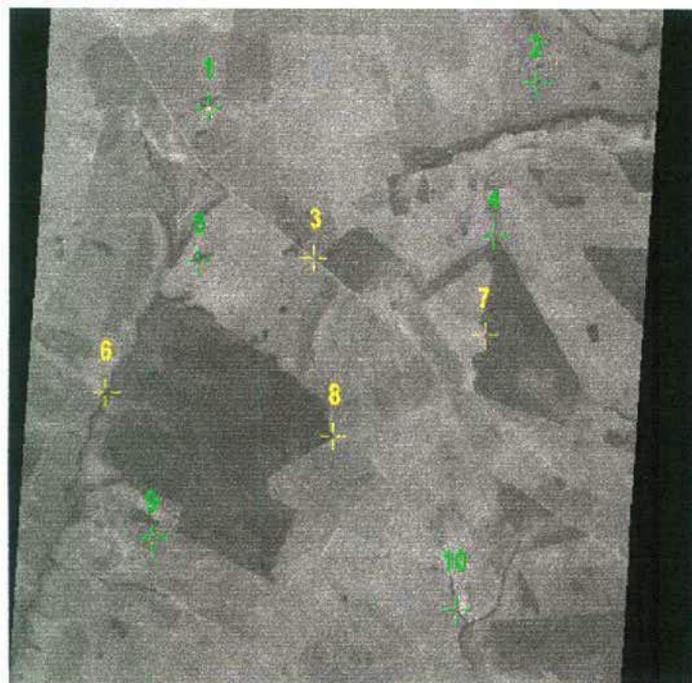


Fig. 6.10- Matol.

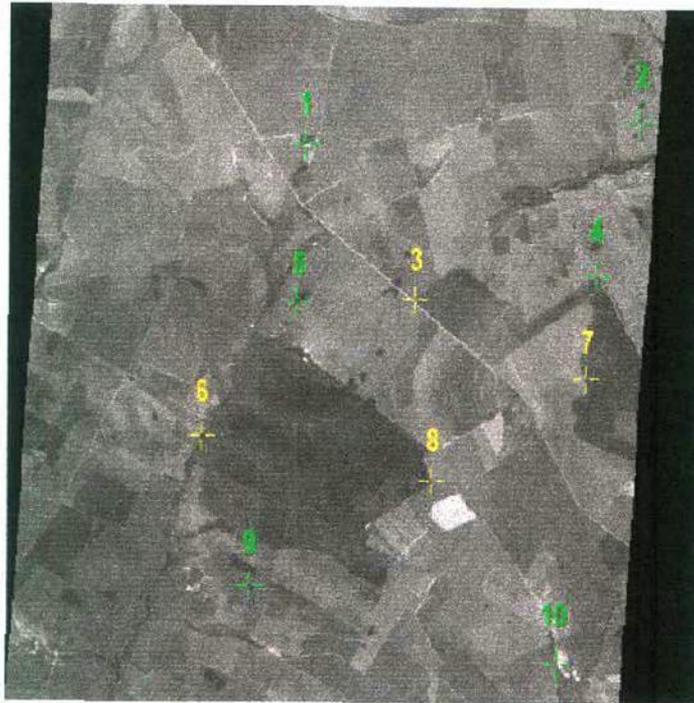


Fig. 6.11- Mato2.

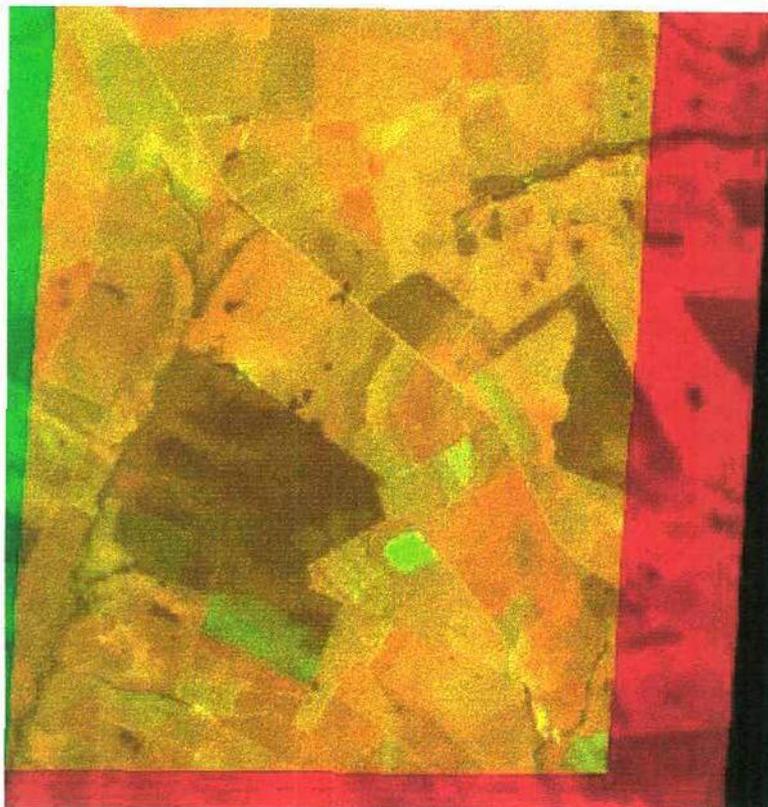


Fig. 6.12- Imagem registrada - Mato - banda 3

TABELA 6.3: COORDENADAS DOS PONTOS DE TESTE

<b>Imagem</b>	<b>Referência</b>	<b>Ajuste</b>
Caraguá	Pot4(286, 263)	Pot4(307, 350)
	Pot5(411, 89)	Pot5(432, 175)
	Pot7(178, 356)	Pot7(198, 442)
	Pot9(129, 147)	Pot9(149, 233)
Pinda	Pot2(120, 436)	Pot2(103, 352)
	Pot5(325, 395)	Pot5(308, 311)
	Pot7(349, 168)	Pot7(331, 84)
	Pot8(485, 174)	Pot8(467, 90)
Natividade	Pot1(71, 135)	Pot1(53, 51)
	Pot4(173, 205)	Pot4(155, 120)
	Pot5(198, 381)	Pot5(180, 296)
	Pot7(278, 166)	Pot7(259, 81)
Mato	Pot3(187, 239)	Pot3(214, 317)
	Pot6(288, 75)	Pot6(315, 152)
	Pot7(245, 374)	Pot7(272, 452)
	Pot8(320, 254)	Pot8(347, 332)

As Figuras 6.10 e 6.11 mostram essas imagens. A Figura 6.12 mostra a imagem registrada.

A Tabela 6.3 mostra as coordenadas dos pontos de teste da imagem de referência e da imagem de ajuste obtidas visualmente através do SPRING.

A Tabela 6.4 mostra os resultados dos parâmetros gerados pela transformação para o método manual (escala, rotação e translações).

As Tabelas 6.5, 6.6, 6.7 e 6.8 mostram as coordenadas dos Pontos de Teste calculadas através da equação 4.4 do Capítulo 4 para as imagens de Caraguá, Pinda, Natividade e Mato, respectivamente.

TABELA 6.4: PARÂMETROS DA TRANSFORMAÇÃO-MÉTODO MANUAL

<b>Imagem</b>	<b>k</b>	<b><math>\Delta\theta</math></b>	<b><math>\Delta x</math></b>	<b><math>\Delta y</math></b>
Caraguá	0.999719	0.001660	-21.273994	-85.973267
Pinda	1.001966	0.001767	16.646309	84.493774
Natividade	1.002223	-0.000263	18.181717	84.134521
Mato	0.997807	0.002390	-27.592312	-76.325684

TABELA 6.5: CARAGUÁ - PONTOS APROXIMADOS - MÉTODO MANUAL

<b>Imagem</b>	<b>Pot4(x,y)</b>	<b>Pot5(x,y)</b>	<b>Pot7(x,y)</b>	<b>Pot9(x,y)</b>
Caraguá	(286, 264)	(411, 90)	(177, 356)	(128, 147)

TABELA 6.6: PINDA - PONTOS APROXIMADOS - MÉTODO MANUAL

<b>Imagem</b>	<b>Pot2(x,y)</b>	<b>Pot5(x,y)</b>	<b>Pot7(x,y)</b>	<b>Pot8(x,y)</b>
Pinda	(120, 437)	(325, 396)	(348, 169)	(485, 175)

TABELA 6.7: NATIVIDADE - PONTOS APROXIMADOS - MÉTODO MANUAL

<b>Imagem</b>	<b>Pot1(x,y)</b>	<b>Pot4(x,y)</b>	<b>Pot5(x,y)</b>	<b>Pot7(x,y)</b>
Natividade	(71, 135)	(174, 204)	(199, 381)	(278, 165)

TABELA 6.8: MATO - PONTOS APROXIMADOS - MÉTODO MANUAL

<b>Imagem</b>	<b>Pot3(x,y)</b>	<b>Pot6(x,y)</b>	<b>Pot7(x,y)</b>	<b>Pot8(x,y)</b>
Mato	(186, 240)	(287, 75)	(244, 375)	(319, 255)

TABELA 6.9: NÚMERO DE CONTORNOS CASADOS E VALOR DOS LIMIARES

Imagem	Banda	Contornos casados	Limiares
Caraguá	3	3	25 – 4 – 10
	5	3	50 – 5 – 30
	7	5	20 – 0 – 10
Pinda	3	3	15 – 0 – 10
	4	2	50 – 5 – 30
	5	5	30 – 0 – 10
	7	–	–
Natividade	3	14	30 – 5 – 10
	4	18	50 – 5 – 30
	5	13	50 – 5 – 30
	7	10	20 – 0 – 10
Mato	3	–	–
	4	–	–
	5	3	30 – 5 – 10
	7	2	50 – 5 – 10

## 6.2 Registro usando o Método de Feições

Foram realizados quatro registros das mesmas imagens utilizadas no Método Manual. Este algoritmo fornece não só o erro encontrado pela equação de mapeamento e os parâmetros da transformação, mas também os pontos de controle (feições) encontradas. Em todos os registros não foram encontrados contornos abertos, somente contornos fechados.

### 6.2.1 Primeiro Registro: Caraguatatuba

Imagem de referência → Caraguá2.raw - (25/07/96)

Imagem de ajuste → Caraguá1.raw - (26/06/97)

A Tabela 6.9 mostra o número de contornos casados encontrados e o valor dos limiares usados

para todas as imagens.

As Figuras 6.13, 6.14 e 6.15 mostram os contornos casados encontrados das bandas 3, 5 e 7 respectivamente.

As Figuras 6.16, 6.17 e 6.18 mostram os mosaicos das imagens casadas das bandas 3, 5 e 7, respectivamente.

### **6.2.2 Segundo Registro: Pindamonhangaba**

Imagem de referência → Pinda1.raw - (25/07/96)

Imagem de ajuste → Pinda2.raw - (26/06/97)

#### **Banda 7:**

Nesta banda o algoritmo não conseguiu encontrar os contornos, sendo assim não foi possível realizar o registro.

As Figuras 6.19 e 6.20 mostram as feições casadas da banda 3 e o mosaico da imagem obtida após o registro.

### **6.2.3 Terceiro Registro: Natividade**

Imagem de referência → Natividade1.raw - (25/07/96)

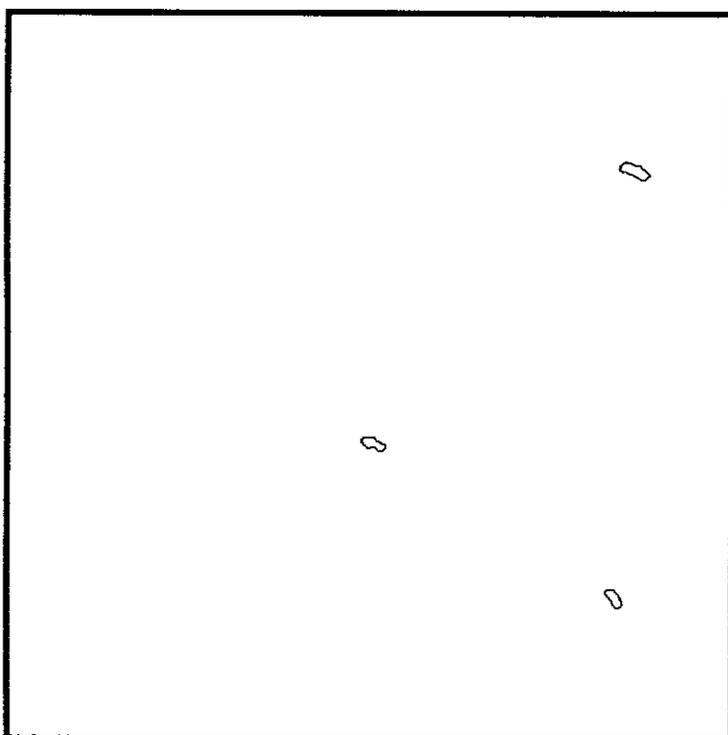


Fig. 6.13- Caraguá - feições encontradas - banda3.

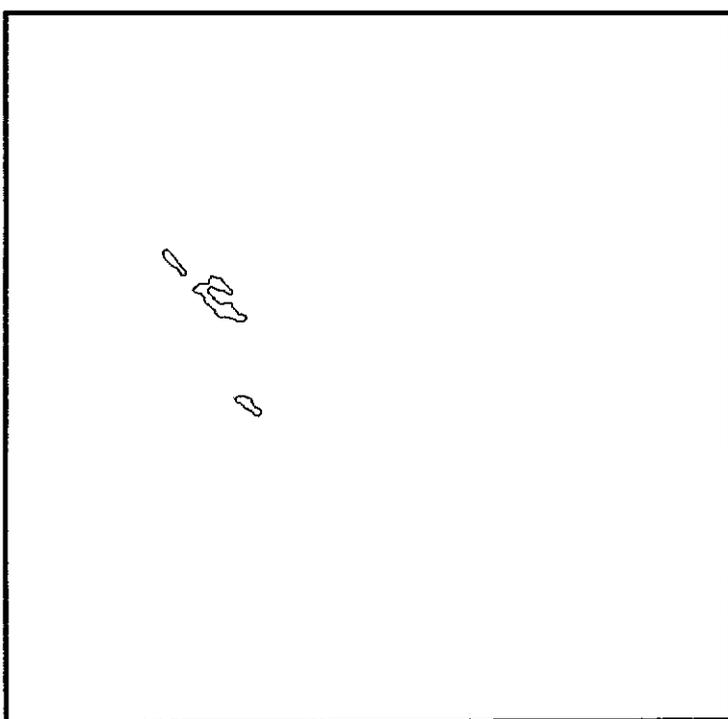


Fig. 6.14- Caraguá - feições encontradas - banda5.

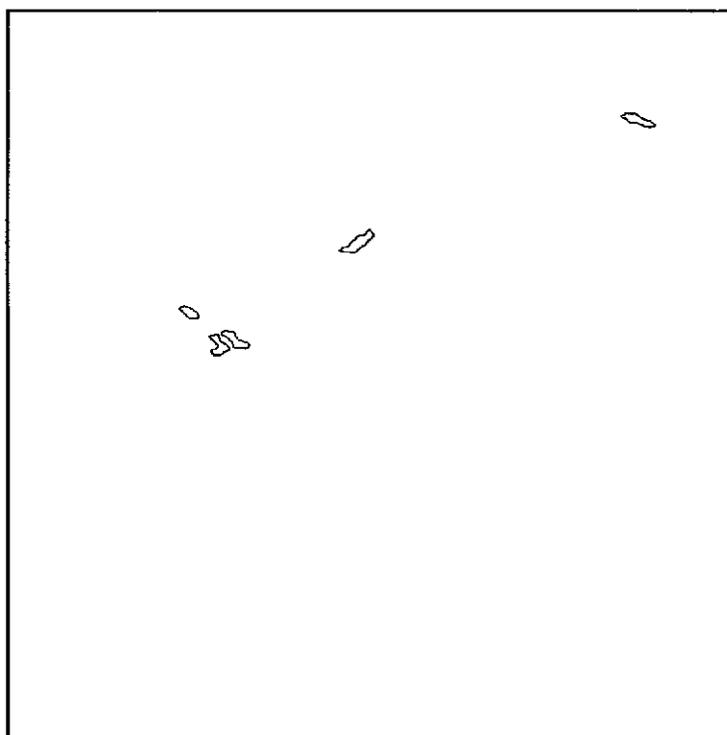


Fig. 6.15- Caraguá - feições encontradas - banda7.

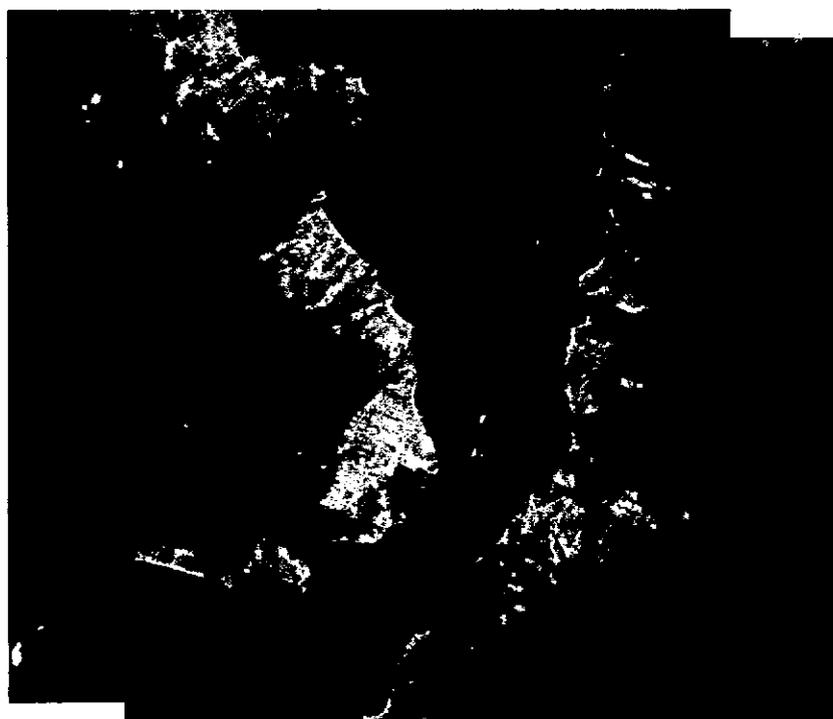


Fig. 6.16- Mosaico - Caraguá - banda3.

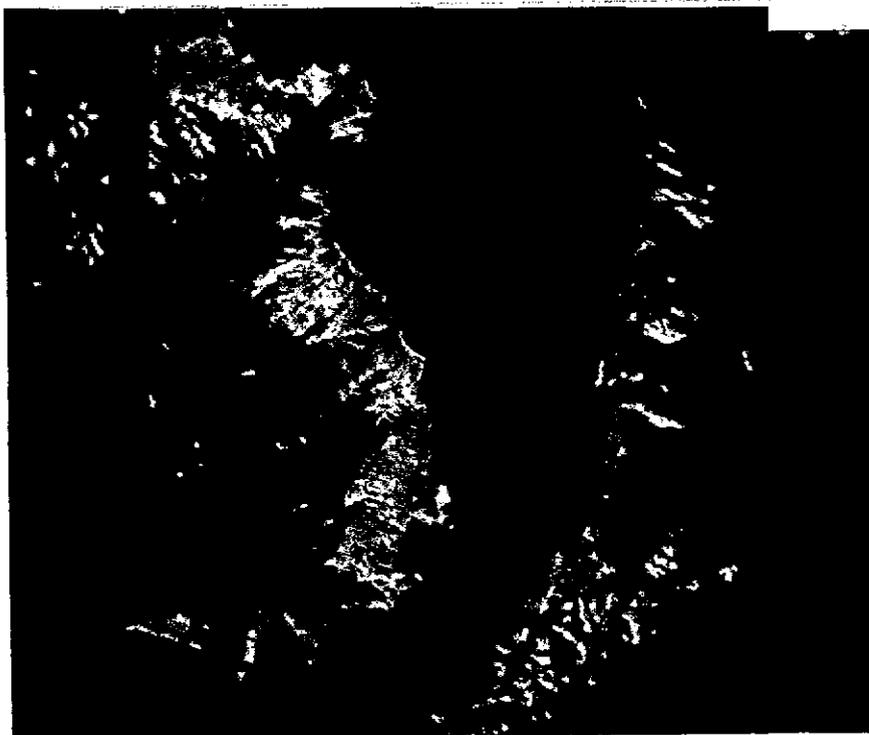


Fig. 6.17- Mosaico - Caraguá - banda5.

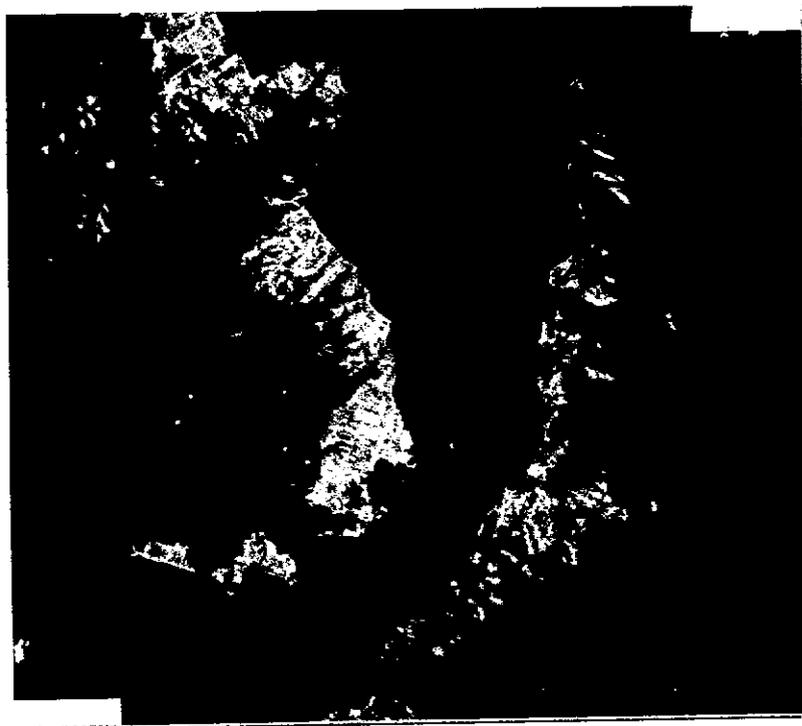


Fig. 6.18- Mosaico - Caraguá - banda7.

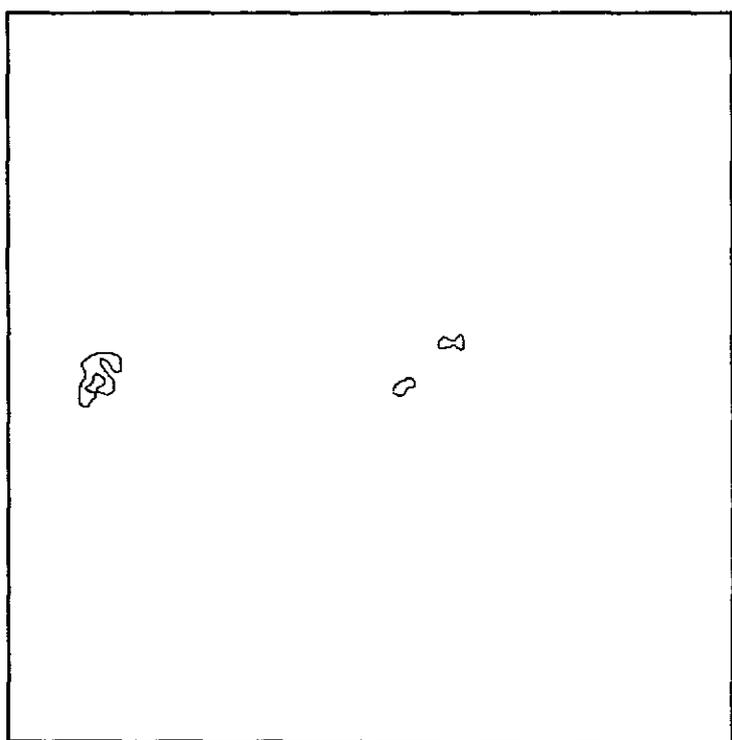


Fig. 6.19- Pinda - feições encontradas - banda3.

Imagem de ajuste → Natividade2.raw - (26/06/97)

As Figuras 6.21 e 6.22 mostram as feições casadas da banda 3 e o mosaico da imagem obtida após o registro.

#### 6.2.4 Quarto Registro: Mato Grosso

Imagem de referência → Mato1.raw - (18/08/95)

Imagem de ajuste → Mato2.raw - (06/07/97)

Nas bandas 3 e 4 não foram adquiridas feições suficientes para se fazer o registro.

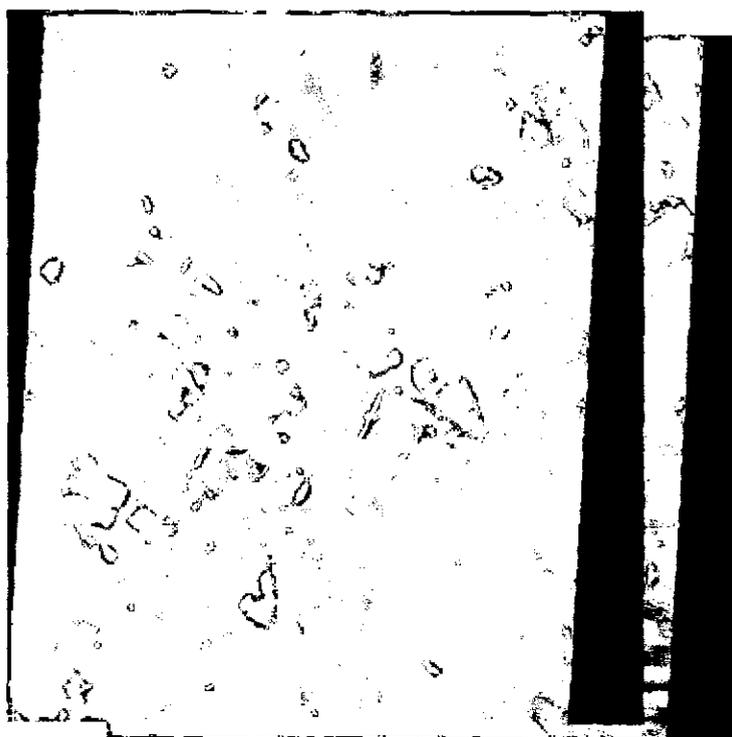


Fig. 6.20- Mosaico - Pinda - banda3.

As Figuras 6.23 e 6.24 mostram as feições casadas da banda 7 e o mosaico da imagem obtida após o registro.

### 6.2.5 Demais Ilustrações

As Figuras 6.25, 6.26, 6.27 e 6.28 mostram um exemplo de zoom obtido dos pontos de teste da imagem de Caraguá.

As Figuras 6.29 e 6.30 mostram os contornos encontrados pelo Método de Feições de algumas imagens.

Aplicando a transformação nos pontos de teste da imagem de ajuste através da equação 4.4 do Capítulo 4 podemos calcular as novas coordenadas para cada uma das bandas, pois os parâmetros

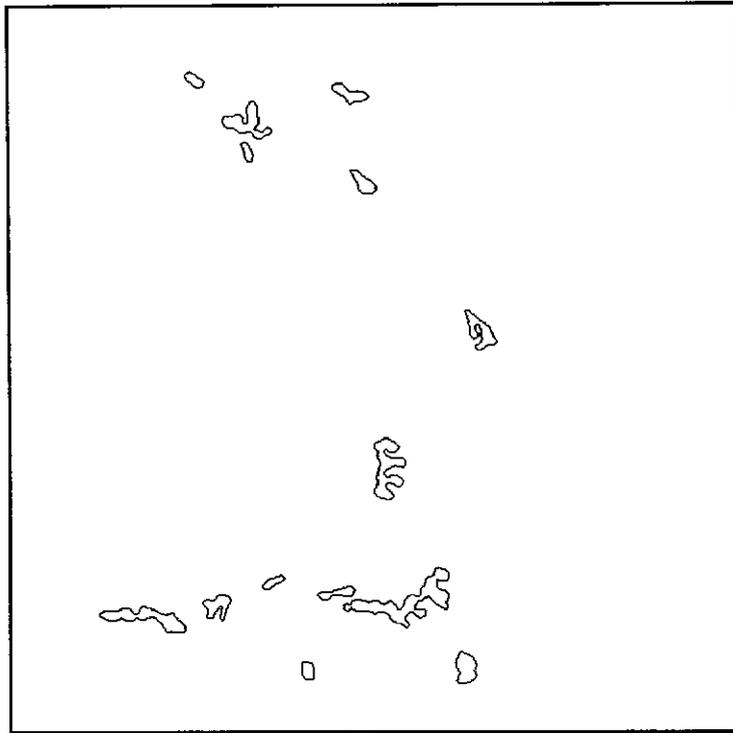


Fig. 6.21- Natividade - feições encontradas - banda3.

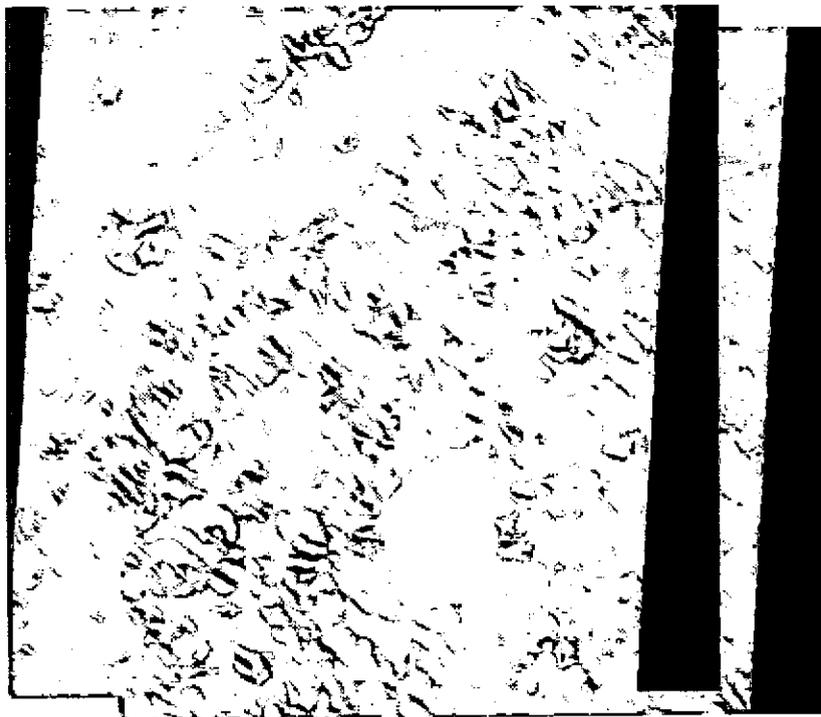


Fig. 6.22- Mosaico - Natividade - banda3.

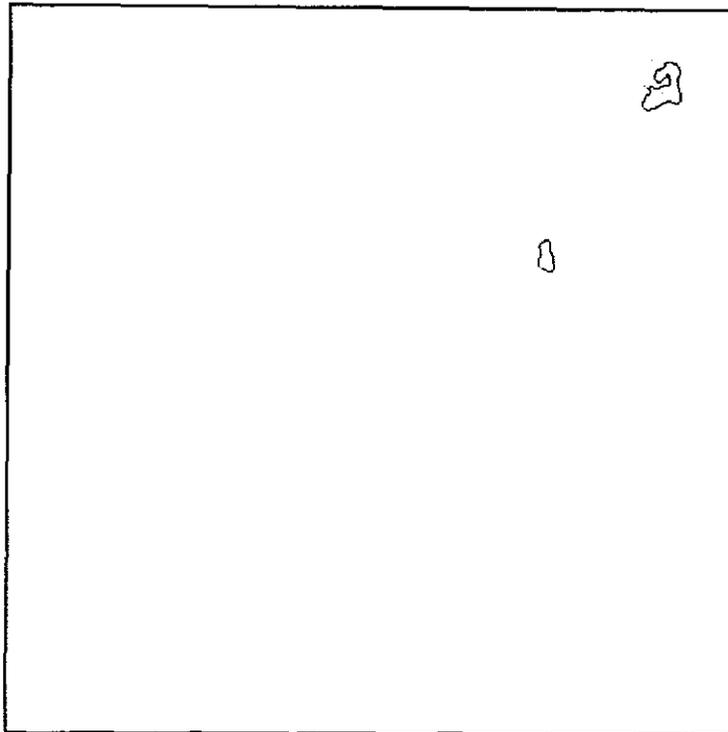


Fig. 6.23- Mato - feições encontradas - banda7.



Fig. 6.24- Mosaico - Mato - banda7.

da transformação mudam de banda para banda. As Tabelas 6.10, 6.11, 6.12 e 6.13 mostram as coordenadas obtidas para a imagem de Caraguá, Pinda, Natividade e Mato, respectivamente.

A Tabela 6.14 mostra os resultados dos parâmetros gerados pela transformação para o Método de feições.

A Tabela 6.15 mostra uma comparação do erro de registro RMSE entre o Método manual e o Método de feições.

A Tabela 6.16 mostra os resultados obtidos para o Método da Variância.

A Tabela 6.17 mostra uma comparação entre o erro de registro calculado pelo Método da Variância e o Método RMSE.

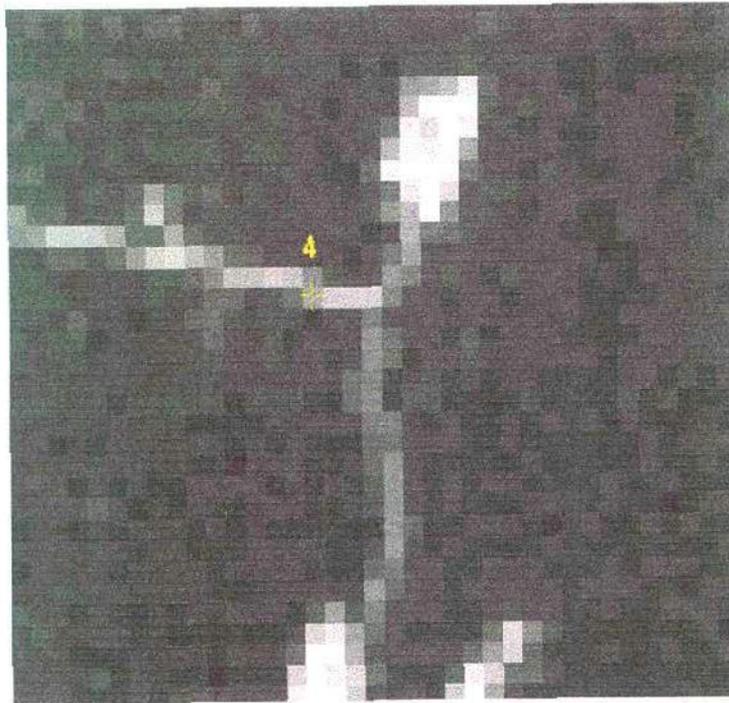


Fig. 6.25- Caraguá - Ponto4.



Fig. 6.26- Caraguá - Ponto5.



Fig. 6.27- Caraguá - Ponto7.

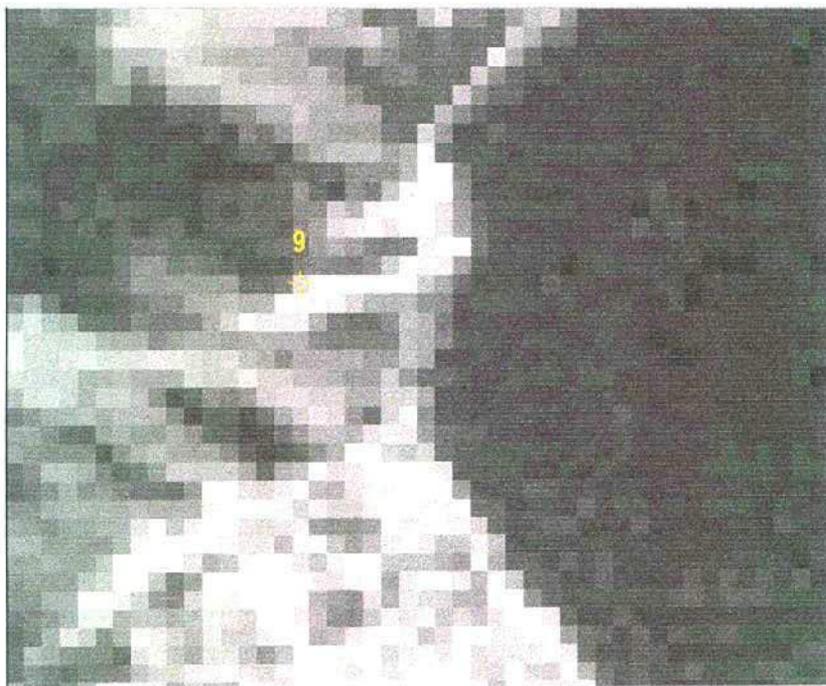


Fig. 6.28- Caraguá - Ponto9.

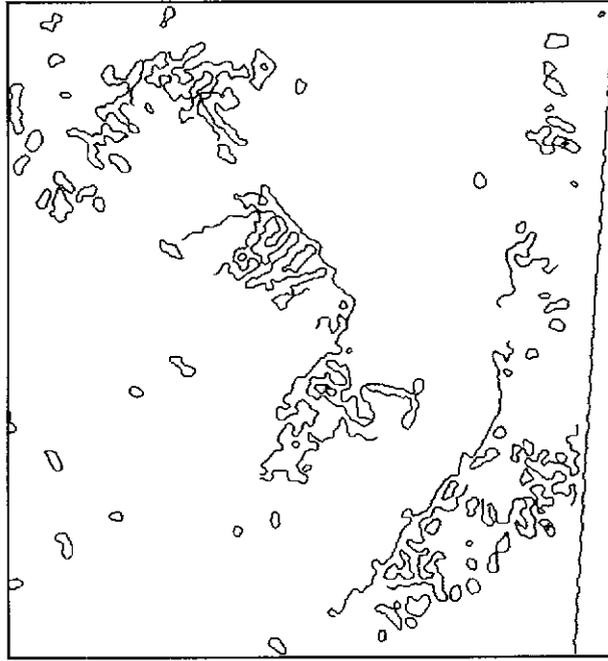


Fig. 6.29- Contornos encontrados de Caraguá1 - banda3.

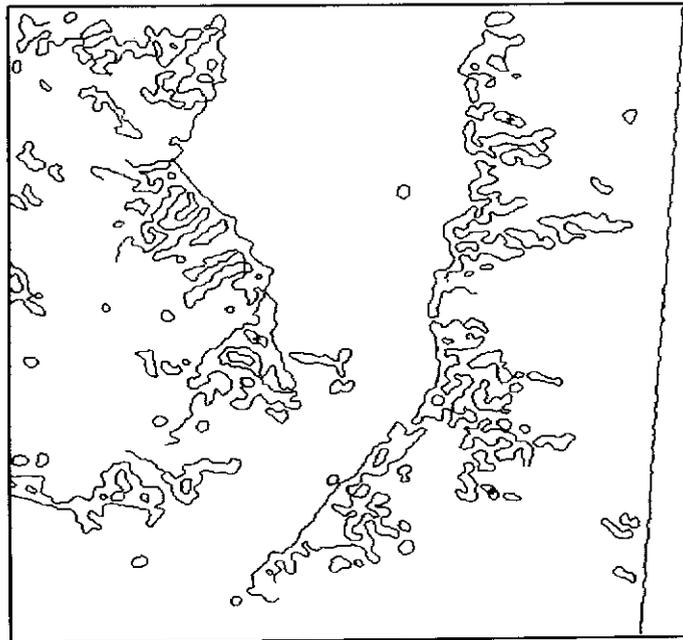


Fig. 6.30- Contornos encontrados de Caraguá2 - banda3.

TABELA 6.10: CARAGUÁ - PONTOS APROXIMADOS - MÉTODO AUTOMÁTICO

Imagem	Banda	Pot4(x,y)	Pot5(x,y)	Pot7(x,y)	Pot9(x,y)
Caraguá	3	286, 263	411, 89	177, 355	128, 146
	5	287, 262	410, 88	180, 354	129, 147
	7	287, 264	412, 90	178, 356	129, 147

TABELA 6.11: PINDA - PONTOS APROXIMADOS - MÉTODO AUTOMÁTICO

Imagem	Banda	Pot2(x,y)	Pot5(x,y)	Pot7(x,y)	Pot8(x,y)
Pinda	3	121, 436	326, 395	349, 168	485, 174
	4	121, 436	326, 396	348, 169	484, 175
	5	120, 436	326, 396	350, 168	486, 175

TABELA 6.12: NATIVIDADE - PONTOS APROXIMADOS - MÉTODO AUTOMÁTICO

Imagem	Banda	Pot1(x,y)	Pot4(x,y)	Pot5(x,y)	Pot7(x,y)
Natividade	3	71, 135	173, 204	198, 381	278, 165
	4	71, 136	173, 205	199, 381	277, 165
	5	72, 135	174, 204	198, 381	278, 165
	7	71, 135	173, 204	198, 381	277, 165

TABELA 6.13: MATO - PONTOS APROXIMADOS - MÉTODO AUTOMÁTICO

Imagem	Banda	Pot3(x,y)	Pot6(x,y)	Pot7(x,y)	Pot8(x,y)
Mato	5	187, 239	288, 71	247, 375	321, 253
	7	188, 239	290, 74	246, 376	322, 255

TABELA 6.14: PARÂMETROS DA TRANSFORMAÇÃO - MÉTODO AUTOMÁTICO

Imagem	Banda	k	$\Delta\theta$	$\Delta x$	$\Delta y$
Caraguá	3	1.001343	0.019989	-21.224058	-87.164589
	5	0.993288	0.431916	-20.326373	-83.828296
	7	0.999564	0.084737	-20.544712	-85.7494
Pinda	3	0.999145	0.041574	17.881774	84.735606
	4	0.997516	0.184309	17.608428	86.648448
	5	1.001927	-0.252268	18.397373	82.680453
Natividade	3	1.002248	-0.113516	18.341969	83.635186
	4	1.001585	0.042967	17.957208	84.421476
	5	1.001539	-0.119046	18.491568	83.727107
	7	1.002151	-0.050152	17.952634	83.732483
Mato	5	1.010314	0.392822	-31.408156	-79.978100
	7	1.006623	-0.252711	-25.985078	80.715321

TABELA 6.15: COMPARAÇÃO ENTRE O MÉTODO MANUAL E O AUTOMÁTICO

RMSE	Manual(pixel)	Hui Li(pixel)
Caraguá	1.0	banda 3: 1.0 banda 5: 1.732 banda 7: 1.0
Pinda	1.118	banda 3: 0.707 banda 4: 1.322 banda 5: 1.118 banda 7: —
Natividade	1.0	banda 3: 0.707 banda 4: 1.0 banda 5: 1.0 banda 7: 0.866
Mato	1.323	banda 3: — banda 4:— banda 5 : 2.398 banda 7 : 2.00

TABELA 6.16: VALORES OBTIDOS PARA O MÉTODO DA VARIÂNCIA

Variância		emedx	emedy	varx	vary	egmx	egmy	egmt
Caraguá	manual	0.0	0.046	0.0	0.205	0.0	0.159	0.159
	banda 3	0.0	0.058	0.0	0.127	0.0	0.069	0.069
	banda 5	0.0	0.110	0.0	0.385	0.0	0.275	0.275
	banda 7	0.0	0.060	0.0	0.214	0.0	0.154	0.154
Pinda	manual	0.0	0.272	0.0	0.667	0.0	0.395	0.303
	banda 3	0.0	0.170	0.0	0.286	0.0	0.115	0.401
	banda 4	0.0	0.314	0.0	1.090	0.0	0.775	0.776
	banda 5	0.129	0.095	0.143	0.571	0.014	0.476	0.490
	banda 7	—	—	—	—	—	—	—
Natividade	manual	0.0	0.0	0.0	0.1	0.0	0.1	0.1
	banda 3	0.0	0.0	0.0	0.167	0.0	0.167	0.167
	banda 4	0.0	0.0	0.0	0.233	0.0	0.233	0.233
	banda 5	0.0	0.0	0.0	0.053	0.0	0.053	0.053
	banda 7	0.0	0.086	0.0	0.163	0.0	0.077	0.077
Mato	manual	0.068	0.203	0.366	0.347	0.298	0.145	0.443
	banda 3	—	—	—	—	—	—	—
	banda 4	—	—	—	—	—	—	—
	banda 5	0.103	0.289	0.470	0.466	0.367	0.177	0.543
	banda 7	0.010	0.255	0.311	0.487	0.301	0.232	0.533

TABELA 6.17: COMPARAÇÃO ENTRE O MÉTODO RMSE E A VARIÂNCIA

Imagem		RMSE(pixel)	Variância(pixel)
Caraguá	manual	1.0	0.399
	banda 3	1.0	0.263
	banda 5	1.732	0.524
	banda 7	1.0	0.393
Pinda	manual	1.118	0.628
	banda 3	0.707	0.340
	banda 4	1.322	0.880
	banda 5	1.118	0.700
	banda 7	—	—
Natividade	manual	1.0	0.316
	banda 3	0.707	0.408
	banda 4	1.0	0.483
	banda 5	1.0	0.229
	banda 7	0.866	0.278
Mato	manual	1.323	0.665
	banda 3	—	—
	banda 4	—	—
	banda 5	2.398	0.737
	banda 7	2.0	0.730

## CAPÍTULO 7

### CONCLUSÕES E COMENTÁRIOS

Foram analisados e testados dois métodos de registro de imagens, um manual (tradicional) e o outro automático (algoritmo de Li et al.). Do ponto de vista de aquisição dos pontos de controle o método manual é mais trabalhoso, uma vez que estes pontos são escolhidos pelo próprio usuário, diretamente na tela do computador. Se as imagens são multi-temporais a dificuldade de obtê-los é maior ainda.

No método manual o registro foi realizado sobre a banda 3, por ser uma banda mais clara, tornando mais fácil visualizar os pontos de controle. A manipulação das imagens e a escolha dos pontos de controle foram feitas utilizando o SPRING [11, 12] .

Os métodos automáticos requerem, via de regra, que as imagens estejam bem nítidas, com os contornos bem definidos; caso contrário o algoritmo não consegue encontrar muitos contornos, e conseqüentemente não obterá muitos casamentos. Com o algoritmo de Li et al., aplicado às imagens exemplificadas neste trabalho, obtiveram-se poucos contornos fechados. Apenas no caso da imagem de Natividade é que se obtiveram mais feições, devido ao fato dessa área exibir muito relevo. O registro automático, porém, é bastante rápido: executado numa estação Sun Sparc Ultra 30/300, consome em torno de 50 segundos de CPU para cada imagem, enquanto que o método manual pode levar horas dependendo da quantidade de pontos de controle que se deseja obter.

Em muitos casos o método automático pode não ser a melhor alternativa; por exemplo, quando

a detecção de contornos não é eficiente, ou quando se deseja um ponto de controle específico que o algoritmo não consegue selecionar.

Para analisar a precisão do registro das imagens foram calculados o Erro de Registro através de dois métodos. Um deles, de uso tradicional, é o RMSE (*Root Mean Square Error*), que é a medida dos erros residuais do conjunto de pares de pontos de teste escolhidos manualmente. O método da Variância, que não é de uso comum, procura separar o erro de registro real do erro que ocorre durante a escolha manual dos pontos de teste nas imagens de referência e de ajuste.

No método RMSE foram usados pontos de teste obtidos manualmente nas imagens de referência e de ajuste. Estes pontos não são usados pela equação de mapeamento, mas os seus resíduos são calculados e usados para medir a precisão do registro.

Analisemos os resultados obtidos na Tabela 6 – 15 do Capítulo 6 para o RMSE:

- *Imagem Caraguá*: comparando o método manual com as bandas 3 e 7 do método automático notamos que o valor RMSE encontrado foi o mesmo. Já para a banda 5 o RMSE do método automático foi maior. Se analisarmos as feições encontradas nesta banda (Fig 6.14), observamos que estão muito próximas espacialmente; o ideal é que estes pontos estivessem espalhados pela imagem para assegurar um bom registro global.
- *Imagem Pinda*: mesmo encontrando somente 3 feições na banda 3, o método automático mostrou melhor desempenho que o método manual. Na banda 5 os dois métodos resultaram em registros comparáveis. Na banda 4 foram encontradas so-

mente 2 feições muito próximas, por isso o valor do RMSE foi maior. Na banda 7 o algoritmo não conseguiu detectar feições.

- *Imagem Natividade*: neste caso o método automático mostrou-se igual ou melhor do que o método manual. Para todas as bandas analisadas o número de feições encontradas foi grande e estão bem espalhadas, o que explica a melhor eficiência do método automático.
- *Imagem Mato*: o método manual foi melhor. Esta imagem é de uma área agrícola que sofreu muitas mudanças, por isso o método automático não conseguiu determinar muitos contornos semelhantes. Isto talvez fosse remediável através de uma mudança nos limiares de semelhança utilizados pelo algoritmo de Li et al., o que, na prática, pode implicar em um processo trabalhoso de tentativa e erro que acaba por não justificar o uso do registro automático.

Analisando a Tabela 6 – 17 para o Método da Variância notamos que, para todas as imagens, o erro de registro encontrado foi menor do que 1 pixel (em muitos casos, até menor que 0.5 pixel). Lembrando que, na prática, um bom registro deve ser menor que 1 pixel, isto mostra que, ao menos operacionalmente, o processo de separação de erros permite evidenciar a qualidade do registro, que fica limitada à precisão visual no cálculo simples RMSE. Em suma, o método da variância mostrou ser mais recomendável, uma vez que trabalha com dois tipos de erros, o erro de medida gerado pelo processo que está sendo usado para calcular o erro, e o erro geométrico que é o valor que desejamos determinar. Ele é mais confiável porque calcula o erro total e subtrai o erro de medida, sendo então o resultado obtido mais próximo do erro de registro verdadeiro entre as imagens.

Por outro lado, no método da Variância, pontos são selecionados automaticamente através do método de correlação. É importante salientar que muitos pontos encontrados estavam errados e precisaram ser descartados. Como exemplo, na imagem de Caraguá foram encontrados pontos na água, enquanto que na imagem de Natividade (bandas 3 e 4) o valor da correlação usado para poder encontrar os pontos foi baixo, o que gerou muitos pontos errados. Sendo assim é necessário analisar e selecionar os pontos encontrados durante a aplicação do Método da Variância.

Para os dois métodos utilizados, manual ou automático, foi utilizada uma transformação de similaridade. À guisa de um teste conceitual e técnico, foi calculado também o RMSE utilizando uma transformação mais geral, como a afim (ver Tabela 7.18). Notamos que, embora a transformação afim possa melhorar o ajuste entre os pontos de controle, em muitos casos isto não significa que o registro ficará bom para o resto da imagem. Havendo motivos para se adotar apenas distorções rígidas, se a técnica utilizada permitir inversões ou cisalhamento, o resultado final pode ser desastroso. Foi o que aconteceu com o método automático nas imagens onde se detectaram poucos contornos: a transformação afim, dá um grau maior de liberdade pois permite variar independentemente as escalas nas direções horizontal e vertical [16], pode no entanto causar efeitos estranhos na imagem como um todo.

TABELA 7.18: COMPARAÇÃO ENTRE OS MÉTODOS USANDO TRANSE. AFIM

RMSE	Manual(pixel)	Hui Li(pixel)
Caraguá	0.707	banda 3: 1.936 banda 5: 4.062 banda 7: 2.345
Pinda	0.707	banda 3: 7.246 banda 4: – banda 5: 1.225 banda 7: –
Natividade	0.707	banda 3: 1.0 banda 4: 0.5 banda 5: 0.707 banda 7: 1.0
Mato	1.0	banda 3: – banda 4: – banda 5: 7.036 banda 7: –

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Erthal, G. J. **Registro translacional de imagens: métodos correlacionais e sequenciais.** São José dos Campos. 76p. (INPE-3456-TDL/189). Dissertação (Mestrado em Sensoriamento Remoto) - Instituto Nacional de Pesquisas Espaciais, 1984 .
- [2] Fonseca, L. **Registro automático de imagens de sensoriamento remoto baseado em múltiplas resoluções.** São José dos Campos. 212p. Dissertação (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, 1999 .
- [3] Silva, L. S. **Desenvolvimento de uma tecnologia fotogramétrico-digital para a medição e a documentação de edificações.** São José dos Campos. 115p. (INPE-7117-TDI/669). Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, 1998.
- [4] Parker, J. R. **Algorithms for image processing and computer vision.** Toronto: John Wiley and Sons, 1997. 417p.
- [5] Alexander, H. W. **Elements of mathematical statistics.** Richmond: John Wiley and Sons, 1961. 367p.
- [6] Barnsley, M. **Fractals everywhre.** 2.ed. Cambridge: AP Professional, 1988. 531p.
- [7] Richards, J. R. **Remote sensing digital image analysis: an introduction.** Berlin: Springer-Verlag, 1986. 281p.
- [8] Wolberg, G. **Digital image warping.** Los Alamitos: IEEE Computer Society Press, 1990. 318p.

- [9] Namikawa, L. M. **Um método de ajuste de superfície para grades triangulares considerando linhas características**. São José dos Campos. 118p. (INPE-6162-TDI/583). Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, 1995.
- [10] Niblack, W. **An introduction to digital image processing**. Englewood Cliffs: Prentice/Hall International, 1986. 215p.
- [11] Instituto Nacional de Pesquisas Espaciais. Divisão de Processamento de Imagens (INPE-DPI). **SPRING- Sistema de Processamento de Informações Georeferenciadas**. [on line]. <<http://www.dpi.inpr.br/spring>>. Jun. 1998.
- [12] Câmara, G.; Souza, R. C. M.; Freitas, U. M.; Garrido, J. SPRING: integrating remote sensing and GIS by object oriented data modeling. **Computers and Graphics**, v. 20, n. 3, p. 395-403, May-June 1996.
- [13] Pratt, W. K. Correlation techniques of image registration. **IEEE Transactions on Aerospace and Electronic Systems**, v.10, n.3, p.353-358, May 1974.
- [14] Zheng, Q.; Chellappa, R. A. A computational vision approach to image registration. **IEEE Transactions on Image Processing**, v.2, n.3, p.311-326, 1993.
- [15] Cideciyan, A. V.; Jacobson, S. G.; Kemp, C. M.; Knighton, R. W.; Nagel, J. H. Registration of high resolution images of the retina. In: The International Society for Optical Engineering (SPIE), Newport Beach, 1992. **Proceedings**. Bellingham: SPIE, 1992. v. 1652, p. 310-322.
- [16] Mascarenhas, N. D. A.; Velasco, F. R. D. **Processamento digital de imagens**. 2.ed. Buenos Aires: Kapelusz, 1989.

- [17] Ballard, D. H.; Brown, C. M. **Computer vision**. Englewood Cliffs: Prentice-hall, 1982.  
523p.
- [18] Li, H., Manjunath, B. S.; Mitra, S. K. A contour-based approach to multisensor image registration. **IEEE Transactions on Image Processing**, v.4, n.3, p.320–334, 1995.
- [19] Chui, C. **Introduction to wavelets**. New York: Academic Press, 1992.
- [20] Mallat, S. G. Multifrequency channel decompositions of images and wavelet models. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v.37, n.12, p.2091–2110, Dez. 1989.
- [21] Pratt, W. K. **Digital image processing**. 2.ed. Mountain View: John Wiley and Sons, 1991.  
698p.
- [22] Gonzales, R. C.; Woods, R. E. **Digital Image Processing**. Massachusetts: Addison-Wesley Publishing Company, 1992. 716p.
- [23] Marr, D. **VISION: A computational investigation into the human representation and processing of visual information**. San Francisco: W. H. Freeman and Company, 1982.  
397p.
- [24] Anzai, Y. **Pattern recognition and machine learning**. San Diego: Academic Press, 1989.  
407p.
- [25] Freeman, H. Shape description via the use of critical points. **Pattern Recognition**, v.9, n.1, p.159-165, 1977.
- [26] Baird, H. S. **Model-based image matching using location**. Massachusetts: The MIT Press, 1984. 103p.

- [27] Heijden, F. **Image based measurement systems**. Chichester: John Wiley & Sons, 1994.  
338p.
- [28] Hu, M. K. Visual Pattern Recognition by Moment Invariants. **IRE Trans. Information Theory**, v.8, n.2, p.179-187, Feb. 1962.
- [29] Dougherty, E. R.; Giardina, C. R. **Image processing - continuous to discrete**. Englewood Cliffs: Prentice-hall, 1987. v. 1, 452p.

## APÊNDICE A

### PROGRAMAS EM C

#### A.1 Cálculo da Correlação

```
//— Arquivos de inclusao —
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
//— Area de dados globais —
# define NROW 512    // tamanho da imagem
# define NCOL 512
# define NPTX 100
# define tam_busca 7
unsigned char imaRef[NROW][NCOL],    // dados de imagens
             imaAju[NROW][NCOL],
             imaSai[NROW][NCOL];
unsigned char nomeRef[100],    // nomes de arquivo
             nomeAju[100],
             nomeSai[100];

//————— parametros de processamento
//int tam_busca = 7;
int tam_chip = 51;
int espaco = 20;
float lim_ho = 0.7;
int num_blocos = 5;
//————— HO / ERROS
double ho[NROW][NCOL];
int delta_lin[NROW][NCOL];
int delta_col[NROW][NCOL];
int l = 0; int k = 0;
//————— Funcoes auxiliares —
int leImagem ( unsigned char*, unsigned char [][][NCOL] );
int salvaImagem( unsigned char*, unsigned char [][][NCOL] );
double correlacao( int iR, int jR, int iA, int jA );
void maxcorrelacao();
```

```

void pontos();
//----- Programa principal -----
void main()
int linr = 100,
    colr = 100,
    lina = 100,
    cola = 100;
double aux[tam_busca][tam_busca];
double ho = 0.;
int l,k,m,n,lr,cr;
// abertura do arquivo de saida
FILE *fp;
if ((fp = fopen("correl.txt","wt")) == NULL)
{
    printf("Erro na abertura do arquivo");
    exit(1);
}
    l = k = 0;
//--- Le arquivos de entrada ---
printf("Imagem de referencia: ");
scanf("%s", nomeRef );
leImagem( nomeRef, imaRef );
printf("Imagem registrada: ");
scanf("%s", nomeAju );
leImagem( nomeAju, imaAju );
// loop para varrer toda a imagem
for (linr = 20; linr < NROW-tam_chip; linr += espaco)
{
    for (colr = 120; colr < (NCOL-50); colr += espaco)
    {
// loop para procurar pontos na janela
        for (lr = linr; lr < linr + 16; lr += 5)
        {
            for (cr = colr; cr < colr + 16 ; cr += 5)
            {
                l = 0;
// loop para varrer a janela de busca

```

```

for (lina = (lr - tam_busca/2); lina <= (lr + tam_busca/2); lina++)
{
k = 0;
for ( cola = (cr - tam_busca/2); cola <= (cr + tam_busca/2); cola++)
{
ho = correlacao( lr, cr, lina, cola );
aux[l][k] = fabs(ho);
k++;
}
l++;
}

fprintf (fp, "%d %d ""\n", lr, cr);
for (m=0; m<7; m++)
{
for (n=0; n<7; n++)
{
fprintf(fp, "% %lf ", aux[m][n]);
}
fprintf(fp, "\n");
}
}
}
}
fclose(fp);
maxcorrelacao();
pontos();
}
//_____
int leImagem( unsigned char*nome, unsigned char image[][NCOL] )
{
int i;
FILE *fp;
if ( (fp = fopen( (const char*)nome, "rb" )) == NULL)
printf("Erro na abertura do arquivo %s", nome);
exit(1);
for( i = 0; i < NROW; i++)

```

```

    {
        fread( image[i], NCOL, 1, fp );
    }
    fclose( fp );
    return 1;
}
//-----
int
salvaImagem(unsigned char*nome, unsigned char image[][NCOL])
{
    int i;
    FILE *fp = fopen( (const char*)nome, "wb" );
    if( !fp ) return 0;
    for( i = 0; i < NROW; i++ )
    {
        fwrite( image[i], NCOL, 1, fp );
    }
    fclose( fp );
    return 1;
}
//-----
int
diferenca( unsigned char ima1[][NCOL],
           unsigned char ima2[][NCOL],
           unsigned char ima3[][NCOL],
           int offset )
{
    int i,j;
    int pix1, pix2, pix3;
    for( i = 0; i < NROW; i++ )
    {
        for( j = 0; j < NCOL; j++ )
        {
            pix1 = ima1[i][j];
            pix2 = ima2[i][j];
            pix3 = pix1 - pix2 + 127;
            if( pix3 > 255 ) pix3 = 255;
        }
    }
}

```

```

        if( pix3 < 0 ) pix3 = 0;
        ima3[i][j] = pix3;
    }
}
return 1;
}
//-----
double
correlacao( int iR, int jR, int iA, int jA )
{
    int i,j ;
    double ex = 0.,
           ey = 0.,
           exy = 0.,
           exx = 0.,
           eyy = 0.;
    double pixR,
           pixA,
           n2 = tam_chip * (double)tam_chip;
    for( i= 0; i < tam_chip; i++ )
    {
        for( j= 0; j < tam_chip; j++ )
        {
            pixR = imaRef[i+iR][j+jR];
            pixA = imaAju[i+iA][j+jA];
            ex += pixR;
            ey += pixA;
            exy += pixR * pixA;
            exx += pixR * pixR;
            eyy += pixA * pixA;
        }
    }
    ex /= n2;
    ey /= n2;
    exx /= n2;
    eyy /= n2;
    exy /= n2;
}

```

```

return ( exy - ex * ey ) / sqrt( ( exx - ex*ex ) * ( eyy - ey*ey ) );
}
//-----
// calcula a correlacao maxima
void
maxcorrelacao()
{
    int    i,j,linr,colr,l,c,ret ;
    double    ho, aux;
    FILE *fp,*fp1;
    if ((fp = fopen("correl.txt","rt")) == NULL)
    {
        printf("Erro na abertura do arquivo");
        exit(1);
    }
    if ((fp1 = fopen("maxcor.txt","wt")) == NULL)
    {
        printf("Erro na abertura do arquivo");
        exit(1);
    }
do
{
ho = 0.0;
if( fscanf(fp,"%d %d", &linr,&colr)==EOF)
    break;
for (i = (linr - tam_busca/2); i <= (linr + tam_busca/2); i++)
{
    for (j = (colr - tam_busca/2); j <= (colr + tam_busca/2); j++)
    {
        ret=fscanf(fp,"%lf", &aux);
        if (aux > ho)
        {
            l = i;
            c = j;
            ho = aux;
        }
    }
}
}

```

```

    }
    if ((ho > lim_ho) && (ho <= 1))
        {
            fprintf (fp1, "%d %d %d %d %d %lf\n", linr, colr, l, c, ho);
        }
} while (ret != EOF); // fim do while
fclose(fp);
fclose(fp1);
}
//-----
void
pontos()
{
    int i,j,k,l,linr,colr,lina,cola,ret;
    int auxi[50], auxj[50], auxlina[50], auxcola[50];
    double homax;
    FILE *fp1;
    FILE *out1;
    if ((fp1 = fopen("maxcor.txt", "rt")) == NULL)
        {
            printf("Erro na abertura do arquivo");
            exit(1);
        }
    if ((out1 = fopen("ptos.txt", "wt")) == NULL)
        {
            printf("Erro na abertura do arquivo");
            exit(1);
        }
    k = 0;
//loop para varrer toda a imagem
for (linr = 5; linr < NROW; linr += espaco)
{
    for (colr = 5; colr < NCOL; colr += espaco)
    {
        do
        {
            if( fscanf(fp1, "%d %d %d %d %d %lf", &i,&j, &lina, &cola, &homax) == EOF)

```

```

break;
if ((i>=linr)&&(i<linr+espaco)&&(j>=colr)&&(j<colr+espaco))
{
//fprintf (out1, "%d %d %d %d ""\n",i,j,lina, cola);
auxi[k] = i;
auxj[k] = j;
auxlina[k] = lina;
auxcola[k] = cola;
k++;
// printf("%d %d %d %d""\n",i,j,lina,cola);
}
}
while (ret != EOF); // fim do while
// fprintf (out1, "0 0 0 0""\n");
if ( k != 0)
{
fprintf (out1, "%d""\n",k);
}
for (l=0; l<k; l++)
{
fprintf(out1, "%d %d %d %d""\n", auxi[l], auxj[l], auxlina[l], auxcola[l]);
}
rewind(fp1);
k = 0;
}
}
fclose(fp1);
fclose(out1);
}

```

## A.2 Cálculo da Variância

```
//— Arquivos de inclusao — #include <stdio.h>
#include <stdlib.h>
#include <math.h>
//————— Programa principal —
void main()
    int i,l,k,m,n,s,x;
    int linr, colr, lina, cola;
    int ex, ey, somaex, somaey;
    double mediaex, mediaey, varex, varey, varx, vary, xx, yy, aux;
    double desvioex[300], desvioey[300];
    double aux1[700], aux2[700], aux3[700], aux4[700], aux5[700];
    int isoex[700], isoe[700], janex[300], janey[300];
    double erro_medx, erro_medy, erro_geomx, erro_geomy, erro_total, erro_quad;
    FILE *fp,*out;
// le arquivo de entrada
    if ((fp = fopen("ptos.txt", "rt")) == NULL)
    {
        printf("Erro na abertura do arquivo");
        exit(1);
    }
//abre arquivo de saida
    if ((out = fopen("var.txt", "wt")) == NULL)
    {
        printf("Erro na abertura do arquivo");
        exit(1);
    }
    i = 0;
    x = 0;
do
{
    if( fscanf(fp, "%d", &n) == EOF)
        break;
    if ( n<=5)
    {
        fscanf(fp, "%d %d %d %d", &linr,&colr, &lina, &cola);
        ex = fabs(linr-lina);
```

```

        ey = fabs(colr-cola);
        isoex[i] = ex;
        isoey[i] = ey;
        i++;
    for (k=0; k<n-1; k++)
    {
        fscanf(fp, "%d %d %d %d", &linr, &colr, &lina, &cola);
    }
}
else
{
    somaex = 0;
    somaey = 0;
    for (k=0; k<n; k++)
    {
        fscanf(fp, "%d %d %d %d", &linr, &colr, &lina, &cola);
        ex = fabs(linr-lina);
        ey = fabs(colr-cola);
        janex[k] = ex;
        janey[k] = ey;
        somaex = janex[k] + somaex;
        somaey = janey[k] + somaey;
    }
    mediaex = (double)somaex/n;
    mediaey = (double)somaey/n;
    varex = varey = 0;
    for (m=0; m<n; m++)
    {
        desvioex[m] = pow(fabs(janex[m] - mediaex), 2);
        desvioey[m] = pow(fabs(janey[m] - mediaey), 2);
        varex = varex + desvioex[m];
        varey = varey + desvioey[m];
    }
    varx = (double)varex/(n-1);
    vary = (double)varey/(n-1);
    aux1[x] = varx;
    aux2[x] = vary;
}

```

```

        aux3[x] = n-1;
        x++;
    } // fim do else
} while ( fp != EOF); // fim do while
// calculo para pontos janelas
    xx = 0;
    yy = 0;
    aux = 0;
for (s=0; s<x; s++)
{
    aux4[s] = aux1[s] * aux3[s];
    aux5[s] = aux2[s] * aux3[s];
    xx = xx + aux4[s];
    yy = yy + aux5[s];
    aux = aux + aux3[s];
}
erro_medx = (double)xx/aux; // erro de medida de x
erro_medy = (double)yy/aux; // erro de medidad de y
//calculo para pontos isolados
    somaex = 0;
    somaey = 0;
    n = i-1;
for (l=0; l<n; l++)
{
    somaex = isoex[l] + somaex;
    somaey = isoey[l] + somaey;
}
    mediaex = (double)somaex/n;
    mediaey = (double)somaey/n;
    varex = varey = 0;
for (m=0; m<n ; m++)
{
    desvioex[m] = pow(fabs(isoex[m] - mediaex),2);
    desvioey[m]= pow(fabs(isoey[m] - mediaey),2);
    varex = varex + desvioex[m];
    varey = varey + desvioey[m];
}

```

```

    varx = (double)varex/(n-1);    // erro total de x
    vary = (double)varey/(n-1);    // erro total de y
// calculo geral
    erro_geomx = varx - erro_medx;
    erro_geomy = vary - erro_medy;
    erro_total = erro_geomx + erro_geomy;
    erro_quad = sqrt(erro_total);
fprintf (out, "erro_medx = %lf erro_medy = %lf\n\n", erro_medx, erro_medy);
fprintf (out, "varx = %lf vary = %lf\n\n", varx, vary);
fprintf (out, "erro_geomx = %lf erro_geomy = %lf\n\n", erro_geomx, erro_geomy);
fprintf (out, "erro_total = %lf erro_quad = %lf\n\n", erro_total, erro_quad);
fprintf (out, "num_ptos_iso = %d num_jan = %d\n", i, x);
}

```

**PUBLICAÇÕES TÉCNICO-CIENTÍFICAS  
EDITADAS PELO INPE**

**RELATÓRIOS DE PESQUISA**

- Reportam resultados de pesquisa tanto de natureza técnica quanto científica.

**PUBLICAÇÕES DIDÁTICAS**

- Apostilas, notas de aula e manuais didáticos

**NOTAS TÉCNICO-CIENTÍFICAS**

- Incluem resultados preliminares de pesquisa, descrição de equipamentos, software, sistemas e experimentos, apresentação de testes, dados e atlas, e documentação de projetos de engenharia.

**TESES E DISSERTAÇÕES**

- Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

**MANUAIS TÉCNICOS**

- Descrevem normas, procedimentos, instruções e orientações.

**PUBLICAÇÕES SERIADAS**

- Periódicos Técnico-Científicos:  
Boletim de Sensoriamento Remoto,  
Climanálise: Boletim de  
Monitoramento e Análise  
Climática.
- Anais de Eventos