



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-11568/TDI/960

**GERAÇÃO DE ORTOIMAGENS PARA APLICAÇÕES URBANAS:
DESENVOLVIMENTO DAS EQUAÇÕES E PROTÓTIPO
EXPERIMENTAL**

Carlos Frederico de Sá Volotão

Dissertação de Mestrado do Curso de Pós-Graduação em Sensoriamento Remoto,
orientada pelos Drs. Antônio Miguel Vieira Monteiro e Gilberto Câmara, aprovada em
28 de setembro de 2001.

528.711.7 : 621.376.5

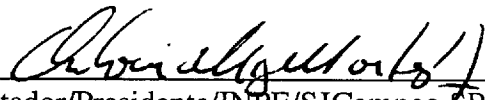
VOLOTÃO, C. F. S.

Geração de ortoimagens para aplicações urbanas:
desenvolvimento das equações e protótipo experimental /
C. F. S. Volotão. – São José dos Campos: INPE, 2001.
115p. – (INPE-11568-TDI/960).

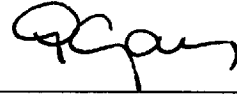
1.Ortofoto. 2.Correção de oclusão de superfícies.
3.Calibração. 4.Fotogrametria. 5.Processamento de
imagens. 6.Algoritmos. 7.Ortorretificação. I.Título.

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

Dr. Antônio Miguel Vieira Monteiro

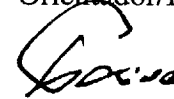

Orientador/Presidente/INPE/SJCampos-SP

Dr. Gilberto Câmara Neto



Orientador/INPE/SJCampos-SP

Dr. João Argemiro de Carvalho Paiva



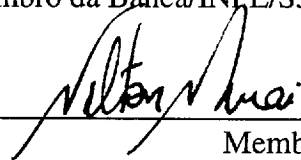
Membro da Banca/INPE/SJCampos-SP

Dr. Luciano Vieira Dutra



Membro da Banca/INPE/SJCampos-SP

Dr. Nilton Nobuhiro Imai



Membro da Banca
Convidado UNESP-Presidente Prudente-SP

Candidato: Carlos Frederico de Sá Volotão

São José dos Campos, 28 de setembro de 2001.

Se alguém necessita de sabedoria, peça-a a Deus, que a todos dá liberalmente e nada retém; e lhe será concedida. (Tia. 1:5)

À minha esposa Vivian que está sempre ao meu lado, apoiando-me e demonstrando diariamente que o casamento é uma união agradável e indispensável.

A meu filho Eric.

AGRADECIMENTOS

Primeiro a Deus, pois está sempre ao meu lado e me dá força e sabedoria; aos meus orientadores, Dr. Antônio Miguel Vieira Monteiro e Dr. Gilberto Câmara, que acompanharam o desenvolvimento deste trabalho e que me incentivaram, apoiaram, orientaram e ensinaram; ao Prof. Guaracy Erthal por importantes discussões e contribuição para a solução do problema de visibilidade e oclusão; ao Prof. Júlio D'Alge por sempre estar disposto a orientar e pelas muitas horas de análises e discussões; ao Antônio José Ferreira Machado e Silva, pela contribuição importante; ao Camillo Daleles Rennó e ao Capitão Antônio Henrique Correia pelo auxílio prestado, especialmente na programação da linguagem IDL; ao Gen João Venâncio de Melo Neto, Gen Armino Carvalho Fernandes, Cel Antônio Eliseu Dias Sommerlatte e Ten Cel Pedro Ronalt Vieira, pelo apoio; ao INPE, pelo curso de tamanha magnitude, com impecável corpo docente, de onde obtive importantes conhecimentos para a elaboração deste trabalho; e aos pesquisadores, funcionários e demais alunos pelo apoio; à FUNCATE, pela cessão de tempo de produção e material que me permitiram realizar o estudo de caso; ao Exército Brasileiro, à Diretoria de Serviço Geográfico (DSG) e ao Instituto Militar de Engenharia (IME), por permitirem a realização do curso.

RESUMO

Esta dissertação apresenta e descreve o desenvolvimento de um processo de ortorretificação, incluindo sua implementação em linguagem IDL. A ortorretificação é o processo de remapeamento de uma imagem numa projeção ortográfica. A ênfase deste trabalho está no desenvolvimento de um procedimento de correção para imagens obtidas por aerolevanteamento feito com câmaras não-métricas. O procedimento calcula parâmetros que definem a geometria interna da imagem e a relação entre seu sistema de referência com o da Terra. Utilizando a técnica proposta é possível calcular todos os parâmetros de uma imagem ou parte dela, sem necessidade de utilização de marcas fiduciais, desde que haja 6 pontos de controle. O procedimento ainda observa a possibilidade de oclusão de partes da superfície do terreno imageado.

**ORTHOIMAGE GENERATION FOR URBAN APPLICATIONS:
EQUATIONS DEVELOPMENT AND EXPERIMENTAL PROTOTYPE**

ABSTRACT

This work presents and describes the theoretical development of an orthorectification process, including its implementation in IDL language. The orthorectification is the remapping process of one image into an orthographic projection. This work emphasises a non-metric camera image correction procedure development. The procedure makes the image internal geometry parameters calculus and the relationship between its reference system and the Earth's reference system. By using this proposed technique it is possible to calculate the parameters of an image or part of it, without needing fiducial marks but having at least 6 control points. The procedure observes the possibility of occlusion of parts of the imaged terrain surface.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

CAPÍTULO 1	21
1.1 Objetivo	21
1.2 Organização do Trabalho.....	22
1.3 Contribuição	22
CAPÍTULO 2	25
2.1 Apresentação	25
2.2 Ortoimagens: Conceitos Básicos	25
2.3 Oclusão.....	29
2.4 Efeitos de Oclusão e Iluminação	33
2.5 Geometria de imageamento	35
2.6 Calibração e Orientação.....	36
CAPÍTULO 3	39
3.1 Dados em Projeção cartográfica.....	39
3.2 Calibração e Orientação.....	40
3.2.1 Considerações sobre a abordagem utilizada.....	40
3.2.2 Sistemas de coordenadas utilizados.....	41
3.2.3 As matrizes para o ajustamento	42
3.2.4 Orientação interior e exterior.....	47
3.3 Ortorretificação.....	52
3.4 Determinação dos Pixels Ocultos.....	53
3.4.1 “Método da Matriz de Distâncias”	55
3.4.2 “Método da Matriz de Distâncias”, interpolada	56
CAPÍTULO 4	59
4.1 Apresentação	59

4.2	Considerações Gerais	59
4.3	O funcionamento do algoritmo.....	60
4.3.1	O algoritmo.....	60
4.3.2	A área de estudo	64
4.3.3	A câmara utilizada no voo	66
4.3.4	Procedimento inicial.....	67
4.3.5	Localização dos pontos de controle.....	67
4.4	Avaliação.....	77
4.4.1	Nível de instabilidade da orientação.....	77
4.4.2	Resultado da Calibração	82
4.4.3	Avaliação da ortorretificação.....	85
4.4.4	Avaliação comparativa com produtos comerciais.....	86
CAPÍTULO 5		87
5.1	Conclusões.....	87
5.2	Sugestões para Trabalhos Futuros	88
REFERÊNCIAS BIBLIOGRÁFICAS		91
GLOSSÁRIO		95
APÊNDICE A - PROGRAMA DE ORTORRETIFICAÇÃO EM IDL		99
1.1	Programa de Ortorretificação	99
1.2	Identificação dos Pontos na Imagem.....	114

LISTA DE FIGURAS

	Pág	
2.1	Processo de geração de ortoimagens	26
2.2	Fenômeno físico, imagem e ortoimagem respectivas	27
2.3	Diferença entre ortoimagem corrigida e ortofoto convencional	28
2.4	Geometria de oclusão	30
2.5	Posição da imagem de um pixel	32
2.6	Perfil da geometria de oclusão	33
2.7	Telhados com superfície não-lambertiana	34
3.1	Transformação de coordenadas para um sistema local de coordenadas.	39
3.2	Representação gráfica da translação da Equação (3.5)	41
3.3	Matriz de Distâncias	53
3.4	Oclusão.	54
3.5	Significado da “Matriz de Distâncias”.	55
3.6	Significado da “Matriz de Distâncias”, interpolada.	56
3.7	Detalhe da interpolação.	57
4.1	Fluxo de utilização e geração de arquivos	60
4.2	Localização de Caraguatatuba	64
4.3	Fotografia aérea utilizada, da região de Caraguatatuba	65
4.4	Ortoimagem criada pela FUNCATE	66
4.5	Localização dos pontos na imagem	69
4.6	Ponto HV32	70
4.7	Ponto HV24	71
4.8	Ponto HV23	72
4.9	Ponto PT1525	73
4.10	Ponto PT1530	74
4.11	Ponto PT1532	75
4.12	Ponto PT2546	76
4.13	Deslocamento dos pixels em torno do pixel “correto”.	77
4.14	Efeito do deslocamento do ponto HV23 e parâmetro relacionado que mais aproxima-se do valor correto.	82
4.15	Efeito do deslocamento do ponto HV32 e parâmetro relacionado que mais aproxima-se do valor correto.	82
4.16	Ortoimagem gerada. Foi obtida a partir de um MNE de espaçamento igual a 1 m e 2001 linhas por 2001 colunas.	85

LISTA DE TABELAS

	Pág	
4.1	Varição máxima dos valores dos parâmetros ao variar um dentre 7 pixels numa vizinhança 12.	78
4.2	Varição máxima dos valores dos parâmetros, ao variar um dentre 7 pixels numa vizinhança 4 em cruz, com peso de pontos de campo 20 vezes maior.	78
4.3	Verificação do comportamento dos parâmetros, quando (ξ, η) variam de um único ponto. Maior e menor módulo do erro de f, ξ_0, η_0 .	80
4.4	Verificação do comportamento dos parâmetros, quando (ξ, η) variam de um único ponto. Maior e menor módulo do erro de $\varphi, \omega, \kappa, X_0, Y_0, Z_0$.	80
4.5	Verificação do comportamento dos parâmetros, quando (ξ, η) variam de um único ponto. Maior e menor módulo do erro de f, ξ_0, η_0 . Peso dos pontos de GPS 20 vezes maiores que os pontos de aerotriangulação.	81
4.6	Verificação do comportamento dos parâmetros, quando (ξ, η) variam de um único ponto. Maior e menor módulo do erro de $\varphi, \omega, \kappa, X_0, Y_0, Z_0$. Peso dos pontos de GPS 20 vezes maiores que os pontos de aerotriangulação.	81
4.7	Comparação dos melhores resultados obtidos com o deslocamento de um pixel, com o valor correto.	83

CAPÍTULO 1

INTRODUÇÃO

1.1 OBJETIVO

As imagens fotográficas surgiram em meados do século XIX e a utilização de câmaras a bordo de balões, aviões e satélites permitiram o registro panorâmico da superfície terrestre (Whitmore, 1952). Foi-se desenvolvendo a representação cartográfica, e a utilização de fotografias aéreas, após a 1ª Guerra Mundial, permitiu que as cartas fossem atualizadas em menor tempo (Branco, 1949). As técnicas tradicionais de mapeamento e o sensoriamento remoto surgiram em consequência natural do desenvolvimento da manipulação desse tipo de imagens.

Um dos maiores desafios da Ciência e Engenharia da Geoinformação é a transição entre os procedimentos tradicionais de coleta e produção da informação (com ênfase no tratamento analógico) para o uso de técnicas totalmente digitais. Um dos maiores impactos vem sendo o uso de imagens de alta resolução na produção de ortoimagens e de modelos numéricos de terreno. Em especial, o uso de câmaras não-métricas para levantamento de dados vem sendo proposto como uma alternativa de baixo custo.

A importância da ortorretificação está refletida nos sistemas analíticos de fotogrametria digital, que vêm substituindo os equipamentos de restituição. Os métodos digitais trazem novos recursos, e, tratando especificamente do escopo deste trabalho, permitem refinar o processo e as características finais de uma ortoimagem (Cory et al., 1999).

As ortoimagens obtidas por meio de fotografias aéreas estão se tornando hoje a base preferida para muitas aplicações de Sistemas de Informações Geográficas (SIG), devido ao baixo custo e alta precisão em relação aos métodos tradicionais, mais trabalhosos (Ammerman, 1998).

Nesta perspectiva, este trabalho explora o desenvolvimento e teste de procedimentos para ortorretificação de fotografias aéreas, obtidas por câmaras métricas ou não-métricas. Este procedimento resulta na construção de um módulo de ortorretificação, com seu desenvolvimento teórico e sua implementação computacional.

1.2 ORGANIZAÇÃO DO TRABALHO

O presente capítulo apresenta o objetivo, a organização e o que traz de contribuição. O capítulo 2 apresenta os fundamentos da metodologia de geração das ortoimagens, discutindo os principais conceitos empregados no desenvolvimento do procedimento computacional, apresentado em detalhe no capítulo 3. No capítulo 4, apresenta-se um estudo de caso que permite avaliar o procedimento proposto, e fazem-se algumas considerações finais no Capítulo 5.

1.3 CONTRIBUIÇÃO

O algoritmo de ortorretificação de imagens pelo processo estabelecido neste trabalho, obtidas a partir de câmaras fotográficas, métricas ou não, é apresentado, implementado em linguagem IDL (Research Systems, 1995a e 1995b), no Apêndice desta dissertação.

Há vasta literatura de fotogrametria abordando os problemas das orientações absoluta e relativa e as equações de colinearidade – que são o fundamento da ortorretificação (Moffitt e Mikhail, 1980; Wolf, 1974; Kraus, 1993; American Society of Photogrammetry, 1966; Albertz e Kreiling, 1989).

Mas, devido basicamente ao interesse comercial pela ortorretificação de imagens, muitos aspectos deste procedimento não são descritos na documentação dos *softwares*. São encontrados apenas seus resultados incorporados em algoritmos proprietários, cujos detalhes de implementação não são documentados. Neste contexto, a principal contribuição do trabalho é apresentar detalhadamente os procedimentos da ortorretificação de imagens, inclusive com os algoritmos na linguagem IDL utilizados.

O custo dos melhores sistemas fotogramétricos do mercado força que muitos usuários busquem alternativas à ortorretificação. A abordagem deste trabalho permite a realização de ortoimagens com o custo do software IDL ou somente com o custo da implementação em outra linguagem de programação. Academicamente pode servir como base para estudos complementares, no sentido de se integrar um pacote de fotogrametria digital automático que não exige o conhecimento prévio de parâmetros da câmara ou do voo.

Mantendo a filosofia adotada pelo projeto SPRING do INPE, a implementação de um módulo de fotogrametria digital, disponibilizado de modo gratuito e aberto (licença GNU), permite um aumento muito maior no desenvolvimento de novas funções e aperfeiçoamentos. Isto contribui para a redução do preço do mesmo tipo de programas no mercado (finalidade comercial), do acesso aos algoritmos pelos estudantes (finalidade acadêmica) e de um muito maior avanço tecnológico nesta área (finalidade científica).

Finalmente, para a Cartografia Brasileira, este trabalho apresenta a possibilidade de geração de um produto cartográfico sem necessidade do investimento de alto custo nas estações fotogramétricas, contribuindo, enfim, com a política brasileira de mapeamento sistemático e com as demais empresas de mapeamento.

CAPÍTULO 2

FUNDAMENTOS METODOLÓGICOS

2.1 APRESENTAÇÃO

O Glossário apresenta alguns termos básicos. Entretanto alguns conceitos apresentam substancial importância para a correta exposição da metodologia. Em decorrência disto, são conceituadas e apresentadas a seguir as explicações sobre ortoimagens, oclusão, iluminação, calibração e orientação, além da geometria de imageamento.

2.2 ORTOIMAGENS: CONCEITOS BÁSICOS

Ortoimagens são imagens em que as incorreções de posicionamento devido à inclinação da câmara e a altitude da superfície (ou do relevo) são removidas. A grande vantagem das ortoimagens sobre as cartas ou plantas cadastrais é a representação de toda a feição e não somente dos contornos e representações.

A teoria da ortorretificação é apresentada em livros e manuais de fotogrametria, dentre os quais estão Ghosh (1979), Thompson (1966), Moffitt e Mikhail (1980), Albertz e Kreiling (1989) e Kraus (1993). Os dois últimos apresentam o assunto numa abordagem que facilita a implementação.

A ortomagem é obtida a partir da ortorretificação, que consiste na correção da imagem devido às diversas distorções, especialmente as causadas pela posição do satélite, pela superfície física da Terra (relevo e curvatura) e pela projeção cartográfica (Zhang, 1999). Para a realização da ortorretificação das imagens, necessariamente deve haver um MNE (modelo numérico de elevação), sem o qual não seria possível a correção devido ao relevo. Se não houver MNE, pode-se considerar a Terra plana, e um valor médio de elevação pode ser utilizado.

O princípio da geração de ortoimagens a partir de um MNE é conceitualmente simples, como ilustrado na Figura 2.1. Na Figura 2.1 (a), a linha marrom representa o terreno, enquanto a linha rosa representa a imagem formada. Os triângulos verdes representam um alvo (sobre o terreno e seu homólogo sobre a imagem, respectivamente). A posição

em que o valor do sinal recebido do alvo verde será registrado é o triângulo azul sobre a reta que vai do alvo ao centro de projeção. A posição na qual deveria estar está representada pelo triângulo vermelho, na vertical do alvo na imagem. A Figura 2.1 (b) mostra a geometria inversa que recupera a posição vermelha a partir do azul e da interseção da linha de imageamento com o MNE.

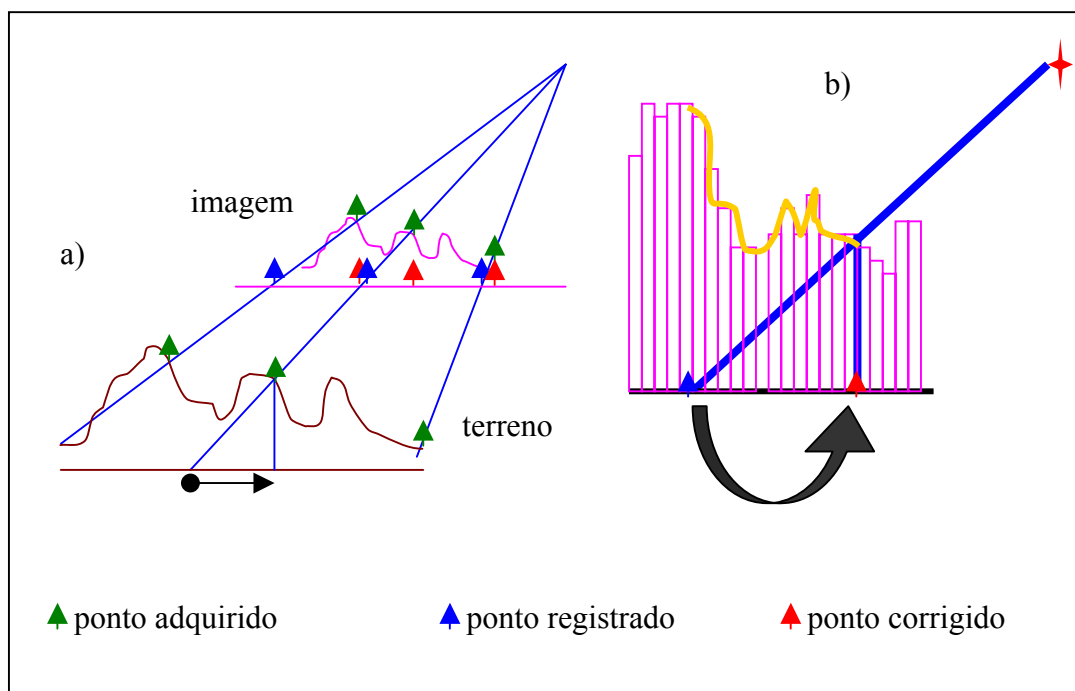


FIGURA 2.1 – Processo de geração de ortofotografias:

- a) geometria de uma linha do sensor: na imagem, os triângulos verdes representam os pontos cujas informações foram adquiridas pelo sensor, os azuis são a posição onde foram registrados e o vermelho, onde deveriam estar;
- b) determinação do deslocamento, pelo processo inverso: pela geometria em (a) determina-se o ponto real.

As ortofotos tradicionais representam exclusivamente o terreno (i.e., excluindo-se os “objetos” sobre o terreno) em sua posição ortogonal. Mas, para as diversas aplicações urbanas, é necessário que se utilize imagens de resolução geométrica suficiente para representar adequadamente os temas urbanos (edificações; rodovias e vias de pedestres; objetos de planejamento urbano, como hidrantes, postes de iluminação, torres, reservatórios; etc).

A fotogrametria convencional, devido às limitações do processo tradicional, definiu a ortofotografia como a imagem fotográfica com as posições de terreno corrigidas. Não era possível, p. ex., eliminar as paredes visíveis nas imagens, portanto a definição tradicional era aceitável. Com o advento do processamento digital de imagens, os elementos de superfície, como edifícios e pontes, que não fazem parte de um MNE, podem também ser retificados à projeção ortogonal, sobre as posições do plano num sistema de projeção cartográfica.

A Figura 2.2 apresenta o solo e uma edificação (a). Ao centro de projeção ("O"), as linhas de visada representadas convergem, resultando na Figura 2.2 (b), onde a parede é representada conforme a legenda. A ortorretificação modifica as posições dos pixels, resultando, finalmente, na Figura 2.2 (c), onde a parede foi completamente removida, o telhado reposicionado, e existe uma região sem informação (área de sombra), chamada, neste trabalho, de área de oclusão.

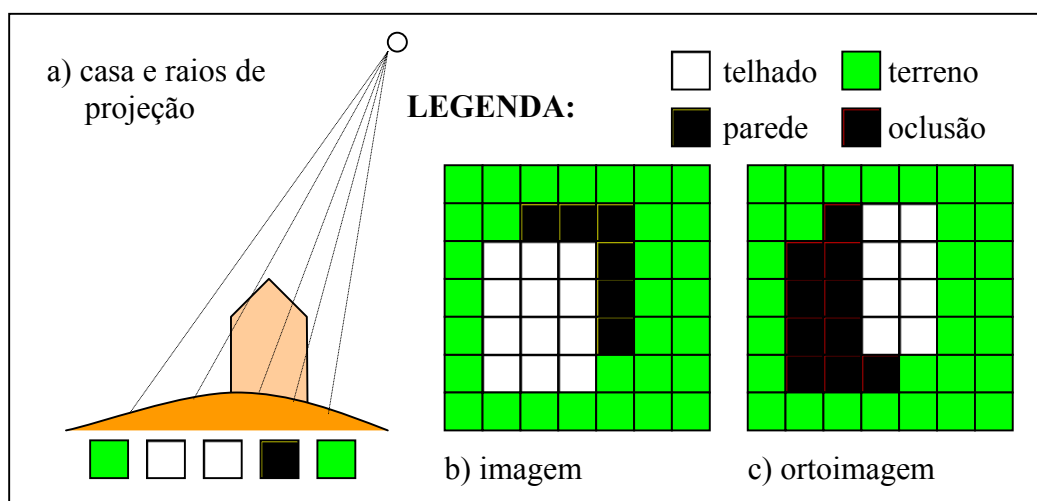


FIGURA 2.2 – Fenômeno físico, imagem e ortofotografia respectivas.

FONTE: Adaptada de Amhar et al. (1998).

Um dos problemas das ortofotografias tradicionais é a não correção das cotas das edificações. Amhar et al. (1998) apresenta uma solução que consiste no uso de dois modelos, um MNE e um modelo numérico de edificações MNEd (*Digital Building Model* – DBM), capazes de gerar ortofotografias com o remapeamento preciso das edificações. A retificação tradicional do terreno é feita e, paralelamente, a retificação apresentada para os telhados. A união algébrica permite a produção da ortofotografia

digital desejada (Amhar e Ecker, 1995). A simplificação desse modelo é apresentada por Boccardo et al. (2001), implementando um sistema que utiliza um MNE denso obtido a partir de *laser scanner* no lugar do MNEd e MNE. Esta simplificação apresenta a viabilidade da mesma consideração feita neste trabalho.

O termo "ortoimagem verdadeira" indica uma correção total das feições representadas na ortoimagem, ou seja, qualquer ponto da imagem deve estar corrigido. No entanto, sempre haverá detalhes de objetos que não estarão mapeados corretamente, como árvores, carros, objetos muito pequenos ou de menor importância. "Deste ponto de vista, ainda permanece uma ficção a ortoimagem verdadeira ideal" (Amhar et al., 1998).

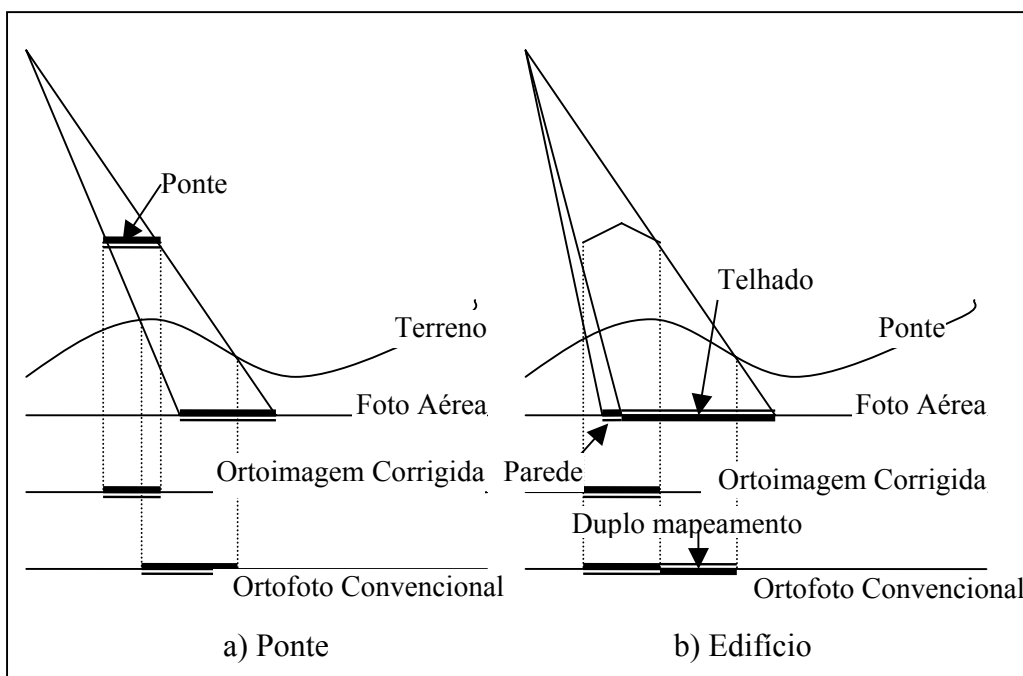


FIGURA 2.3 – Diferença entre ortoimagem corrigida e ortofoto convencional.

A Figura 2.3 apresenta a diferença entre a ortoimagem (ou ortofoto) corrigida e a ortofoto convencional em detalhes. O deslocamento pode ser muito significativo, principalmente nas áreas urbanas, não sendo possível desconsiderá-lo.

Smith (1999) apresenta algumas recomendações em no processo de geração de ortoimagens:

- a revisão de imagens digitais envolve a amostragem de exatidão espacial em campo com GPS: deve-se amostrar pontos distantes de estradas e locais onde pontos do MNE deverão ser tomados;
- para examinar os erros e defeitos das imagens digitais, pode-se utilizar os recursos de *zoom*: obter as imagens digitais e o MNE antes de iniciar a revisão;
- deve-se obter uma cobertura de pontos para os centros das fotografias originais, que possam ser posicionados no topo das imagens digitais, o que permite que o revisor identifique a foto de onde a ortoimagem foi obtida;
- aonde for possível sobrepor a imagem com uma cobertura linear como uma estrada ou uma rede de drenagens de uma outra fonte para ajudar a encontrar erros espaciais;
- estabelecer um critério de aceitação/rejeição antes de iniciar a revisão;
- se linhas de quebra forem geradas para corpos d'água, estradas e rios, estas linhas podem ser usadas para criar coberturas que se ajustarão exatamente com a ortoimagem.

2.3 OCLUSÃO

Para resolver a oclusão, considere-se as Equações da condição de colinearidade (3.5) e (3.6). O efeito do terreno depende da diferença $(Z-Z_0)$ no numerador e denominador das equações. Portanto, se $X=X_0$ e $Y=Y_0$, então Z não tem qualquer efeito sobre o terreno. Afastando-se do ponto (X_0, Y_0) , entretanto, o efeito aumenta.

Pode ocorrer que, por exemplo, uma parte da estrada na imagem não pode ser vista por causa de árvores altas que estão às suas margens.

Sem considerar este efeito, a correção irá aproximar os pixels da árvore que cobriam a estrada para sua posição ortogonal, bastando usar as equações. Entretanto a posição em que a estrada está não poderá ser representada pela própria estrada, pois a informação da estrada não aparece na imagem. Irá ocorrer uma duplicidade de informações na mesma imagem, em que a árvore aparece duas vezes: uma em sua posição ortogonal; outra, ocupando os pixels que deveriam ser preenchidos com informação radiométrica da estrada. Este é, portanto o fenômeno que se deseja eliminar: a duplicidade.

Para eliminar a duplicidade precisa-se colocar alguma outra informação na posição da oclusão. O valor nulo pode ser definido para representar a oclusão. Em termos práticos, deve-se buscar um valor inexistente ou de baixa frequência nas imagens, para que as áreas realmente sem informação sejam apontadas na ortoimagem criada.

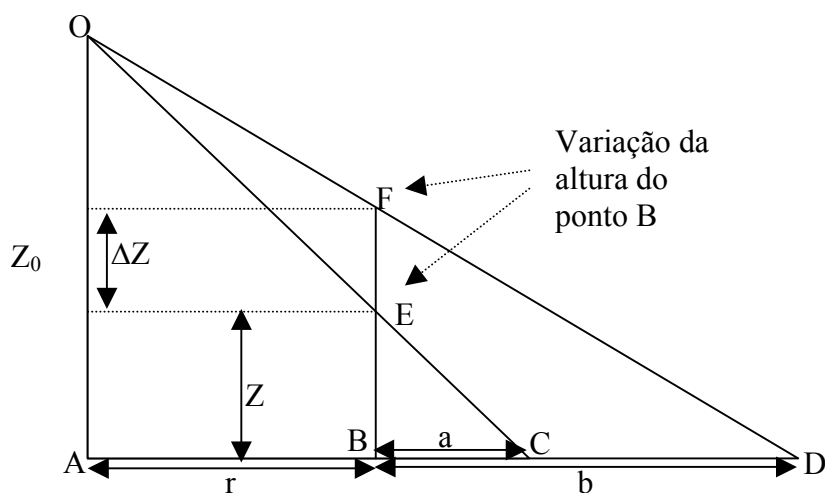


FIGURA 2.4 – Geometria de oclusão. Estudo do efeito do relevo em função da variação da altitude do ponto B, deslocando-se de E até F, deslocando a imagem de C para D.

A solução para o algoritmo modelar a oclusão é a seguinte: considere-se a geometria de aquisição da imagem. Acima da superfície está o sensor, com seu ponto focal definindo o centro de projeção. A vertical, sobre o terreno, do centro de projeção é o ponto (X_0, Y_0) . A paralaxe deste ponto é nula. A paralaxe de um ponto qualquer, -- considerado o raio, i.e., a distância r ao ponto (X_0, Y_0) , -- é diretamente proporcional ao

raio, para uma mesma cota Z . Variando-se o valor de Z o efeito comporta-se conforme representado pela Figura 2.4 e Equação (2.1).

$$\frac{b}{a} = 1 + \left(\frac{\Delta Z}{Z} \right) \left(1 + \frac{Z + \Delta Z}{Z_0 - (Z + \Delta Z)} \right) \quad (2.1)$$

Analisando-se a Equação (2.1), pode-se verificar que quanto maior a altitude da superfície, maior o numerador e menor o denominador da última fração da equação. Portanto o efeito do relevo cresce assintoticamente com o aumento da altitude do ponto, até que o denominador $(Z + \Delta Z)$ atinge o valor de Z_0 , anulando o denominador e fazendo com que b seja infinito. Isso pode ocorrer, p.ex., se uma montanha atinge o nível da câmara (a altitude de vôo).

$$Z + \Delta Z \rightarrow Z_0 \Rightarrow b \rightarrow \infty \quad (2.2)$$

Conhecendo-se os parâmetros envolvidos na geometria, pode-se definir quando um pixel será ou não visível. Para isto, dado um pixel B qualquer sobre a ortomagem, esse pixel representa uma parte da superfície terrestre que, digamos, seja representada por um edifício.

Para determinar que pixels são ocultados pela superfície correspondente ao ponto B , a linha imaginária que une o centro da projeção ao ponto B' , que é definido pelos valores da coordenada de B e sua cota, armazenada no modelo numérico da superfície, ao interceptar o plano de referência, define um ponto C . Assim, B' é mapeado para o ponto C sobre a superfície do plano. O deslocamento r_B representa a distância entre o ponto A e o ponto C . A direção do deslocamento é definida pela direção AB , pois A , B e C são colineares (ver Figura 2.5).

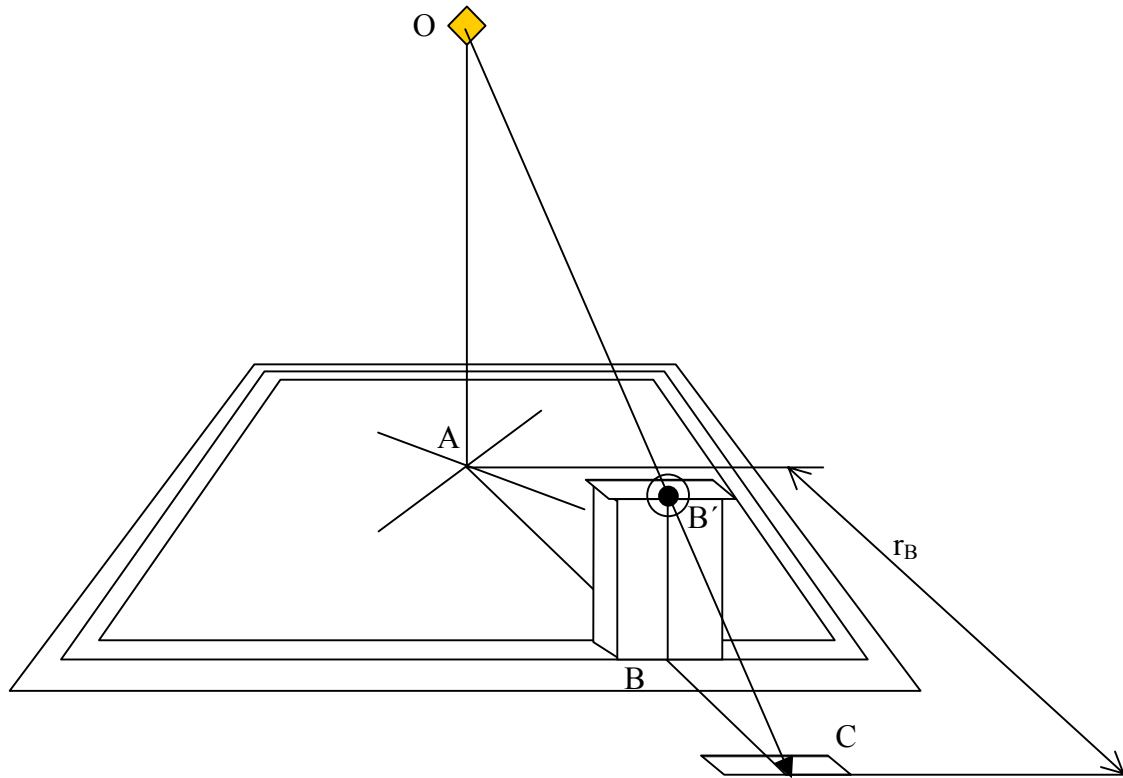


FIGURA 2.5 – Posição da imagem de um pixel. A distância r_B e o vetor AB determinam o pixel C onde é mapeada a imagem de um ponto B' .

Um ponto que esteja sobre a reta AB , portanto, pode não ser visível a O , bastando, para isso, que sua cota represente um ponto interno ao triângulo $BB'C$. Se isso ocorrer para um ponto entre B e C , o ponto está ocluído. Todos os pontos da superfície do espaço-objeto que estiverem sobre a linha OC serão mapeados em C , portanto o primeiro obstáculo será o responsável pela informação armazenada no pixel C .

Ainda, todos os pontos em AB ou mais distantes que r_B mesmo na direção AB e com cota positiva sobre a superfície, não poderão ter sua imagem bloqueada pelo ponto B' .

A solução que resolve esta situação é achar todos os pontos que estão ocultos devido a um mesmo ponto B .

Em uma primeira consideração, pode-se traçar retas que passam por A em direção a cada ponto. Obtém-se um perfil do terreno e faz-se a consideração supracitada.

Entretanto, para isso despende-se um grande esforço computacional que pode ser minimizado conforme metodologia apresentada adiante. Para calcular cada perfil, desta forma, é necessário que: cada ponto seja identificado; seja definida a reta que o une ao ponto central; obter, para cada ponto da linha, o correspondente valor de elevação; testar um-a-um os pontos que estão entre o ponto escolhido e o ponto central. Se algum ponto X o encobre (Figura 2.6), o ponto B está oculto.

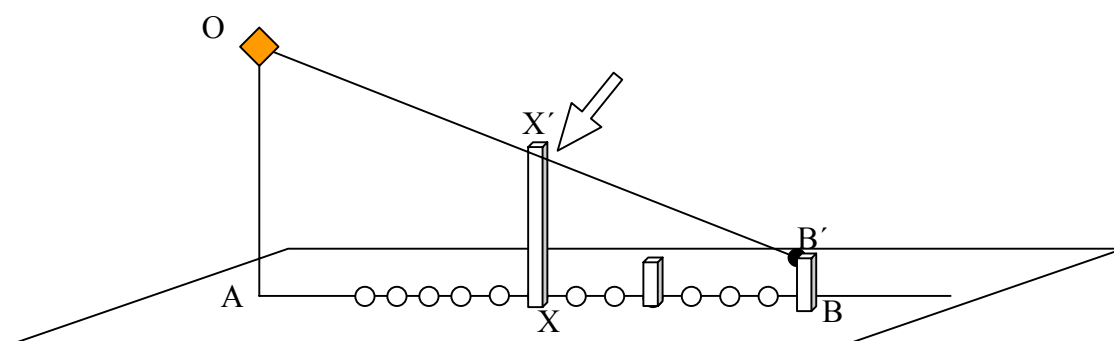


FIGURA 2.6 – Perfil da geometria de oclusão. Um ponto B' tem o ponto X' como obstáculo, e a sua imagem não seria representada.

2.4 EFEITOS DE OCLUSÃO E ILUMINAÇÃO

Um dos maiores desafios na produção de ortoimagens em áreas urbanas é o fato que os objetos elevados sobre o terreno apresentam-se deslocados de suas posições e tornam-se obstáculo para a visibilidade de partes da imagem. Visualmente, esse efeito proporciona a mesma aparência da imagem não-ortorretificada após a ortorretificação.

Como exemplo, um muro deveria aparecer numa imagem ortorretificada como aparece numa carta: uma linha estreita, representando a superfície superior do muro (o que só ocorre se o seu prolongamento intercepta a vertical do centro de perspectiva). Mas o que se tem com um MNE convencional é uma “inclinação” do muro, radialmente ao eixo óptico do sistema sensor.

Quando a informação de parte da superfície não está presente na imagem, precisa eliminada da ortoimagem, o que é feito utilizando-se uma única imagem de uma cena

para ser ortorretificada, um modelo numérico da superfície e pontos com coordenadas do espaço-objeto (terreno) conhecidas (pontos de controle).

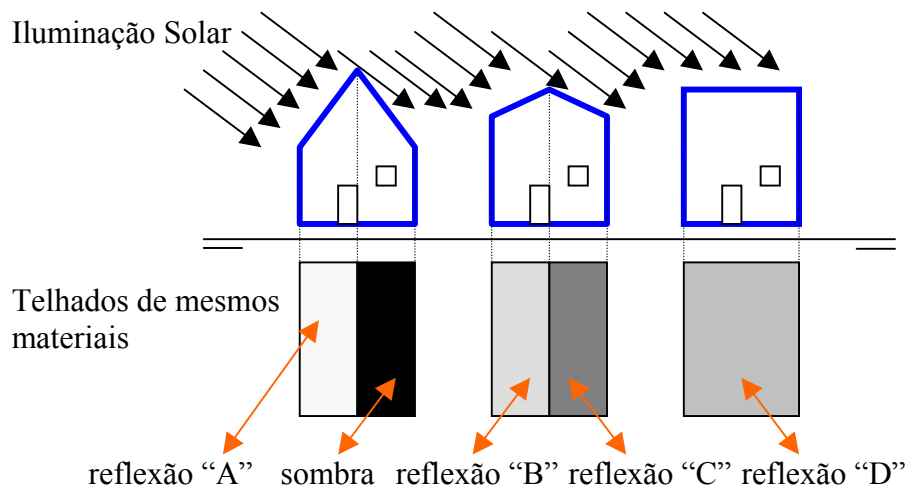


FIGURA 2.7 - Telhados com superfície não-lambertiana. Quando submetidos a iluminação sob diferentes ângulos de incidência ou reflexão, refletem a radiação com taxas também diferentes.

Outro aspecto negativo que faz uma ortoimagem atual diferir de uma carta topográfica é o sombreamento. A iluminação solar faz com que a variação da intensidade de um mesmo alvo não seja resultado somente das propriedades físico-químicas das superfícies, mas também das características espaciais (orientação, declividade, sombras etc). Rau et al. (2000) fazem a proposta de um estudo para minimizar esse tipo de problema.

Para exemplificar que o fenômeno da iluminação não se restringe somente a sombras, o telhado apresentado na Figura 2.7 apresenta os raios solares em feixes paralelos sobre casas cuja cobertura apresenta materiais idênticos. A vista de topo (parte de baixo) é a representação ortogonal das casas e apresenta um tom de cinza diferenciado de acordo com o caso, representando as diferentes reflectâncias para os mesmos materiais e ângulo de incidência da iluminação.

Similarmente pode-se considerar um outro caso em que a superfície é anisotrópica. Pode-se considerar um telhado, que geralmente apresenta as telhas geometricamente

assimétricas, o que indica que uma anisotropia pode ser detectada de acordo com as condições e sensibilidade do sensor.

Embora a iluminação seja também uma característica que pode ser melhorada nas ortofotografias, este trabalho manterá o objetivo de resolver somente o aspecto geométrico da ortorretificação, deixando para trabalhos futuros este aspecto radiométrico.

2.5 GEOMETRIA DE IMAGEAMENTO

Este trabalho utiliza uma abordagem aerofotogramétrica ao considerar que a câmara deve estar posicionada verticalmente, de modo que os ângulos de inclinação do plano da imagem são limitados. O ângulo de rotação do plano em torno de seu eixo normal não tem limitação, pois uma imagem bruta pode não estar orientada para o norte e, mesmo assim, os pontos podem ser identificados e a transformação equacionada. Além disso, pontos na imagem podem ser identificados e medidos em campo.

Nas outras aplicações, desde a fotogrametria terrestre até a visão computacional, seriam indesejáveis tais limitações angulares, pois restringem a livre inclinação espacial do modelo. Mas as restrições deste trabalho não invalidam o modelo em estudo, pois as imagens utilizadas em sensoriamento remoto são compatíveis com a condição de verticalidade, e o modelo permite determinar os ângulos em relação ao modelo perfeitamente vertical.

O tema deste trabalho torna-se especialmente importante quando uma imagem terrestre, obtida por um sensor com eixo óptico vertical (não oblíquo), apresenta uma geometria que não pode ser utilizada diretamente em determinadas aplicações urbanas por necessitarem de precisão.

Como consideramos neste trabalho imagens obtidas por câmaras a bordo de aeronaves, a geometria do imageamento será definida por uma projeção perspectiva. No caso de imagens de satélite, obtidas por varredura, a geometria de imageamento difere muito da projeção perspectiva até agora apresentada, sendo mais próxima a uma projeção cilindro-cônica, onde o foco deixa de ser um único ponto e passa a ser uma seqüência de pontos.

No cálculo da geometria de imageamento, o trabalho utiliza as equações da condição de colinearidade, que são equações referentes a um sistema de coordenadas local. A transformação para um sistema de projeção cartográfica é uma questão importante pois um usuário final poderá não desejar dispor de um sistema local de coordenadas, mas de um sistema de projeção cartográfica.

2.6 CALIBRAÇÃO E ORIENTAÇÃO

Uma parte fundamental do processo de geração da ortoimagem refere-se à determinação dos parâmetros do imageamento, que envolvem a calibração da câmara e a orientação interior. Neste particular, há uma diferença importante entre as câmaras fotogramétricas métricas, semi-métricas ou não-métricas. As câmaras métricas apresentam a geometria mais estável, com maior garantia de perpendicularidade dos eixos, imagem mais próxima de um plano por maior aderência do filme, menores distorções e aberrações dos sistemas de lentes e distância da câmara igual à distância focal podem ter distâncias focais variáveis ou não medidas.

Sendo x_i a distância da imagem e x_o a distância do objeto, vale a relação $\frac{1}{x_i} = \frac{1}{f} + \frac{1}{x_o}$.

Para a fotografia aérea, como $x_o \gg f$, pode-se considerar que $x_o \rightarrow \infty$ quando $x_i = x_c$, e tem-se:

$$\frac{1}{c} = \frac{1}{f} + \frac{1}{\infty} \Rightarrow c = f$$

A maior desvantagem das câmaras não-métricas são os erros sistemáticos significativos (Deng, 1999), cujas três causas principais são:

- a) distorções das lentes;
- b) deformações dos meios de gravação; e
- c) imperfeições nos mecanismos de medição de coordenadas (como erros de comprimento e não- perpendicularidade de eixos).

Neste trabalho são considerados somente a distância principal (aqui utilizada sem distinção da distância focal) do sistema sensor em que a imagem é formada por quadros e o centro da imagem. O sistema fotográfico forma a imagem por quadros por definir toda a área da cena imageada em um único intervalo de tempo.

O centro “exato” da imagem geralmente é obtido pelo “erro” das marcas fiduciais. Não serão levados em conta os demais parâmetros de calibração, como as constantes de distorções das lentes e os gráficos ou tabelas de aberrações. Para o ajustamento, o algoritmo apresentado considera cada conjunto de parâmetros obtidos naquela foto, ainda que sejam obtidos de uma mesma câmara. Deng (1999) considera este o procedimento adequado de se tratar imagens não-métricas.

Foi desenvolvido um procedimento para a determinação conjunta da orientação interna (calibração) e externa (orientação absoluta). Consiste num desenvolvimento análogo aos apresentados na bibliografia, sendo que são considerados 9 parâmetros de orientação e buscou-se uma solução algébrica direta, conforme desenvolvido por Wolf (1974), sem utilizar a aproximação utilizada na linearização de equações pelo uso do teorema de Taylor.

A abordagem da linearização por Taylor consiste em considerar um valor inicial para cada parâmetro e utilizar equações de correções das aproximações, enquanto neste trabalho os valores são obtidos diretamente a partir das equações deduzidas no Capítulo 3.

CAPÍTULO 3

METODOLOGIA

3.1 DADOS EM PROJEÇÃO CARTOGRÁFICA

As formulações apresentadas neste capítulo consideram um sistema dextrógiro de coordenadas tri-ortogonais. Dados em projeção podem ser utilizados, mas necessitam de mudança de sistema, que é realizada por uma transformação das coordenadas de cada ponto de controle e dos pontos do modelo numérico da superfície ao sistema local.

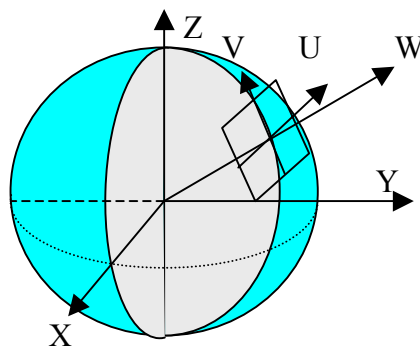


FIGURA 3.1 – Transformação de coordenadas para um sistema local de coordenadas.

Conhecendo-se as coordenadas em um sistema qualquer de projeção, pode-se convertê-las em coordenadas cartesianas geocêntricas. Considerando o norte como sendo o eixo Z, pode-se utilizar o procedimento a seguir descrito para preparar as informações para o formato exigido pela formulação apresentada neste capítulo.

Um cuidado a ser tomado, entretanto, é a homogeneidade dos sistemas de referência estabelecidos. As coordenadas dos pontos de controle devem estar no mesmo sistema de representação (de projeção ou cartesiano) que as coordenadas do modelo numérico da superfície, que determinarão as coordenadas da imagem resultante.

Para determinar-se um sistema local a partir de um sistema de projeção cartográfica como, por exemplo, o UTM, sugere-se o uso da seguinte transformação:

$$(UTM) \Rightarrow (geográficas) \Rightarrow [geocêntricas cartesianas] \Rightarrow [cartesianas locais]$$

Primeiramente as coordenadas do sistema de projeção UTM (E, N, h) são transformadas em coordenadas geográficas (ϕ, λ, h) do mesmo datum-elipsoidal; em seguida, transformadas em coordenadas cartesianas geocêntricas; por fim, transformadas para o novo sistema local.

3.2 CALIBRAÇÃO E ORIENTAÇÃO

Antes de realizar a ortorretificação, deve-se determinar os parâmetros de orientação interior e exterior, que são utilizados nas equações.

O desenvolvimento apresentado a seguir consiste toma como base as equações da condição de colinearidade, havendo algumas considerações a respeito dos sistemas de coordenadas utilizadas. A obtenção dos parâmetros faz com que se defina um conjunto de matrizes e, assim, o desenvolvimento das equações necessárias à obtenção dos parâmetros de calibração é apresentado.

3.2.1 CONSIDERAÇÕES SOBRE A ABORDAGEM UTILIZADA

A abordagem clássica minimiza, iterativamente, uma função erro não-linear (Heikkilä e Silvén, 1997). A utilizada é semelhante à transformação linear direta (DLT) apresentada por Abdel-Aziz e Karara (1974), embora o desenvolvimento siga as equações e convenções utilizadas por Kraus (1993).

O maior problema da DLT, segundo Hatze (1980), citado por Kwon (1998) é que os parâmetros da calibração obtidos não são mutuamente independentes e que isto compromete a ortogonalidade da matriz de rotação, o que não ocorre na presente abordagem em relação aos ângulos, que são ortogonais por imposição.

Além da DLT, outros métodos são apresentados na literatura, como Triggs (1999), Quan e Lan (1999) e Kolesnik (2001). Armangué (2000) descreve e avalia 5 métodos quanto à exatidão.

Este trabalho, entretanto, não consiste em uma simplificação ou em uma variante de outro método específico, embora possa, matematicamente, apresentar constantes ou variáveis iguais, semelhantes ou distintas.

3.2.2 SISTEMAS DE COORDENADAS UTILIZADOS

O valor das coordenadas referentes a um sistema de projeção qualquer deve ser transformado em sistema local tri-ortogonal, em que um dos eixos esteja sobre a direção definida pelo eixo que liga o centro da Terra ao centro de projeção (ou a um ponto próximo), para garantir que algumas condições mantenham-se conforme desenvolvidas nas equações.

Se os pontos estiverem em projeção UTM e os parâmetros forem calculados diretamente (como se fosse sistema local), quanto mais próximos os pontos estiverem do fuso central, mais preciso é o resultado. O algoritmo realiza a translação no plano (XY) , ou especificamente neste caso, no plano (EN) , para que a origem arbitrária seja transladada até a média geométrica dos valores das coordenadas.

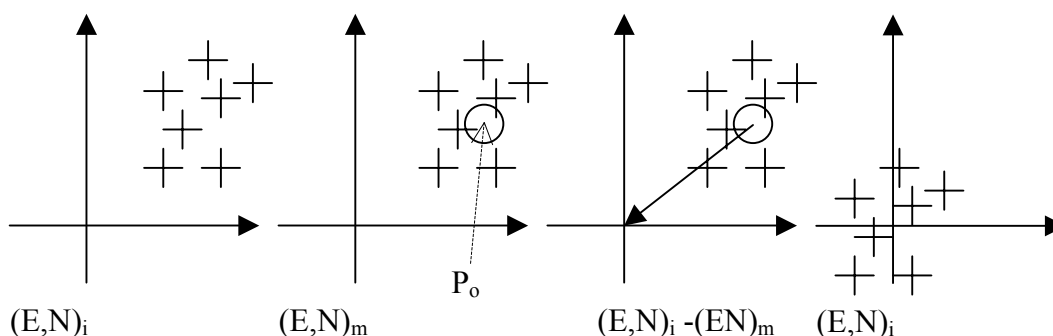


FIGURA 3.2 - Representação gráfica da translação da Equação (3.5).

$$(X_0, Y_0, 0) \approx (\bar{E}, \bar{N}, 0) = \left(\frac{\sum_{i=1}^n E_i}{n}, \frac{\sum_{i=1}^n N_i}{n}, 0 \right) \quad (3.1)$$

A seqüência da Figura 3.2 apresenta o plano (EN) representado em 4 gráficos. Cada gráfico apresenta os eixos X (E) e Y (N), e 7 pontos. Em seguida calcula-se o valor de P_0 , ou seja, o ponto (X_0, Y_0) . No terceiro gráfico, faz-se a translação pela média. No último os pontos estão transladados e apresentam a média $(0,0)$.

Entretanto, como refinamento, pode-se utilizar os valores das coordenadas da projeção como se fossem coordenadas locais, para que se obtenha um valor aproximado de P_0 , e em seguida, utilizar-se o procedimento apresentado em (3.1) para que se determine um

conjunto de pontos (de controle e para gerar o modelo numérico de elevação) num sistema local de coordenadas.

Obtém-se o valor da coordenada do centro da imagem na projeção cartográfica utilizada, ou pode-se calcular a média aritmética de cada coordenada para todos os pontos de controle. O importante é que se tenha um ponto interior que sirva como aproximação inicial da posição XY do centro de projeção, que então será efetivamente calculado.

Utiliza-se esse valor inicial projetado ortogonalmente no próprio plano de projeção, i.e., sobre o plano XY , definindo-se, assim, o ponto $P_{0XY}(X_0, Y_0, 0)$.

Particularmente no caso das coordenadas dos pontos de controle estarem no sistema de projeção UTM, utilizando-se a notação (E, N, h) , utilizam-se, inicialmente, os valores $P_i(E_i, N_i, h_i)$ dos pontos de controle. Pode-se proceder de duas maneiras: a obter um valor inicial aproximado para $P_0(E_0, N_0, 0)$ e realizar a transformação apresentada em (3.1); ou utilizar as coordenadas UTM como se fossem coordenadas locais.

3.2.3 AS MATRIZES PARA O AJUSTAMENTO

As equações da condição de colinearidade (Kraus, 1993) definem o par de coordenadas de imagem em função dos seguintes parâmetros de orientação:

- a) da distância focal (c);
- b) das coordenadas de imagem da projeção vertical do centro de projeção sobre a imagem (ξ_0, η_0) ;
- c) da matriz de rotação $\mathbf{R}_{\omega\phi\kappa}$, que é definida pelos três ângulos ω , ϕ , κ , e que representam a rotação espacial do plano da imagem em torno de seu centro focal, tendo o sistema cartesiano local de coordenadas do terreno (plano cartesiano XYZ) como referência; e, ainda,
- d) das coordenadas (X_0, Y_0, Z_0) da projeção vertical do centro de projeção sobre o plano cartesiano XYZ .

A matriz de rotação de um terno cartesiano dextrógiro e as equações da condição de colinearidade podem ser escritas (Kraus, 1993):

$$R_{\omega\phi\kappa} = \begin{bmatrix} \cos\varphi.\cos\kappa & -\cos\varphi.\sen\kappa & \sen\varphi \\ \cos\omega.\sen\kappa + \sen\omega.\sen\varphi.\cos\kappa & \cos\omega.\cos\kappa - \sen\omega.\sen\varphi.\sen\kappa & -\sen\omega.\cos\varphi \\ \sen\omega.\sen\kappa - \cos\omega.\sen\varphi.\cos\kappa & \sen\omega.\cos\kappa + \cos\omega.\sen\varphi.\sen\kappa & \cos\omega.\cos\varphi \end{bmatrix} \quad (3.2)$$

$$R_Z(\kappa).R_Y(\phi).R_X(\omega) = R_{\omega\phi\kappa} \quad (3.3)$$

O ponto do espaço-objeto, sua imagem e o centro de projeção são colineares (Andrade e Olivas, 1981):

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = k.R_Z(\kappa).R_Y(\phi).R_X(\omega). \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \quad (3.4)$$

Sendo ξ_0 e η_0 a ordenada e a abscissa do ponto principal, respectivamente, as equações da condição de colinearidade são:

$$\xi = \xi_0 - c. \frac{r_{11}(X - X_0) + r_{21}(Y - Y_0) + r_{31}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \quad (3.5)$$

$$\eta = \eta_0 - c. \frac{r_{12}(X - X_0) + r_{22}(Y - Y_0) + r_{32}(Z - Z_0)}{r_{13}(X - X_0) + r_{23}(Y - Y_0) + r_{33}(Z - Z_0)} \quad (3.6)$$

Onde tem-se:

$$R_{\omega\phi\kappa} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (3.7)$$

$$(R_{\omega\phi\kappa})^{-1} = (R_{\omega\phi\kappa})^T = \begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{bmatrix} \quad (3.8)$$

e as propriedades da matriz de rotação:

$$r_{13}^2 + r_{23}^2 + r_{33}^2 = 1 \quad (3.9)$$

$$r_{11} \cdot r_{13} + r_{21} \cdot r_{23} + r_{31} \cdot r_{33} = 0 \quad (3.10)$$

$$r_{12} \cdot r_{13} + r_{22} \cdot r_{23} + r_{32} \cdot r_{33} = 0 \quad (3.11)$$

resultando nas relações:

$$\operatorname{tg} \omega = -\frac{r_{23}}{r_{33}} \Rightarrow \omega = \arctan\left(-\frac{r_{23}}{r_{33}}\right) \quad (3.12)$$

$$\operatorname{sen} \varphi = r_{13} \Rightarrow \varphi = \operatorname{arcsen}(r_{13}) \quad (3.13)$$

$$\tan \kappa = -\frac{r_{12}}{r_{11}} \Rightarrow \kappa = \arctan\left(-\frac{r_{12}}{r_{11}}\right) \quad (3.14)$$

Como os ângulos φ e ω , para imagens aéreas verticais não podem atingir valores iguais ou superiores a 90° , as equações apresentam uma única solução no intervalo

$$\varphi, \omega \in \left] -\frac{\pi}{2}, \frac{\pi}{2} \right]$$

Como seus valores não são conhecidos, os parâmetros das nossas equações são os nove seguintes: ω , φ , κ , X_0 , Y_0 , Z_0 , c , ξ_0 , η_0 . Nove parâmetros exigem um mínimo de 9 (nove) equações, se for possível linearizá-las.

Entretanto, as equações não são lineares, e os 9 (nove) parâmetros produzem outras 11 (onze) incógnitas: A , B , C , D_1 , D_2 , E_1 , E_2 , F_1 , F_2 , G_1 , G_2 . Determinando-se estas incógnitas, os parâmetros de orientação ficam determinados, conforme as Equações (3.33) a (3.52).

$$\xi = \xi X A + \xi Y B + \xi Z C + X D_1 + Y E_1 + Z F_1 + G_1 \quad (3.15)$$

$$\eta = \eta X A + \eta Y B + \eta Z C + X D_2 + Y E_2 + Z F_2 + G_2 \quad (3.16)$$

Para resolver a equação acima, necessitamos de 11 (onze) equações. Entretanto, cada ponto forma duas equações, e podem ser utilizados 6 (seis) ou mais pontos. O método dos mínimos quadrados é representada pela Equação (3.17) (Destri, 1987; Cooper e Cross, 1988; Cooper e Cross, 1991):

$$\mathbf{X} = (\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \mathbf{L}_b \quad (3.17)$$

onde as matrizes correspondentes, para um exemplo em que 7 (sete) pontos de controle são utilizados, seguem conforme as equações seguintes.

$$\mathbf{A} = \begin{bmatrix} \xi_1 X_1 & \xi_1 Y_1 & \xi_1 Z_1 & X_1 & 0 & Y_1 & 0 & Z_1 & 0 & 1 & 0 \\ \eta_1 X_1 & \eta_1 Y_1 & \eta_1 Z_1 & 0 & X_1 & 0 & Y_1 & 0 & Z_1 & 0 & 1 \\ \xi_2 X_2 & \xi_2 Y_2 & \xi_2 Z_2 & X_2 & 0 & Y_2 & 0 & Z_2 & 0 & 1 & 0 \\ \eta_2 X_2 & \eta_2 Y_2 & \eta_2 Z_2 & 0 & X_2 & 0 & Y_2 & 0 & Z_2 & 0 & 1 \\ \xi_3 X_3 & \xi_3 Y_3 & \xi_3 Z_3 & X_3 & 0 & Y_3 & 0 & Z_3 & 0 & 1 & 0 \\ \eta_3 X_3 & \eta_3 Y_3 & \eta_3 Z_3 & 0 & X_3 & 0 & Y_3 & 0 & Z_3 & 0 & 1 \\ \xi_4 X_4 & \xi_4 Y_4 & \xi_4 Z_4 & X_4 & 0 & Y_4 & 0 & Z_4 & 0 & 1 & 0 \\ \eta_4 X_4 & \eta_4 Y_4 & \eta_4 Z_4 & 0 & X_4 & 0 & Y_4 & 0 & Z_4 & 0 & 1 \\ \xi_5 X_5 & \xi_5 Y_5 & \xi_5 Z_5 & X_5 & 0 & Y_5 & 0 & Z_5 & 0 & 1 & 0 \\ \eta_5 X_5 & \eta_5 Y_5 & \eta_5 Z_5 & 0 & X_5 & 0 & Y_5 & 0 & Z_5 & 0 & 1 \\ \xi_6 X_6 & \xi_6 Y_6 & \xi_6 Z_6 & X_6 & 0 & Y_6 & 0 & Z_6 & 0 & 1 & 0 \\ \eta_6 X_6 & \eta_6 Y_6 & \eta_6 Z_6 & 0 & X_6 & 0 & Y_6 & 0 & Z_6 & 0 & 1 \\ \xi_7 X_7 & \xi_7 Y_7 & \xi_7 Z_7 & X_7 & 0 & Y_7 & 0 & Z_7 & 0 & 1 & 0 \\ \eta_7 X_7 & \eta_7 Y_7 & \eta_7 Z_7 & 0 & X_7 & 0 & Y_7 & 0 & Z_7 & 0 & 1 \end{bmatrix} \quad (3.18)$$

A matriz de pesos \mathbf{W} tem a seguinte forma:

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ w_{2,1} & w_{2,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_{3,3} & w_{3,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_{4,3} & w_{4,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{5,5} & w_{5,6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{6,5} & w_{6,6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & w_{7,7} & w_{7,8} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & w_{8,7} & w_{8,8} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{9,9} & w_{9,10} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{10,9} & w_{10,10} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{11,11} & w_{11,12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{12,11} & w_{12,12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{13,13} & w_{13,14} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w_{14,13} & w_{14,14} \end{bmatrix} \quad (3.19)$$

A matriz de pesos \mathbf{W} representa os pesos de 7 pontos sem qualquer correlação entre si, entretanto podendo haver correlação entre os valores das coordenadas de um mesmo ponto. Apresenta dois valores para cada coordenada (linha ou coluna), e w_{11} é o peso da coordenada de imagem ξ_1 , enquanto $w_{12} = w_{21}$ é o peso da relação entre ξ_1 e η_1 , representando a influência do valor de ξ_1 em relação ao valor obtido por η_1 .

Neste trabalho foi considerado que os valores de abscissa e ordenada de um mesmo ponto são independentes e, portanto, não há correlação entre elas. Isso faz com que a matriz \mathbf{W} seja uma matriz diagonal.

A matriz de observações representa os pontos obtidos na imagem. Os valores de coordenadas do espaço-objeto (do terreno) são considerados valores corretos, sendo que as imprecisões consistem na identificação imprecisa ou incorreta das coordenadas de imagem, e por isso o peso torna-se necessário.

Os pontos de controle de campo, quando medidos por levantamento de precisão, referem-se a pontos de fácil identificação na imagem. A baixa incerteza associada a esses pontos é representada com um valor alto de peso.

A matriz dos parâmetros \mathbf{B} e a das observações \mathbf{L}_b , de modo que $\mathbf{AB}=\mathbf{L}$, podem ser representadas da seguinte maneira:

$$\mathbf{B} = \begin{bmatrix} A \\ B \\ C \\ D_1 \\ D_2 \\ E_1 \\ E_2 \\ F_1 \\ F_2 \\ G_1 \\ G_2 \end{bmatrix} \quad \text{e} \quad \mathbf{L}_b = \begin{bmatrix} \xi_1 \\ \eta_1 \\ \xi_2 \\ \eta_2 \\ \xi_3 \\ \eta_3 \\ \xi_4 \\ \eta_4 \\ \xi_5 \\ \eta_5 \\ \xi_6 \\ \eta_6 \\ \xi_7 \\ \eta_7 \end{bmatrix} \quad (3.20)$$

A partir das matrizes assim definidas, a aplicação do método dos mínimos quadrados irá garantir a minimização dos resíduos. A partir de então, estarão definidos os valores desses novos parâmetros. A relação entre os novos parâmetros (aqueles da matriz \mathbf{B}) e os parâmetros de orientação interior e exterior são apresentados a seguir.

3.2.4 ORIENTAÇÃO INTERIOR E EXTERIOR

Uma vez calculados os valores dos parâmetros das Equações (3.15) e (3.16), define-se a variável auxiliar d na Equação (3.21):

$$d = -(r_{13}X_0 + r_{23}Y_0 + r_{33}Z_0) \quad (3.21)$$

Desenvolvendo-se algebricamente a equação as constantes:

$$A = -\frac{r_{13}}{d} \quad (3.22)$$

$$B = -\frac{r_{23}}{d} \quad (3.23)$$

$$C = -\frac{r_{33}}{d} \quad (3.24)$$

$$D_1 = \left(\frac{r_{13}}{d} \xi_0 - c \frac{r_{11}}{d} \right) \quad (3.25)$$

$$D_2 = \left(\frac{r_{13}}{d} \eta_0 - c \frac{r_{12}}{d} \right) \quad (3.26)$$

$$E_1 = \left(\frac{r_{23}}{d} \xi_0 - c \frac{r_{21}}{d} \right) \quad (3.27)$$

$$E_2 = \left(\frac{r_{23}}{d} \eta_0 - c \frac{r_{22}}{d} \right) \quad (3.28)$$

$$F_1 = \left(\frac{r_{33}}{d} \xi_0 - c \frac{r_{31}}{d} \right) \quad (3.29)$$

$$F_2 = \left(\frac{r_{33}}{d} \eta_0 - c \frac{r_{32}}{d} \right) \quad (3.30)$$

$$G_1 = \xi_0 + \frac{c}{d} (r_{11}X_0 + r_{21}Y_0 + r_{31}Z_0) \quad (3.31)$$

$$G_2 = \eta_0 + \frac{c}{d} (r_{12}X_0 + r_{22}Y_0 + r_{32}Z_0) \quad (3.32)$$

Utilizando-se (3.12), (3.23) e (3.24), tem-se:

$$\omega = \arctan\left(-\frac{B}{C}\right) \quad (3.33)$$

Utilizando-se (3.9), (3.22), (3.23) e (3.24), obtém-se:

$$d = \pm \left(\frac{1}{\sqrt{A^2 + B^2 + C^2}} \right) \quad (3.34)$$

Para analisar o sinal de (3.34), com base em (3.21), ainda considerando-se que o sistema de coordenadas cartesianas local terá valores baixos de X_0 e Y_0 e que os ângulos de atitude referentes ao vôo vertical são pequenos, o sinal de d – definido na Equação (3.21) – será negativo e tem-se:

$$d = - \left(\frac{1}{\sqrt{A^2 + B^2 + C^2}} \right) \quad (3.35)$$

Utilizando-se (3.13), (3.22) e (3.35), obtém-se:

$$\varphi = \arcsen\left(\frac{A}{\sqrt{A^2 + B^2 + C^2}}\right) \quad (3.36)$$

Utilizando-se (3.22), (3.23), (3.24), (3.25), (3.27) e (3.29), obtém-se:

$$\frac{d^2}{c} [(D_1 + A\xi_0)A + (E_1 + B\xi_0)B + (F_1 + C\xi_0)C] = 0 \quad (3.37)$$

Utilizando-se (3.22), (3.23), (3.24), (3.26), (3.28) e (3.30), obtém-se:

$$\frac{d^2}{c} [(D_2 + A\eta_0)A + (E_2 + B\eta_0)B + (F_2 + C\eta_0)C] = 0 \quad (3.38)$$

A condição de denominador não nulo está garantida, pois o valor de c não é zero e impossível que d seja nulo (ver a definição de A , B , C e a matriz de rotação), tem-se:

$$c \neq 0 \quad \text{e} \quad d \neq 0$$

$$\therefore \xi_0 = -\left(\frac{AD_1 + BE_1 + CF_1}{A^2 + B^2 + C^2}\right) \quad (3.39)$$

$$\therefore \eta_0 = -\left(\frac{AD_2 + BE_2 + CF_2}{A^2 + B^2 + C^2}\right) \quad (3.40)$$

Os valores de ξ_0 e η_0 conforme as Equações (3.39) e (3.40) são assim calculados. A exatidão dos valores depende diretamente da qualidade dos pontos observados e medidos no terreno.

Conhecendo-se ξ_0 e η_0 , (3.2), (3.22), (3.25) e (3.26), obtém-se:

$$\kappa = \arctan\left(-\frac{D_2 + A\eta_0}{D_1 + A\eta_0}\right) \quad (3.41)$$

De (3.2), (3.22) e (3.25), resulta:

$$c = \left(-d \frac{A\xi_0 + D_1}{r_{11}}\right) \quad (3.42)$$

E analogamente, pode ser medido utilizando-se (3.26), (3.27), (3.28), (3.29) e (3.30) respectivamente, tem-se:

$$c = \left(-d \frac{A\eta_0 + D_2}{r_{12}}\right) \quad (3.43)$$

$$c = \left(-d \frac{B\xi_0 + E_1}{r_{21}}\right) \quad (3.44)$$

$$c = \left(-d \frac{B\eta_0 + E_2}{r_{22}}\right) \quad (3.45)$$

$$\mathbf{c} = \left(-d \frac{C\xi_0 + F_1}{r_{31}} \right) \quad (3.46)$$

$$\mathbf{c} = \left(-d \frac{C\eta_0 + F_2}{r_{32}} \right) \quad (3.47)$$

Obtendo-se o mesmo parâmetro por 6 equações diferentes, pode-se minimizar o erro utilizando-se a média aritmética. Resulta, portanto, na Equação:

$$\mathbf{c} = -\frac{d}{6} \left(\frac{A\xi_0 + D_1}{r_{11}} + \frac{A\eta_0 + D_2}{r_{12}} + \frac{B\xi_0 + E_1}{r_{21}} + \frac{B\eta_0 + E_2}{r_{22}} + \frac{C\xi_0 + F_1}{r_{31}} + \frac{C\eta_0 + F_2}{r_{32}} \right) \quad (3.48)$$

Faltando, agora, determinar X_0 , Y_0 , Z_0 . Utilizando-se (3.35), (3.31) e (3.32), chega-se a uma equação da forma:

$$\mathbf{R}^T \cdot \mathbf{X}_0 = \mathbf{S} \quad (3.49)$$

Onde \mathbf{R}^T vem da Equação (3.8), e:

$$\mathbf{X}_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} \quad (3.50)$$

$$\mathbf{S} = \begin{bmatrix} \frac{d}{c}(G_1 - \xi_0) \\ \frac{d}{c}(G_2 - \eta_0) \\ -d \end{bmatrix} \quad (3.51)$$

A solução de (3.49), considerando (3.8) é:

$$\mathbf{R}^T \cdot \mathbf{X}_0 = \mathbf{S} \Rightarrow \mathbf{R}^{-1} \cdot \mathbf{X}_0 = \mathbf{S}$$

$$\therefore \mathbf{X}_0 = \mathbf{R} \cdot \mathbf{S} \quad (3.52)$$

Deste modo os parâmetros de orientação são obtidos.

3.3 ORTORRETIFICAÇÃO

A ortorretificação, conforme foi visto anteriormente, tem o mesmo fundamento: um feixe de luz, após interagir (refletir ou refratar) com o alvo (terreno), atinge o centro de perspectiva e passa exatamente pelo plano da imagem, definindo o pixel que foi mapeado (equações da condição de colinearidade).

Para corrigir, então, a elevação da superfície, precisamos definir um modelo numérico relativo às cotas dos pontos sobre a superfície do terreno com certas características, para que se defina uma metodologia de retificação.

Definido-se o modelo numérico da superfície do terreno como sendo obrigatoriamente quadrangular regular e cujo elemento central corresponde exatamente à projeção ortogonal do centro de perspectiva, temos o desenvolvimento a seguir. Para isso, é necessário, primeiramente, calcular a matriz \mathbf{X}_0 pela Equação (3.52).

Diversos pixels da ortoimagem referem-se a áreas que estão escondidas atrás de edifícios, pontes, montanhas etc. Esses pixels não podem apresentar qualquer valor de imagem, portanto, arbitra-se um valor para a ausência de informação, e realiza-se a correção de cada pixel. Ocorrendo uma oclusão, o pixel receberá aquele valor de fundo.

A ortorretificação nada mais é do que a aplicação da Equação (3.5) e (3.6) após conhecidos todos os parâmetros.

Conhecendo-se os parâmetros, para cada ponto (X, Y, Z) em coordenadas de espaço-objeto, obtém-se o ponto correspondente em coordenadas de imagem, incluindo a paralaxe resultante do valor de Z . A correção consiste em identificar a posição em que o ponto aparece na imagem e, com base no próprio valor de (X, Y) , a posição em que deveria aparecer. Para determinar o ponto em que deveria aparecer, basta considerar o mesmo ponto com cota sobre o plano (i.e., $Z=0$).

3.4 DETERMINAÇÃO DOS PIXELS OCULTOS

A solução prática desenvolvida para o problema da oclusão é apresentado com base na nas Figuras 3.3 e 3.4, que representam uma imagem (ou matriz) quadrada com o centro A coincidente com a perpendicular da câmara, que será utilizada para verificar a oclusão.

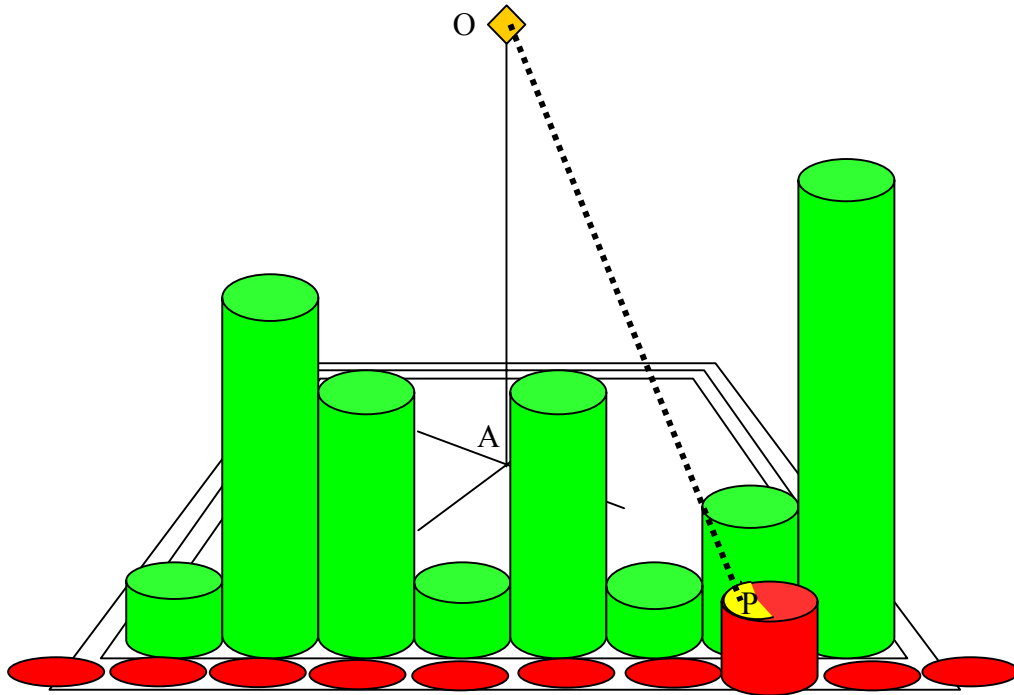


FIGURA 3.3 – Matriz de Distâncias.

Inicialmente, considere a divisão da imagem em quadrados concêntricos com 1 pixel de espessura. O pixel central deve coincidir com o ponto nadir da câmara. Para cada pixel sobre um quadrado (representado em vermelho no item a) a barreira que pode gerar oclusão é armazenada no quadrado interior, representado em verde no item b). O valor armazenado no quadrado verde é a maior distância r_B (da Figura 3.2) atingida por qualquer ponto capaz de ocultar o determinado pixel do quadrado. Essa distância r_B é a distância entre o ponto A e a base do cilindro P .

A Figura 3.3 apresenta a matriz de distâncias, onde a linha de um dos lados do “quadrado de distâncias”, formada pelos valores de deslocamento a partir do ponto A até o pixel onde está a informação. Dentre os diversos círculos vermelhos, que representam

os pixels que serão ou foram analisados, o único que está apresentando a forma de cilindro vermelho, corresponde ao pixel que está sendo analisado num dado instante. A parte amarela do cilindro é a porção visível, devido à geometria. A parte vermelha do topo do cilindro corresponde à porção não visível.

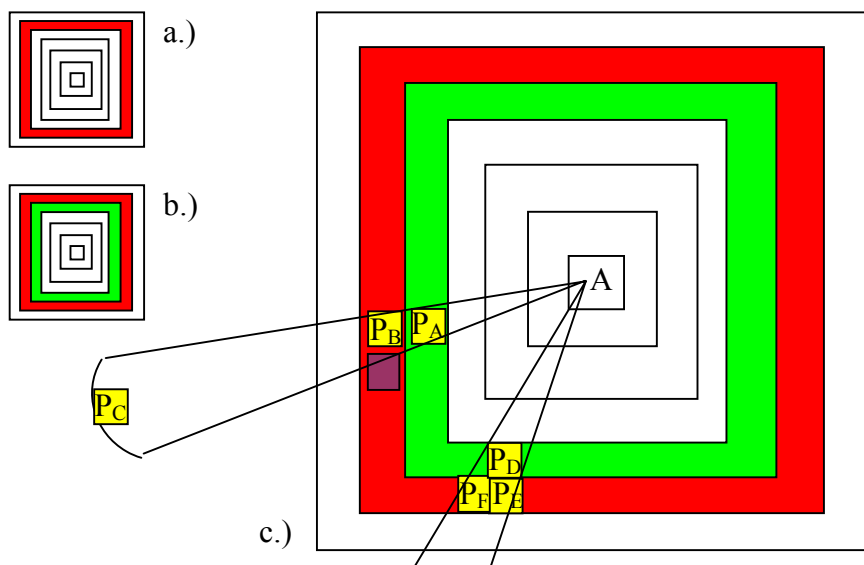


FIGURA 3.4 – Oclusão: a) Pixels que estarão sendo verificados; b) pixels com a informação referente ao maior deslocamento dos pixels internos; c) área de influência do quadrado verde.

Dentre os pontos que serão testados estão, na Figura 3.4 item c), os pontos P_F , P_E , P_B e o ponto mais escuro abaixo de P_B . Na Figura, P_B somente poderá ser ocultado por P_A , pois os demais pontos do quadrado verde não estão no plano vertical que passa pelo centro de projeção. O ponto abaixo de P_B será testado para um ponto abaixo de P_A . Note-se que o mesmo não ocorre com P_F e P_E , que têm o mesmo pixel P_D com as informações para teste de oclusão.

A equação da reta que passa pelo ponto e por A permite definir o pixel que pode oferecer a barreira mais significativa (caso do vizinho mais próximo) ou os pixels e a respectiva proporção (para interpolação).

Faz-se a comparação do valor onde o vermelho terá sua imagem (para modelos numéricos com valores positivos), sempre será mais distante que a própria distância do

ponto ao ponto A . Portanto o valor das colunas, que representa diretamente o valor da distância máxima alcançada por um pixel no segmento que vai desde o ponto até A , será um número real e a matriz gerada com todos os valores verdes deve ser atualizada com os valores de vermelho conforme o método definido na Seção 3.4.1.

3.4.1 “MÉTODO DA MATRIZ DE DISTÂNCIAS”

Considerando a Figura 3.5, o valor que será armazenado em PB

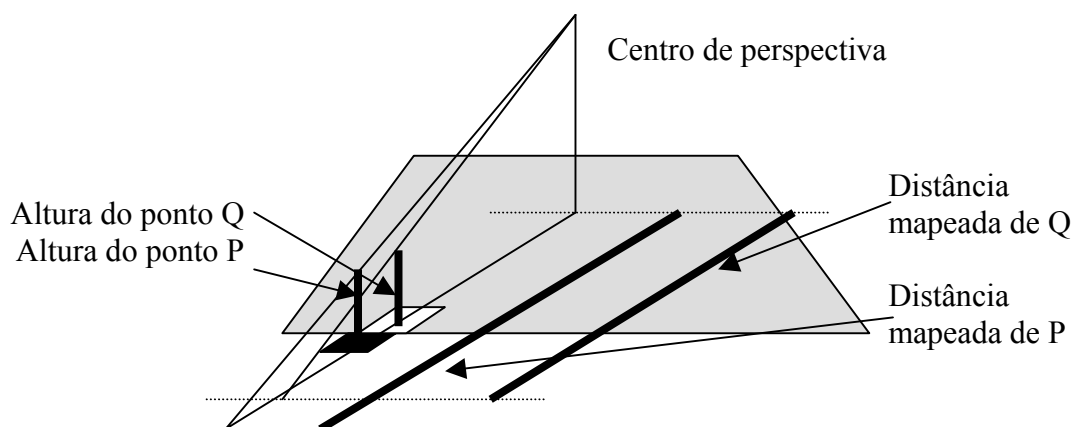


FIGURA 3.5 – Significado da “Matriz de Distâncias”.

A seqüência de procedimentos para a obtenção de uma maneira de verificar se um pixel está ou não oculto segue as seguintes etapas:

- para cada pixel P no quadrado externo;
- determinar o pixel no quadrado interno que tenha possibilidade de ocultar pelo menos 50% desse pixel P ;
- calcular o valor do deslocamento do ponto P devido ao relevo, como se não houvesse oclusão;
- como o valor obtido representa uma distância no plano-objeto, basta comparar a distância mapeada de P com o valor da distância mapeada do pixel interno Q correspondente;

- e) se o valor correspondente ao pixel interno Q for maior, há oclusão e o valor interno deve ser repetido na posição P ;
- f) se o valor obtido para o pixel P for maior que o obtido para o seu interno correspondente, o ponto é visível, proceda-se a ortorretificação e armazene-se o valor da distância do deslocamento ao pixel P na “Matriz de Distâncias”, assim definida.

3.4.2 “MÉTODO DA MATRIZ DE DISTÂNCIAS”, INTERPOLADA

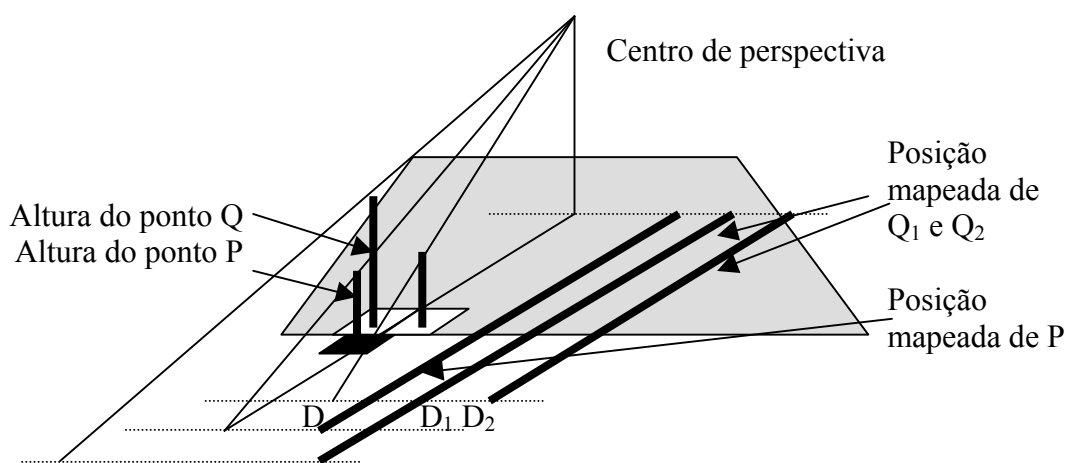


FIGURA 3.6 – Significado da “Matriz de Distâncias”.

No algoritmo apresentado no Apêndice foi utilizado o método do vizinho-mais-próximo. Para utilizar a interpolação, pode-se fazer uma pequena modificação no processo acima descrito e seguir a seguinte seqüência (Figura 3.6):

- a) para cada pixel “P” do quadrado externo;
- b) considerando os valores de cada pixel (sempre referidos ao centro), determinar os dois pixels (em alguns casos somente um, como já visto anteriormente), mais próximos do quadrado interno cujas distâncias à interseção são, respectivamente, a e b , conforme Figura 3.7. Os valores D_1 e D_2 , correspondentes a Q_1 e Q_2 na Matriz de Distâncias, permite que se calcule um valor interpolado, correspondente à média ponderada pelos

valores das distâncias (a e b) correspondentes aos pontos. Ou seja, o valor médio V pode ser calculado em função do valor D_1 e D_2 . Na Equação (3.53), se os valores estiverem calculados em pixel, o denominador será $a+b=1$.

$$V = \frac{D_1 \cdot a + D_2 \cdot b}{a + b} \quad (3.53)$$

- c) calcular o valor do deslocamento do ponto P devido ao relevo, como se não houvesse oclusão;
- d) como o valor obtido representa uma distância sobre o plano, compara-se esse valor com o valor V , calculado no item b);
- e) se o valor D do ponto P for menor que V , há oclusão e o V deve ser repetido na posição P ; desta forma, ao invés de P armazenar D , armazenará V ;
- f) se o valor D obtido para o pixel P for maior que V , o ponto é visível, proceda-se a ortorretificação e armazene-se o valor D da distância do deslocamento ao pixel P na “Matriz de Distâncias” (que é uma matriz de distâncias), desse modo definida.

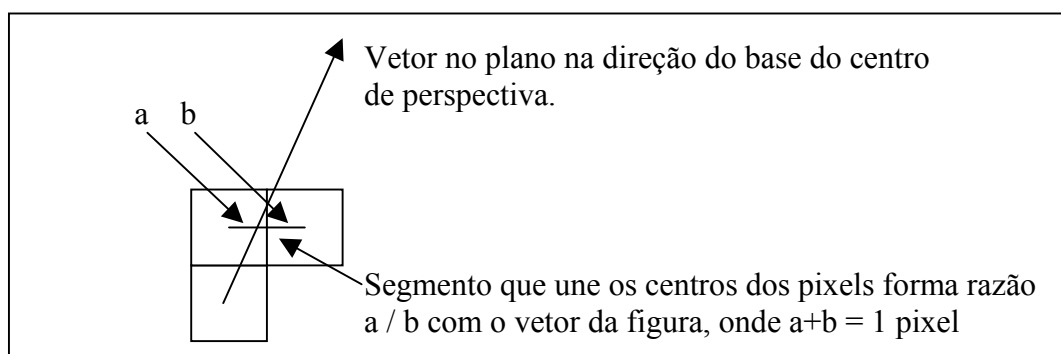


FIGURA 3.7 – Detalhe da interpolação.

CAPÍTULO 4

ESTUDO DE CASO

4.1 APRESENTAÇÃO

Este capítulo apresenta um estudo de caso de aplicação do procedimento de geração de ortoimagem, detalhado no capítulo anterior, para o caso de fotografia aérea produzida por câmara métrica.

Embora sem apresentar os dados ideais, este estudo apresenta a capacidade do algoritmo de lidar com dados reais, bastando, para isso, que tais dados sejam disponibilizados. É apresentado um caso para o estudo de viabilidade de produção de ortomagens sem a preocupação de que os dados fossem perfeitos (Skarlatos, 1999), incluindo a calibração prévia e uma breve análise dos resultados.

4.2 CONSIDERAÇÕES GERAIS

A maior parte dos programas fotogramétricos que realizam ortorretificação não determina os parâmetros de calibração da câmara. Geralmente esses valores são fornecidos pelo fabricante da câmara após medição de alta precisão.

Os valores obtidos no certificado de calibração de uma câmara métrica são valores confiáveis e, provavelmente melhores do que os obtidos pelos procedimentos apresentados neste trabalho. Entretanto, se os valores não forem conhecidos, a determinação desses parâmetros será obrigatória.

Comparativamente, um programa comercial que não calcule os parâmetros intrínsecos da câmara (ou a orientação interior) não permite que um fragmento de imagem aéreo seja utilizado. Não apresentando as 4 marcas fiduciais, a solução seria considerar o ponto principal coincidente com o centro geométrico da imagem. Mas nesse caso, a determinação do centro a partir de uma parte da imagem é impossível.

Em uma imagem de câmara não-métrica, que não apresente marcas fiduciais, os parâmetros fornecidos de desvio em relação ao ponto principal não podem ser aplicados, e o ponto principal pode estar deslocado, podendo, inclusive, o plano da

imagem pode não ser perfeitamente perpendicular ao eixo óptico da câmara, podendo haver um deslocamento significativo do ponto principal em relação ao centro da imagem.

4.3 O FUNCIONAMENTO DO ALGORITMO

4.3.1 O ALGORITMO

Para permitir uma visão geral do trabalho, é apresentada a seqüência de matrizes e arquivos essenciais, conforme implementado no IDL.

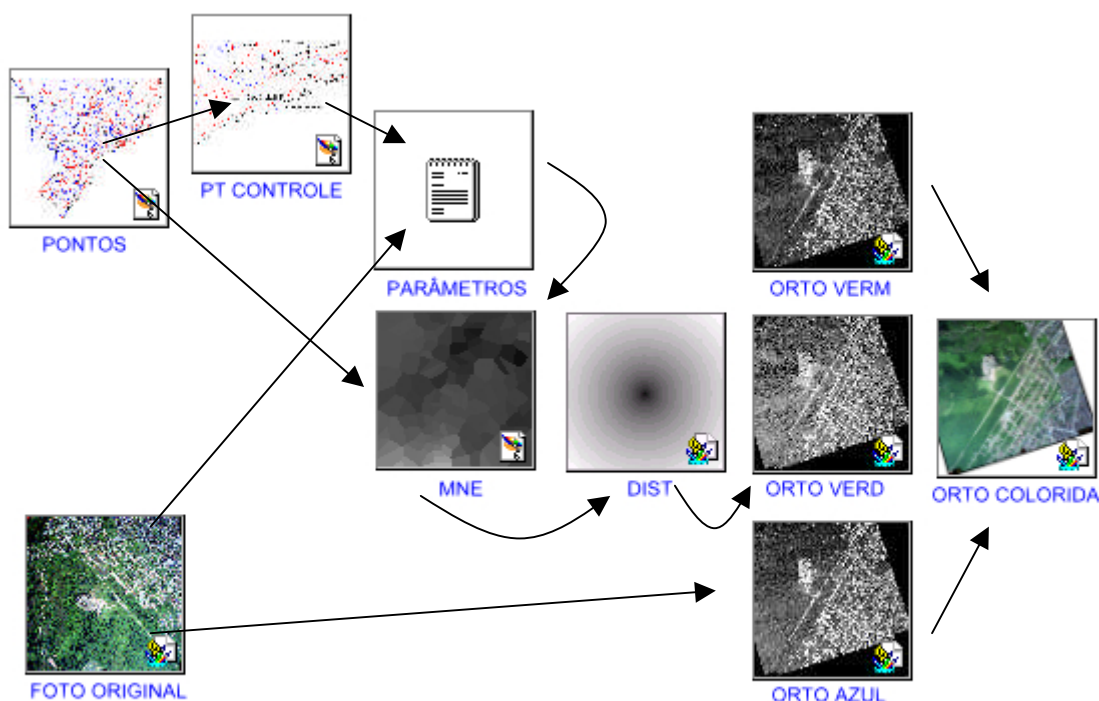


FIGURA 4.1 – Fluxo de utilização e geração de arquivos.

A Figura 4.1 apresenta a seqüência de geração e utilização de matrizes e imagens. O algoritmo utiliza apenas uma imagem do terreno (“foto original”) e pontos de controle da área do projeto (“pontos”).

A partir desses pontos, a seleção de alguns (no caso, sete) pontos define o conjunto de Pontos de Controle (“pt controle”) e, juntamente com a imagem, pode-se determinar os parâmetros (“parâmetros”).

Os parâmetros e os pontos de campo permitem gerar-se um modelo numérico de elevação (“mne”). Para o algoritmo apresentado, é então criada uma Matriz de Distâncias (“dist”), que permite a verificação de oclusão e a produção dos produtos finais (“orto”).

Na Figura 4.1, é apresentado o caso de uma imagem composta por 3 bandas (“orto verm”, “orto verd”, “orto azul”), processadas independentemente e recomposta na ortoimagem final (“orto colorida”). Esta operação foi realizada em um programa de edição de imagens.

O Apêndice 1 apresenta o algoritmo utilizado para gerar uma ortoimagem, como exemplo da implementação das equações do Capítulo 3. Foi utilizada a versão 5.2 da linguagem IDL. Foi criado um procedimento principal, além de algumas funções e outros procedimentos.

As funções e procedimentos são os seguintes: “JAN1”, “CRIAMNE”, “MATRIZDAT”, “VZOOMWINDOW”, “V_ORIENT_MMQ”, “ORTOGEN” e “DADOS”. Uma breve descrição de seu conteúdo é apresentada a seguir.

4.3.1.1 “JAN1”

Define um ambiente de janelas, a partir dos valores das coordenadas (X_0, Y_0) da câmara, do espaçamento da grade e do número de linhas e colunas, que devem ser iguais. O usuário preenche a tabela e este procedimento chama a função “MATRIZDAT”.

4.3.1.2 “CRIAMNE”

É um procedimento simples que utiliza uma função “MATRIZDAT” para criar um modelo numérico de elevação quadrado (p.ex.: 1501 linhas por 1501 colunas), com um espaçamento medido em metros (p.ex.: 1 m) e centro de projeção coincidente com o centro do modelo (454.863,46; 7.386.341,62). Procede na seguinte seqüência: gera

uma matriz quadrada de dimensão 200, apresenta a imagem, solicita ao usuário um nome de arquivo, por meio de uma caixa de diálogo, e grava o arquivo com o nome fornecido. Para manter a precisão, a matriz não pode ser armazenada em arquivo cuja resolução radiométrica seja o Byte, nem em 3 bandas, pois somente é utilizado um nível de informação. Desta forma, utilizou-se o “TIFF” com a opção “float”.

4.3.1.3 “MATRIZDAT”

É uma função que permite gerar um modelo numérico de elevação a partir de um número qualquer de pontos de controle registrados em um arquivo “DAT” (criado por programa de ortorretificação do sistema ERDAS). Esse modelo numérico deve suprir as informações de elevação requeridas pelo procedimento de ortorretificação, devendo ser regular, quadrado, tendo um pixel central que corresponde ao centro de perspectiva. O número de linhas e colunas fornecido como parâmetro, para isto, deve ser ímpar ou será acrescido de 1. Para calcular o valor de um ponto, foram identificados os pontos de coordenadas compreendidas na área abrangida pelo MNE, e utilizado o vizinho-mais-próximo, definindo-se regiões ao redor de um ponto de controle em que os elementos da matriz estão mais próximos. Obtém-se regiões equipotenciais, formando descontinuidades nos valores das linhas divisórias.

É apresentada outra opção para a utilização do fator de ponderação do inverso da k -ésima potência, gerando-se uma superfície mais suave. Para valores crescentes de k , a superfície resultante recebe crescente influência do ponto mais próximo. A Equação (4.1) representa essa média ponderada, onde z_p é o valor da cota de um ponto P , z_i é o valor de elevação do ponto i , r_i é a distância do ponto P ao ponto i , sendo os pontos i os pontos de controle. A constante k é um número real pré-definido

$$z_p = \frac{z_1 \cdot \frac{1}{r_1^k} + z_2 \cdot \frac{1}{r_2^k} + z_3 \cdot \frac{1}{r_3^k} + \dots}{\frac{1}{r_1^k} + \frac{1}{r_2^k} + \frac{1}{r_3^k} + \dots}, k \in \mathbb{R} \quad (4.1)$$

4.3.1.4 “VZOOMWINDOW”

Permite ampliar ou reduzir a imagem por um fator real positivo, podendo a imagem ser de uma ou mais bandas, sendo utilizada simplesmente a repetição ou supressão de pixels em função do fator escolhido.

4.3.1.5 “V_ORIENT_MMQ”

Transforma os valores de coordenadas dos pontos de controle e suas coordenadas de imagem respectivas em matrizes e utiliza a matriz peso para calcular o vetor-solução, pelo método dos mínimos quadrados (MMQ), obtido pela multiplicação das matrizes, conforme apresentado no Capítulo 3.

4.3.1.6 “ORTOGEN”

É o procedimento principal que calcula os parâmetros de orientação necessários à ortorretificação e gera a imagem ortorretificada a partir de um modelo numérico de elevação. Em primeiro lugar, obtém-se os valores de coordenadas de campo e imagem dos pontos de controle, armazenados em arquivo ASCII. Define os diferentes pesos para os tipos de pontos. Em seguida cria os valores de teste de instabilidade. Para simplificação numérica, é feita uma translação para que todos valores (X, Y) do terreno tenham média zero, posteriormente retornando à posição inicial, não sendo mudado o ponto de aplicação da rotação. São calculados os parâmetros para que seja possível realizar a ortorretificação. O fator de espaçamento é definido de modo a representar o mesmo espaçamento utilizado pelo modelo numérico de elevação, lido como a matriz “MAT_MNE”. A imagem fotográfica é lida como “MAT_FOTO”. O resultado será armazenado em “MAT_ORTO”, e será criada uma matriz com valores de distâncias alcançadas por cada pixel, chamada de “Matriz de Distâncias”, e lida como “MAT_DIST”. Foi utilizada apenas uma banda da imagem na matriz “MAT_FOTO” por simplificação, embora não haja qualquer problema em tratar-se das três bandas coloridas da imagem fotográfica, podendo-se gerar uma ortoimagem de três bandas. É feita a verificação de oclusão pixel-a-pixel e cria-se a matriz “MAT_DIST” simultaneamente com “MAT_ORTO”. Desta forma, ao término do processo a imagem está completa e a ortoimagem é gravada em forma de arquivo.

FUNCATE. O arquivo resultante ocupa cerca de 200 MBytes em formato TIFF de 24 Bytes.



FIGURA 4.3 – Fotografia aérea utilizada, da região de Caraguatatuba.

A imagem foi fornecida com 7 pontos de controle, sendo 3 obtidos em campo e 4 por aerotriangulação (por *software*). A imagem foi digitalizada em resolução de 28 micrômetros. Os parâmetros de calibração da câmara foram fornecidos e seus valores foram utilizados como a “verdade” para a avaliação dos “erros” observados, ou seja, as diferenças entre os valores observados e esses valores.

A imagem é constituída basicamente por duas partes: uma área urbana litorânea, plana, ocupando a metade da imagem, e a outra metade, rica em vegetação arbórea, correspondente ao início de uma serra, fracamente povoada. Os pontos de controle

utilizados foram os disponíveis. Não houve trabalho de campo específico para este



FIGURA 4.4 – Ortoimagem criada pela FUNCATE.

trabalho, mas a finalidade principal deste estudo de caso é verificar o funcionamento global do algoritmo. Os detalhes de funcionamento podem ser verificados pela aplicação da formulação utilizada (Capítulo 3).

4.3.3 A CÂMARA UTILIZADA NO VÔO

A câmara utilizada é da marca Zeiss, modelo RMK TOP 15, com número de série 145856 e lente PLEOGON A3, número de série 143100, com distância focal nominal de

153 mm e, aferida em 25 de março de 1997, obtendo-se o valor de sua distância focal calibrada de 152.755 mm. O centro fiducial calibrado é 11 μm em ξ e -2 μm em η .

4.3.4 PROCEDIMENTO INICIAL

Foi realizada a orientação da imagem a partir de pontos medidos em campo, identificados por meio de imagens de localização, ou croquis.

Cada croqui é constituído por três imagens capturadas em tela, em três escalas: uma apresenta toda a imagem, com um pequeno retângulo desenhado sobre ela, que aponta para a região da imagem em que se encontra o ponto; uma segunda janela apresenta uma ampliação da primeira, referente ao retângulo anterior; e uma terceira, que identifica o ponto na maior escala, permite identificá-lo com o máximo de detalhes.

Após executar o programa de ortorretificação desenvolvido, os pontos foram selecionados sobre a imagem apresentada em janela do IDL. O clique do botão esquerdo do mouse definia cada ponto e o valor era seqüencialmente solicitado. Digitou-se os valores de coordenadas de terreno (no caso, E, N e h, respectivamente) pelo teclado e, pelo botão direito do mouse, encerrou-se a execução dessa tarefa (ver Seção 1.2 do Apêndice 1).

O valor armazenado em disco permite que as execuções conseguintes não necessitem de nova identificação dos pontos. Após identificados os pontos, pode-se realizar os testes das seções seguintes, sem a necessidade de uma nova coleta de pontos e o fornecimento dos valores correspondentes (ver Apêndice 1.2).

Depois disso, o programa de ortorretificação calcula os elementos de orientação e executa a ortorretificação com base nas equações fotogramétricas do Capítulo 3.

4.3.5 LOCALIZAÇÃO DOS PONTOS DE CONTROLE

Os pontos de controle utilizados são de grande importância para o resultado. A quantidade de pontos, a confiabilidade dos valores correspondentes (geralmente em função dos equipamentos utilizados na medição e dos dados, algoritmos e parâmetros utilizados na aerotriangulação), a localização geométrica na imagem e a facilidade de

identificação do alvo na imagem (dependente do contraste radiométrico e da geometria do alvo e seu contexto) são fatores responsáveis pela qualidade dos parâmetros obtidos.

A resolução de leitura da imagem no *scanner* é, certamente, um dos fatores mais importantes, pois está diretamente ligado à precisão e exatidão dos pontos de controle. Quanto maior a dimensão do pixel na imagem, menor é a resolução de leitura matricial da fotografia e, conseqüentemente, maior também a dimensão do pixel no terreno. Desta forma, não adianta obter uma exatidão grande na leitura do ponto no campo, se a sua identificação na imagem somente permite a identificação do pixel correspondente, i.e., um erro de até meio pixel.

Neste trabalho foi utilizada uma imagem cuja resolução é de 28 μm (micrometros). Isto significa que um pixel mede 23 cm em uma imagem com escala média de 1/8.200. Meio pixel corresponde a 12 cm no terreno e é a menor unidade identificável na imagem, por eixo.

Além disso, os pontos não podem conter grandes erros, pois a formulação utilizada no Capítulo 3 utiliza uma matriz de pesos que não considera a possibilidade de erro de medição ou identificação do ponto. Caso pudesse haver erro, a matriz de pesos deveria ser recalculada e o método dos mínimos quadrados reaplicados para que a matriz de variância-covariância pudesse ser aplicada como matriz de pesos. Essa implementação não chegou a ser feita, tendo-se considerado que os dados de campo eram precisos o suficiente. Para avaliar os parâmetros de orientação, os pontos utilizados foram considerados sem erros que comprometessem o estudo, aliás, foi realizado um estudo de sensibilidade para verificar que esta consideração era válida, não havendo erro grosseiro no posicionamento dos pontos.

A geometria dos pontos era conforme Figura 4.5. Foram dois tipos de ponto: pontos de campo, medidos com GPS, chamados de pontos HV (horizontal e vertical). Os demais pontos foram obtidos por aerotriangulação, chamados de PT (pontos de aerotriangulação).

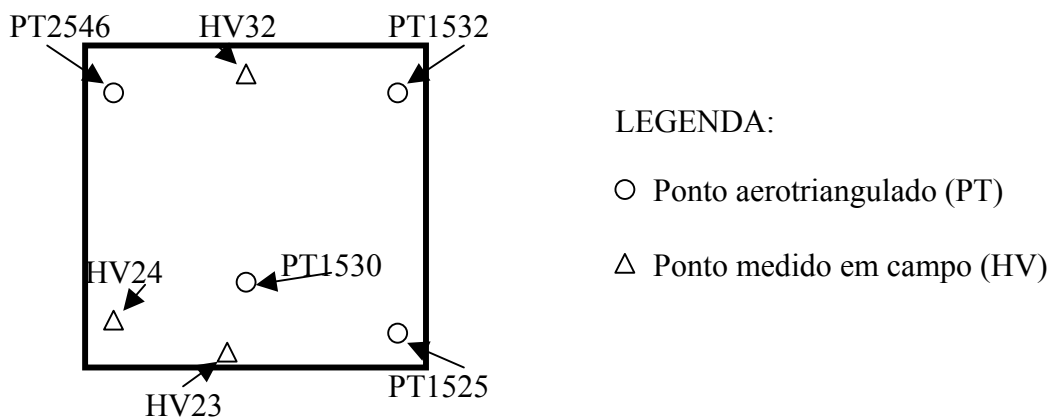


FIGURA 4.5 – Localização dos pontos na imagem.

Assim estão localizados os pontos: a noroeste, um ponto PT; a norte um ponto HV; a nordeste, um ponto PT; a sudeste, um ponto PT; a sul um ponto HV e a centro-sul, um ponto PT; a sudoeste, um ponto HV. Não há pontos fornecidos a leste, a oeste e ao centro da imagem. As imagens dos pontos (croquis) estão no Apêndice 2.

Os três pontos de campo (HV32, HV24 e HV23) correspondem a um vértice sobre a quina de um muro de uma residência, um vértice no pé da quina de uma edificação e a interseção do meio-fio com a linha de chegada de uma pista de *Kart*, respectivamente.

Os quatro pontos de aerotriangulação (PT1525, PT1530, PT1532 e PT2546) correspondem ao centro de uma pequena ilha de areia numa área de vegetação rasteira, ao vértice da sombra de um telhado, idem, e à interseção dos galhos de uma pequena árvore com o meio-fio de uma rua, respectivamente.

4.3.5.1 Croqui do ponto HV32



FIGURA 4.6 – Ponto HV32.

4.3.5.2 Croqui do ponto HV24



FIGURA 4.7 – Ponto HV24.

4.3.5.3 Croqui do ponto HV23



FIGURA 4.8 – Ponto HV23.

4.3.5.4 Croqui do ponto PT1525



FIGURA 4.9 – Ponto PT1525.

4.3.5.5 Croqui do ponto PT1530



FIGURA 4.10 – Ponto PT1530.

4.3.5.6 Croqui do ponto PT1532

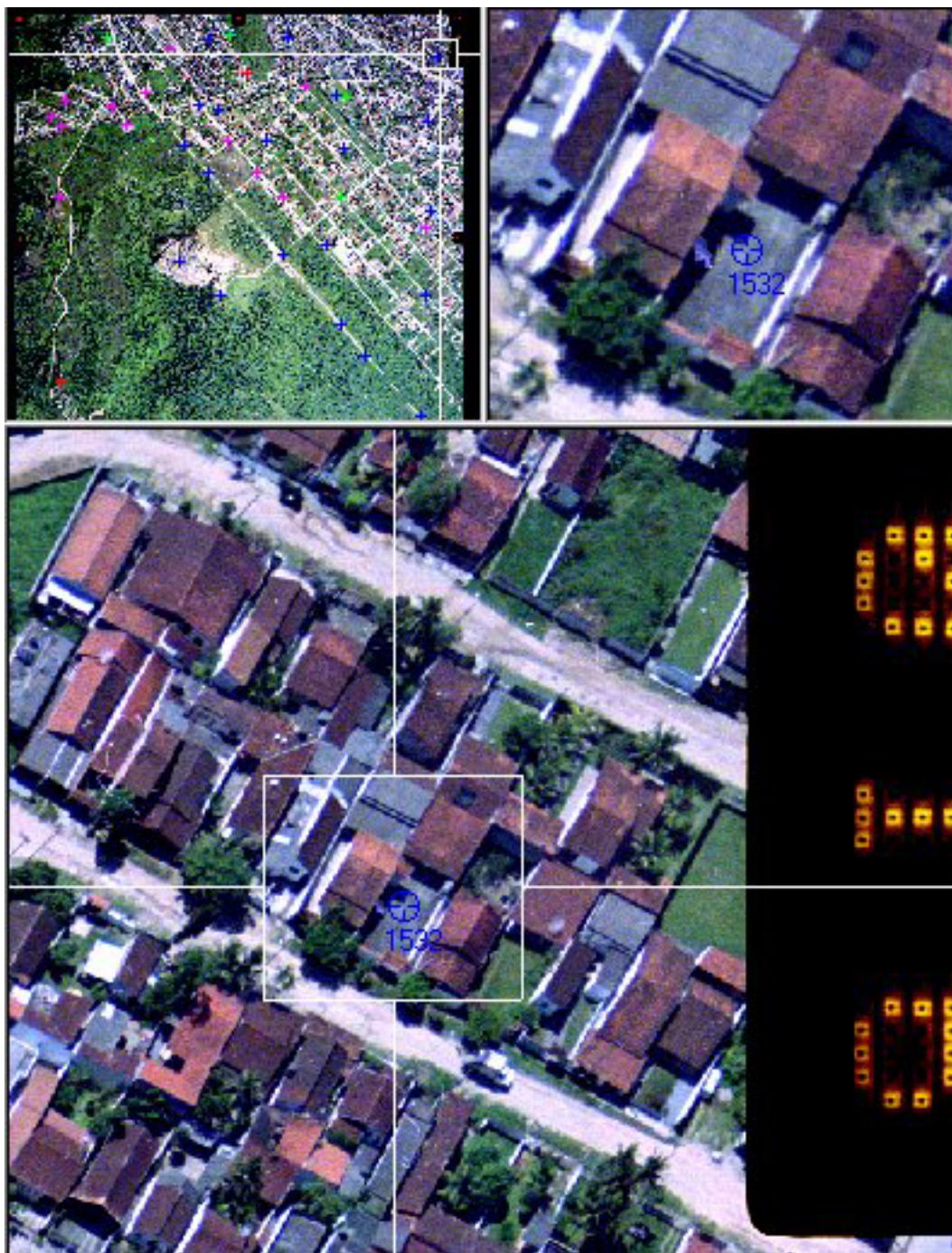


FIGURA 4.11 – Ponto PT1532.

4.3.5.7 Croqui do ponto PT2546

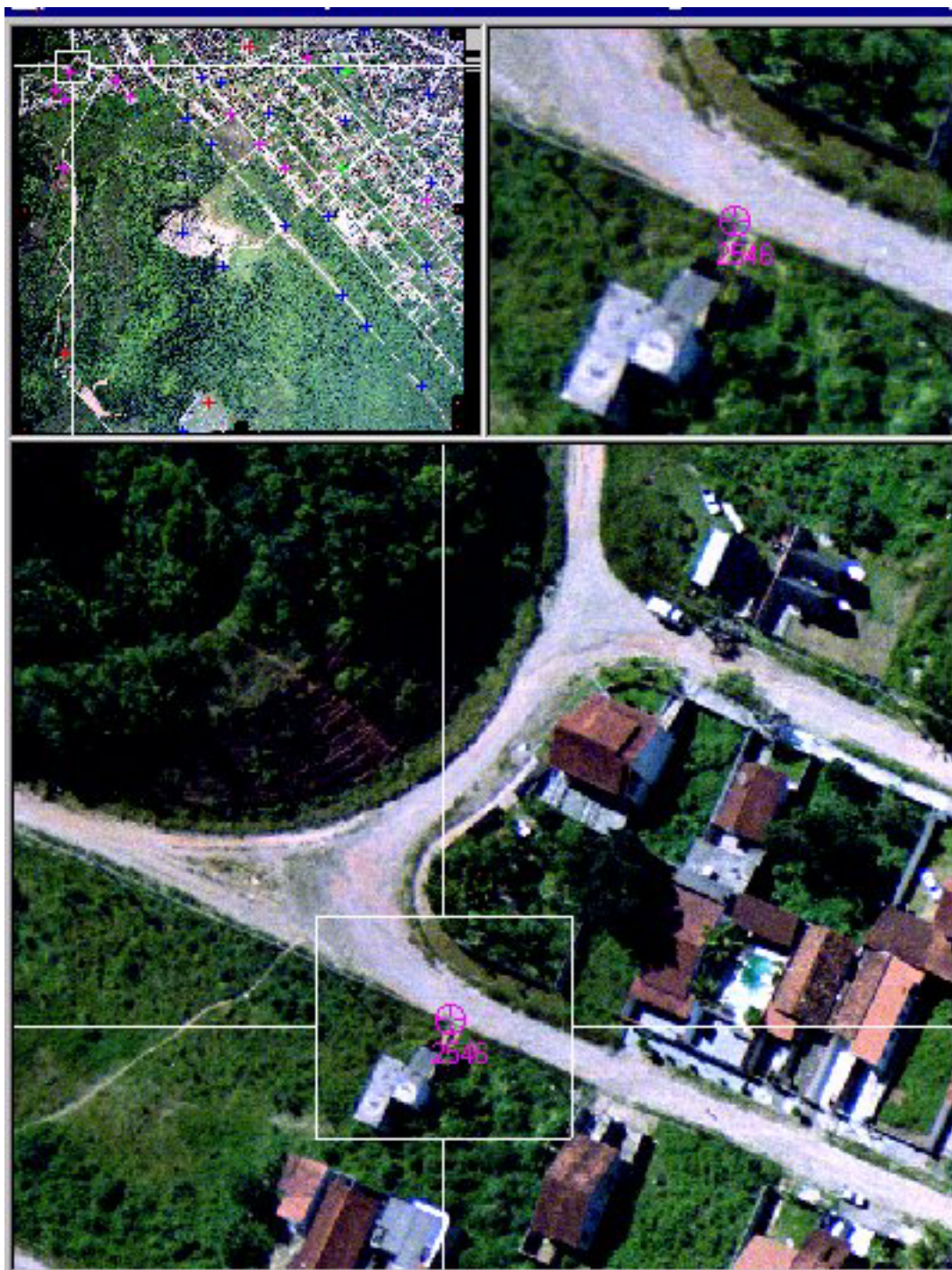


FIGURA 4.12 – Ponto PT2546.

4.4 AVALIAÇÃO

4.4.1 NÍVEL DE INSTABILIDADE DA ORIENTAÇÃO

4.4.1.1 Criando uma variação

Para medir a variação dos parâmetros em função de um pequeno erro de identificação do ponto, foram acrescentadas, no procedimento principal do programa, linhas de código que simulam um erro distando de 1 a 2 pixels na identificação dos pontos da imagem, em relação à posição inicial (considerado o pixel “correto”), sendo utilizada a vizinhança de 8 pixels em cruz e 4 pixels em “X”, conforme apresentado na Figura 4.13.

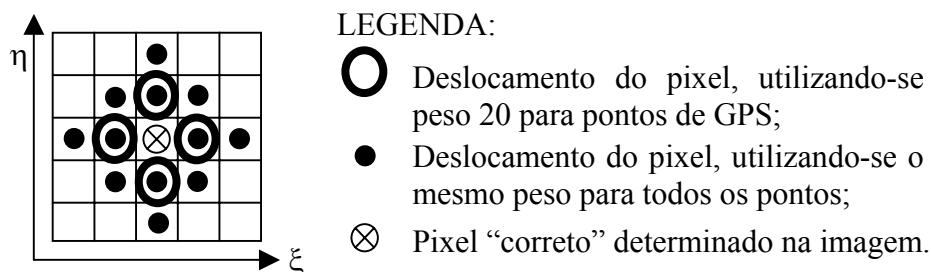


FIGURA 4.13 – Deslocamento dos pixels em torno do pixel “correto”.

As 12 posições, de cada um dos 7 pixels referentes aos pontos de controle, estão apresentadas como ponto preto. Referem-se ao cálculo dos parâmetros para 6 pontos corretos e um deslocado para aquela posição, todos os pontos com mesmo peso.

As 4 posições adicionais, também de cada um dos 7 pixels, estão representadas como um círculo, mas nesse caso, foi utilizada uma matriz de peso que permitia informar que os pontos que eram de controle deveriam ter um peso 20 vezes maior que os demais pontos, obtidos por aerotriangulação. Neste caso, para os 7 pontos que, cada um por vez, tiveram sua posição modificada conforme a Figura 4.13, 3 deles estavam recebendo peso 20, enquanto os 4 restantes receberam peso 1.

4.4.1.2 O teste dos parâmetros

O objetivo deste teste era verificar o quanto variavam os parâmetros devido a uma pequena variação na posição do pixel na imagem. A variação das distâncias aos valores originais resultantes (sejam com sinais positivos ou negativos), foi bastante grande. A distância focal variou 33% nos casos de mesmo peso; a posição do centro da imagem

variou 8,5% em relação ao lado da fotografia aérea de 23 cm; os ângulos de atitude variaram em 36 minutos; X_0 e Y_0 tiveram variação maior que 150 metros; e Z_0 atingiu os 426 metros de variação.

Atribuindo diferentes pesos aos dois tipos de pontos de controle e calculando-se o módulo da distância ao valor inicial, os erros foram menores. O resultado numérico desses valores é apresentado nas Tabelas 4.1 e 4.2.

TABELA 4.1 – VARIAÇÃO MÁXIMA DOS VALORES DOS PARÂMETROS AO VARIAR UM DENTRE 7 PIXELS NUMA VIZINHANÇA 12

Parâmetro	Variação (referente a 84 casos)
f	64,87 mm
ξ	20,57 mm
η	19,39 mm
φ	0,76 °
ω	0,24 °
κ	0,23 °
X_0	178,79 m
Y_0	163,69 m
Z_0	524,30 m

TABELA 4.2 – VARIAÇÃO MÁXIMA DOS VALORES DOS PARÂMETROS, AO VARIAR UM DENTRE 7 PIXELS NUMA VIZINHANÇA 4 EM CRUZ, COM PESO DE PONTOS DE CAMPO 20 VEZES MAIOR

Parâmetro	Variação (referente a 28 casos)
f	31,19 mm
ξ	10,67 mm
η	9,17 mm
φ	0,37 °
ω	0,12 °
κ	0,12 °
X_0	88,93 m
Y_0	84,80 m
Z_0	253,02 m

4.4.1.3 Resultado inicial

Os valores menores são observados na Tabela 4.2 por três possíveis razões: por representar a variação de apenas 4 casos; porque houve apenas variação de um pixel dentro desses casos; e porque o peso maior para os pontos que se tinha maior certeza apontam para uma maior rigidez nos resultados.

4.4.1.4 Teste de deslocamento de todos os pontos simultaneamente

Foi verificado um outro caso, em que todos os pontos foram submetidos à mesma modificação. Foi realizado o cálculo dos pontos com o mesmo peso, deslocando-se a posição de csi de um pixel no sentido positivo, para os 7 pontos. O resultado obtido mostrou que houve apenas um deslocamento nos parâmetros que definem o ponto principal (ξ_0, η_0) . A variação de todos os outros parâmetros foi nula. Isso mostra que se houver um erro de mesma magnitude e sinal na identificação de todos os pontos, os parâmetros não são comprometidos.

4.4.1.5 Maiores e menores erros

As Tabelas 4.3 a 4.6 apresentam o resumo dos maiores e menores erros em relação ao valor “correto” da Tabela 4.1. Para esse cálculo, foi realizado o deslocamento de cada um dos 7 pontos e cada célula das Tabelas indica qual o valor do ponto que mais se afastou ou aproximou do valor correto do parâmetro correspondente. Foram verificados os pontos que, ao serem deslocados, proporcionaram o melhor resultado. Deste modo, o resultado, considerando-se apenas o cálculo sem realizar a ponderação (Tabelas 4.3 e 4.4), apontou dois pontos, o ponto HV23 e o ponto HV32 como os responsáveis pelos menores valores. As Figuras 4.7 e 4.8 apresentam os resultados dos efeitos dos deslocamentos do ponto HV23 e HV32 que resultaram no menor erro.

TABELA 4.3 – VERIFICAÇÃO DO COMPORTAMENTO DOS PARÂMETROS, QUANDO (ξ, η) VARIAM DE UM ÚNICO PONTO. MAIOR E MENOR MÓDULO DO ERRO DE f, ξ_0, η_0

Caso (ξ, η)W	f		ξ_0		η_0	
	+	-	+	-	+	-
(-1, -1)1	21,28	0,28	22,97	13,96	46,45	38,69
(-1, +1)1	32,77	0,77	21,85	13,56	44,82	37,14
(+1, -1)1	26,42	2,26	21,44	12,30	46,04	37,88
(+1, +1)1	27,05	4,71	20,89	11,92	44,39	36,88
(0, +2)1	51,15	1,19	20,54	14,75	47,17	31,97
(0, +1)1	29,88	4,24	18,79	15,99	44,48	37,15
(0, -1)1	21,89	1,69	18,65	15,99	46,25	38,28
(0, -2)1	32,24	0,98	20,07	14,86	51,36	34,70
(+2, 0)1	23,75	3,57	23,60	6,84	44,19	39,23
(+1, 0)1	18,01	7,97	20,44	12,11	42,89	40,41
(-1, 0)1	17,52	7,53	22,39	14,16	42,58	40,09
(-2, 0)1	22,73	2,75	27,41	11,05	43,59	38,57
Total	51,15	0,28	27,41	6,84	51,36	31,97

TABELA 4.4 – VERIFICAÇÃO DO COMPORTAMENTO DOS PARÂMETROS, QUANDO (ξ, η) VARIAM DE UM ÚNICO PONTO. MAIOR E MENOR MÓDULO DO ERRO DE $\varphi, \omega, \kappa, X_0, Y_0, Z_0$

Caso (ξ, η) W	φ		ω		κ		X_0		Y_0		Z_0	
	+	-	+	-	+	-	+	-	+	-	+	-
(-1, -1)1	0,32	0,073	0,11	0,023	0,24	0,12	280,41	222,77	237,63	150,92	185,72	15,91
(-1, +1)1	0,46	0,087	0,15	0,035	0,20	0,13	285,33	190,99	205,30	157,58	278,61	19,32
(+1, -1)1	0,37	0,023	0,12	0,014	0,20	0,12	289,63	201,27	205,57	152,83	227,63	16,14
(+1, +1)1	0,38	0,122	0,13	0,039	0,21	0,09	266,79	204,87	212,08	126,24	232,03	51,52
(0, +2)1	0,67	0,023	0,22	0,016	0,21	0,13	301,86	139,54	211,69	152,88	426,38	3,20
(0, +1)1	0,42	0,115	0,14	0,043	0,19	0,14	275,11	198,02	195,53	166,07	255,07	48,01
(0, -1)1	0,33	0,048	0,11	0,019	0,18	0,14	285,07	212,88	195,58	165,84	190,31	0,25
(0, -2)1	0,45	0,081	0,14	0,027	0,21	0,12	318,33	176,40	212,14	152,17	273,47	18,72
(+2, 0)1	0,33	0,126	0,11	0,032	0,23	0,05	265,78	219,98	228,74	97,13	205,82	42,51
(+1, 0)1	0,27	0,169	0,09	0,052	0,20	0,11	255,81	232,96	204,44	139,03	159,29	78,04
(-1, 0)1	0,26	0,164	0,09	0,055	0,22	0,13	257,92	235,14	220,86	156,24	155,24	74,38
(-2, 0)1	0,32	0,115	0,12	0,038	0,28	0,09	269,91	224,41	260,82	132,36	197,27	35,54
Total	0,67	0,02	0,22	0,01	0,28	0,05	318,33	139,54	260,82	97,13	426,38	0,25

TABELA 4.5 – VERIFICAÇÃO DO COMPORTAMENTO DOS PARÂMETROS, QUANDO (ξ, η) VARIAM DE UM ÚNICO PONTO. MAIOR E MENOR MÓDULO DO ERRO DE f, ξ_0, η_0 . PESO DOS PONTOS DE GPS 20 VEZES MAIORES QUE OS PONTOS DE AEROTRIANGULAÇÃO

Caso (ξ, η) W	f		ξ_0		η_0	
	+	-	+	-	+	-
(+1, 0) 20	5,25	1,23	15,87	7,87	47,71	46,19
(-1, 0) 20	6,30	0,50	18,54	10,63	47,74	46,23
(0, -1) 20	16,39	1,72	15,56	11,61	51,59	44,25
(0, +1) 20	14,80	0,54	14,96	11,12	49,40	42,42
Total	16,39	0,50	18,54	7,87	51,59	42,42

TABELA 4.6 – VERIFICAÇÃO DO COMPORTAMENTO DOS PARÂMETROS, QUANDO (ξ, η) VARIAM DE UM ÚNICO PONTO. MAIOR E MENOR MÓDULO DO ERRO DE $\varphi, \omega, \kappa, X_0, Y_0, Z_0$. PESO DOS PONTOS DE GPS 20 VEZES MAIORES QUE OS PONTOS DE AEROTRIANGULAÇÃO

Caso (ξ, η) W	φ		ω		κ		X_0		Y_0		Z_0	
	+	-	+	-	+	-	+	-	+	-	+	-
(+1, 0) 20	0,07	0,001	0,03	0,002	0,17	0,07	317,24	296,36	181,11	116,98	28,57	2,12
(-1, 0) 20	0,07	0,005	0,03	0,000	0,19	0,10	315,89	295,23	201,79	137,49	37,10	2,13
(0, -1) 20	0,16	0,002	0,05	0,002	0,16	0,11	346,16	281,54	174,38	141,73	119,09	0,09
(0, +1) 20	0,24	0,018	0,08	0,001	0,16	0,11	328,21	257,23	178,41	146,24	133,93	10,75
Total	0,24	0,00	0,08	0,00	0,19	0,07	346,16	257,23	201,79	116,98	133,93	0,09

Os valores correspondentes aos menores erros para cada parâmetro são apresentados na Tabela 4.7.

4.4.1.6 Correção dos pixels baseado nos resultados de instabilidade

Considerando-se a hipótese de ter havido, neste estudo de caso, erro na determinação dos pixels da imagem, devido a alguma falha de identificação, as Tabelas 4.3 a 4.6 foram utilizadas para indicar uma possível correção.

Foi verificado o caso que proporcionava o menor erro final e verificado, na planilha de cálculos que deu origem às tabelas, qual era o pixel que estava sendo modificado.

Foi constatado que apenas dois pontos, mas em diversas posições, proporcionavam os valores mais próximos dos parâmetros “verdadeiros”. Houve também um terceiro ponto que obteve o mesmo resultado em um dos parâmetros. Esses pontos podem estar mal identificados na imagem, embora isto seja apenas uma das possíveis hipóteses.

Os pontos foram o HV23 e o HV32, além do PT1525.

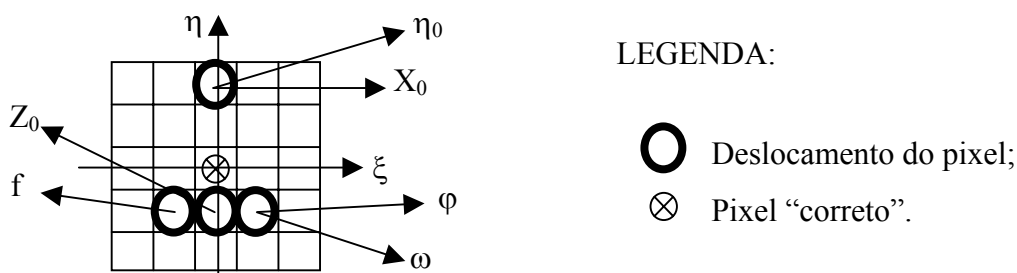


FIGURA 4.14 – Efeito do deslocamento do ponto HV23 e parâmetro relacionado que mais aproxima-se do valor correto.

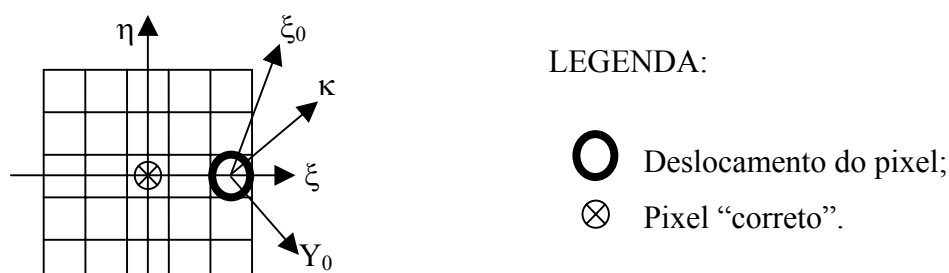


FIGURA 4.15 – Efeito do deslocamento do ponto HV32 e parâmetro relacionado que mais aproxima-se do valor correto.

4.4.2 RESULTADO DA CALIBRAÇÃO

Os resultados da calibração, apresentados na Tabela 4.7, apresentam valores diferentes para pontos diferentes. Após realizada uma revisão na formulação, em que:

- a) os valores não apresentaram estabilidade; com uma pequena variação na identificação de um ponto de controle, o resultado variou, chegando aos limites apresentados na Tabela 4.1 e 4.2;

- b) os 7 (sete) pontos disponíveis foram testados 6 a 6; não houve evidências que indicassem que 1 (um) único ponto estivesse mal localizado ou com coordenadas erradas; ou mais de um ponto desviou-se do valor real ou a resolução geométrica da imagem pode ter sido insuficiente;
- c) a variação de altitude entre o ponto mais alto e o mais baixo era de 10 metros; a pouca variação faz com que a modelagem aproxime-se de um plano, o que resultaria em determinante nulo, impossibilitando a obtenção dos parâmetros; a proximidade com o plano pode estar prejudicando a estabilidade da geometria;

chegou-se à conclusão de que:

- a) a instabilidade aponta para o fato de que o modelo necessita de variação maior de altimetria do que a variação dos pontos utilizados;
- b) podem haver pontos mal identificados na imagem;
- c) podem haver pontos com valores de coordenadas incorretos;

TABELA 4.7 – COMPARAÇÃO DOS MELHORES RESULTADOS OBTIDOS COM O DESLOCAMENTO DE UM PIXEL, COM O VALOR CORRETO

Parâmetro	Comparação dos resultados		
	“Correto”	7 pontos de mesmo peso	Varição nos pesos (1 ou 30)
f	152,8 mm	165,4 mm	151,9 mm
ξ	5,8 mm	-11,53 mm	-11,35 mm
η	-5,3 mm	36,22 mm	40,42 mm
φ	-1,661 °	-1,877 °	-1,715 °
ω	-0,2062 °	-0,2786 °	-0,2292 °
κ	-73,20 °	-73,04 °	-73,02 °
X_0	454.863,46 m	455.109,07 m	455.149,64 m
Y_0	7.386.341,62 m	7.386.521,89 m	7.386.529,06 m
Z_0	1.253,71m	1.369,27 m	1.259,85 m

Os valores apresentados na coluna “Correto” da Tabela 4.7 correspondem aos registrados no certificado de calibração ou os parâmetros calculados por trabalho de gabinete na FUNCATE, utilizando *software* de fotogrametria digital e outras informações (ajustamento de bloco, pontos de controle, marcas fiduciais, etc).

A Tabela 4.7 apresenta as células destacadas quando o valor correspondente a um dos casos aproxima-se significativamente mais do valor “correto”. Utilizando-se ponderação, chegou-se a um melhor resultado. Isto indica que os pontos de controle de campo podem estar melhor identificados na imagem ou que apresentam valores de coordenadas de terreno com exatidão significativamente maior do que os demais.

Dois pontos de controle de aerotriangulação apresentam um certo grau de dificuldade para a identificação na imagem. Em um dos casos, embora seja fácil identificar a área em que se situa, não é tão certa a identificação do pixel exato referente ao centro, pois deve-se determinar o centro de um grupo de pixels (PT1525). No outro caso, uma linha intercepta um grupo de pixels (o meio-fio faz interseção com uma árvore, PT2546).

4.4.3 AVALIAÇÃO DA ORTORRETIFICAÇÃO



FIGURA 4.16 – Ortoimagem gerada. Foi obtida a partir de um MNE de espaçamento igual a 1 m e 2001 linhas por 2001 colunas.

4.4.4 AVALIAÇÃO COMPARATIVA COM PRODUTOS COMERCIAIS

Ao testar um programa fotogramétrico chamado “PhotoModeler Pro”, da empresa canadense EOS Systems, Deng (1999) apresenta em seus testes uma baixa convergência nos valores de distância focal obtidos em repetidos ajustamentos para um mesmo caso, enquanto apresenta outros resultados que sugerem que os coeficientes de distorção simétrica radial e de distorção de descentralização são significativos e devem ser compensados quando utilizadas imagens de fotografias lidas em *scanner*. Citando Karara e Abdel-Aziz (1974), reforça a significância dos termos de K1 para as distorções de lentes em câmaras não-métricas.

No “PhotoModeler”, a medição dos parâmetros de orientação interior para várias fotos e o mesmo número e tipo de pontos de controle, resultou em valores diferentes; e para uma mesma foto, variando os pontos de controle, indicou a incerteza do *software*. Os valores de distância focal calculados chegaram a apresentar uma variação de 17% numa mesma foto. Segundo Deng (1999), o referido programa “necessita de melhorias para resolver o problema das incertezas e falhas na solução dos parâmetros de orientação interior e dos coeficientes de distorção das lentes”.

CAPÍTULO 5

CONCLUSÕES E SUGESTÕES

5.1 CONCLUSÕES

O algoritmo apresentado pode ser aprimorado, tornando-se mais conciso pela utilização de um menor número de operações computacionais básicas. Otimizar a solução que é apresentada neste trabalho, entretanto, deve ser um passo posterior. Tomando-se este trabalho como base, espera-se poder chegar a uma solução muito mais eficiente, que permita ortorretificar grandes arquivos de imagens (i.e., arquivos digitalizados em alta-resolução), consumindo um mínimo de memória física e de tempo. Portanto, uma importante contribuição deste trabalho é poder ser um “modelo” para a criação de um programa (ou módulo) de ortorretificação.

Além de servir como referência de implementação, a programação em IDL (Apêndice) proporciona uma grande flexibilidade no sentido de realizar pequenas modificações que permitam modificar disposição de janelas, novos formatos de arquivos de imagem de entrada ou saída, diferentes relatórios, acrescentar outras correções não implementadas ou mesmo utilizar o próprio programa.

Esta dissertação apresenta um procedimento computacional detalhado para geração de ortoimagens para fotografias aéreas de áreas urbanas, com um algoritmo de ortorretificação que não necessita dos parâmetros de calibração da câmara, o que é particularmente importante para as câmaras não-métricas, que podem não ter um certificado de calibração. Também não são utilizadas as marcas fiduciais para a orientação interior. Isso permite que uma imagem de câmara não-métrica, ou de uma câmara cujos parâmetros não sejam conhecidos, seja ortorretificada. É possível calcular os parâmetros e ortorretificar uma imagem de parte de uma fotografia, pois não há restrição que exija a imagem inteira.

O estudo de caso apresenta um MNE pobre, criado a partir de 7 pontos, apesar de os recursos desenvolvidos permitirem a modelagem de áreas urbanas com o fenômeno da oclusão, esperando-se um MNE de alta resolução. A funcionalidade do mecanismo de

oclusão apresentado foi verificada em testes com o IDL, que não foram incluídos neste trabalho por tratarem-se de testes rudimentares. Mesmo sem o modelo de terreno adequado, considero que o objetivo principal foi atingido, i.e., sintetizar num único trabalho os fundamentos metodológicos, a metodologia de ortorretificação, incluindo o fenômeno da oclusão, um estudo de caso que mostra a operacionalidade do algoritmo e os detalhes computacionais apresentados com o algoritmo apresentado em linguagem IDL.

5.2 SUGESTÕES PARA TRABALHOS FUTUROS

Foi apresentada uma metodologia que corrige o efeito geométrico da altimetria da superfície (ou o relevo do terreno) resultante na imagem.

A primeira sugestão é a ortorretificação de uma fotografia aérea de um centro urbano, utilizando a metodologia aqui apresentada, com o uso de um MNE de alta resolução, obtido, p. e., por *laser scanner*.

Os erros geométricos de distorções e deformações de lentes e filmes ou sensores, refração atmosférica e aberrações cromáticas, podem ser estudados e equacionados para implementação conjunta de tais correções com a ortorretificação.

O aumento da precisão pode ser obtido com o acréscimo de quatro parâmetros, que seriam os coeficientes polinomiais para a distorção radial das lentes e a distorção de excentricidade das lentes, podendo-se utilizar as equações implementadas no “PhotoModeler Pro” e apresentadas em Deng (1999). Youcai e Haralick (1999) apresentam restrições que podem ser acrescentadas para buscar valores reais na calibração da câmara. Apresentam 10 (dez) condições geométricas e citam Huang e Haralick (1997) como uma breve introdução aos modelos matemáticos correspondentes.

Outra implementação seria considerar a existência de um par de fotos (um modelo fotogramétrico) para a obtenção de ortoimagens, já possuindo-se o modelo numérico da superfície. Neste caso, a área de oclusão diminuiria e a qualidade das informações radiométricas aumentaria pela duplicidade de informações obtidas nas duas imagens. Este não seria uma tarefa complexa, podendo ser realizada pela sobreposição das duas

ortoimagens obtidas em cada foto e algum procedimento de análise comparativa e decisão.

Como a mais interessante e difícil tarefa, considero a obtenção do modelo numérico de superfície a partir de duas fotografias aéreas. A sugestão para o procedimento é que fosse inteiramente automático, baseado em autocorrelação, mas com muitas outras restrições morfológicas de, digamos, casamento de segmentos (após segmentação da imagem) e pontos automaticamente identificados. O grande desafio é a existência de sombras e nuvens, que precisam ter uma solução estudada em profundidade.

REFERÊNCIAS BIBLIOGRÁFICAS

Albertz, J.; Kreiling, W. **Photogrammetrisches Taschenbuch**. 4.ed. Wichmann: Karlsruhe, 1989.

Amhar, F.; Ecker, R. **Accurate mapping of buildings in digital orthophotos**.

[online] 12 Apr 1999

< http://www.ipf.tuwien.ac.at/veroeffentlichungen/fa_re_p_ica95.html >

Amhar, F; Jansa, J.; Ries, C. **The generation of true orthophotos using a 3D building model in conjunction with a conventional DTM**. IAPRS, v.32/4, Stuttgart, 1998, pp.16-22. [online] 13 Apr 1999

[online] 13 Apr 1999

< <http://www.ipf.tuwien.ac.at/veroeffentlichungen/jansa57.pdf> > e

< <http://www.ipf.tuwien.ac.at/veroeffentlichungen/96-98.html> >

Ammerman, P. Public works GIS data technologies lend crucial support to local governments. In: **Public Works: behind the scenes**. EOM Archives. Aug 1998.

[online] 2001

<<http://www.eomonline.com/Common/Archives/August%2098/ammerman3.htm>>

Andrade, J. B.; Olivas, M. A. A. Calibração de câmeras aerofotogramétricas. Boletim da Universidade Federal do Paraná: **Geodésia**, n.26, p.39, Curitiba, Paraná. UFPr, 1981.

Armangué, X.; Salvi, J.; Batlle, J. **A comparative review of camera calibrating methods with accuracy evaluation**. 2000 [online]

< <http://eia.udg.es/~armangué/research/papers/siarp00.pdf> >

Boccardo, P.; Dequal, S.; Lingua, A.; Rinaudo, F. True orthophoto for architectural and archaeological applications. In: International Workshop, Recreating the past – visualization and animation of cultural heritage, Fev. 2001, Ayutthaya, Thailand.

Proceedings... Ayutthaya, Thailand : ACRS, 2001. [online]

< <http://www.gisdevelopment.net/events/isprs/2001/ts2/isprs2001pf.htm> >

Branco, J. F. C. O emprego da fotografia no levantamento topográfico do Distrito Federal. **Anuário da diretoria do serviço geográfico do exército, n.1, 1948**, p.65-75. Serviço Geográfico do Exército, Ministério da Guerra. Rio de Janeiro, 1949.

Cooper, M. A. R.; Cross, P. A. Statistical concepts and their application in photogrammetry and surveying. **Photogrammetric Record**, v.12, n.71, p.637-663, Apr, 1988.

Cooper, M. A. R.; Cross, P. A. Statistical concepts and their application in photogrammetry and surveying (continued). **Photogrammetric Record**, v.13, n.77, p.645-678, Apr, 1991.

Cory, M. J.; Kirwan, R. A.; Walker, A. S. Ordnance Survey Ireland and its transition to digital photogrammetry. **Photogrammetric Engineering & Remote Sensing**, v.65, n.3, Mar, p.218-226, 1999.

Deng, G. **Practical testing and evaluating of the EOS PhotoModeler, an off-the-shelf digital close range photogrammetric software package**. New Brunswick, Canada: University of New Brunswick, Fredericton. Department of Geodesy and Geomatics Engineering. Technical Report No. 201, 1999. 93p.

Destri, A. R. **Modelos digitais do terreno**. 1987, 267. Tese de Mestrado. Instituto Militar de Engenharia, Rio de Janeiro, 1987.

Ghosh, S. K. **Analytical Photogrammetry**. New York: Pergamon Press, 1979. 23 p.

Hatze, H. High-precision three-dimensional photogrammetric calibration and object space reconstruction using a modified DLT-approach. **J. Biomechanics**, n.21, p.533-538. 1998.

Heikkilä, J.; Silvén, O. **A four-step camera calibration procedure with implicit image correction.**

Huang, Y. C.; Haralick, R. M. **Camera calibration with constraints.** Washington: Department of Electrical Engineering, University of Washington, 1997. Technical Report.

Karara, H. M.; Abdel Aziz, Y. I. Accuracy aspects of non-metric imageries. **Photogrammetric Engineering**, v.40, n.9, pp.1107-1117. 1974.

Kolesnik, M. **Direct recover of the camera internal parameters using known angles.** ACCV 2002, Jan 2002, Melbourne [online].

<<http://viswiz.gmd.de/~marina/papers/pdf/accv'2002.pdf>> Set 2001.

Kraus, K. **Photogrammetry: fundamentals and standard processes.** 4.ed. Bonn: Dümmler, v.1, 1993.

Kwon, Y. -H. **Kwon3D motion analysis Web.** 1998. [online]

< <http://kwon3d.com/index.html> >

Moffitt, F. H.; Mikhail, E. M. **Photogrammetry**, 3.ed. New York: Harper and Row: 1980.

Oliveira, C. **Dicionário cartográfico.** 4.ed, Rio de Janeiro: IBGE, 1993.

Quan, L.; Lan, Z. **Linear n-point camera pose determination.** 1999 [online]

< <http://citeseer.nj.nec.com/quan99linear.html> >

Rau, J. -Y.; Chen, N. -Y.; Chen, L. -C. **Hidden compensation and shadow enhancement for true orthophoto generation.** ACRS 2000. [online]

< www.gisdevelopment.net/aars/acrs/2000/ts4/digi0003pf.htm >

_____. **IDL Basics:** Interactive Data Language version 4.0. Boulder, CO: Research Systems, Inc. 1995a.

_____. **Learning IDL: An interactive training course.** Boulder, CO: Research Systems, Inc. 1995b

Skarlatos, D. Orthophotograph production in urban areas. **Photogrammetric Record**, v.16, n. 94, p.643-650. Oct 1999. [online]
<<http://snap.lut.ac.uk/PhotSoc/abs/phroct99/Skarlatos.pdf>>

Smith, G. S. **Digital orthophotography and GIS.** [online] 08 Apr 1999
< <http://pasture.ecn.pardue.edu/~aggrass/esri95/to150/p124.html> >

Thompson, M. M. (ed) **Manual of photogrammetry.** 3.ed. Virginia : American Society of Photogrammetry:, v.1 e 2., 1966.

Triggs, B. **Camera pose and calibration from 4 or 5 known 3D points.** ICCV, 1999 [online] < <http://citeseer.nj.nec.com/cache/papers/cs/8178/http:zSzzSzwww.inrialpes.frzSzmovizSzpeoplezSzTriggszSzdlt-iccv99.pdf/triggs99camera.pdf> >

Whitmore, G. D. The development of Photogrammetry, In: _____ **Manual of Photogrammetry.** 2.ed. Washington: American Society of Photogrammetry, p.1-16, 1952.

Youcai, H.; Haralick, R. M. Testing camera calibration with constraints. **Photogrammetric Engineering & Remote Sensing**, v.65, n.3, Mar, p.249-258, 1999.

Zhang, Z. **Determining the epipolar geometry and its uncertainty:** a review. July 1996, 26p. INRIA Research Report N°.2927. [online] 12 Apr 1999.
< <ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-2927.pdf> >

GLOSSÁRIO

Para melhor esclarecimento de termos existentes no presente trabalho, recomenda-se a consulta do glossário do Manual of Photogrammetry e de Oliveira (1993). Visando facilitar a consulta, alguns termos são listados e, a seguir, descritos conforme apresentado nos documentos da referência bibliográfica.

Aberração: é um defeito de uma imagem óptica causado pela incapacidade de um sistema de lentes formar uma imagem perfeita.

Aerotriangulação: qualquer tipo de triangulação para estender controle por meio de fotografias aéreas.

Alta resolução (imagem): é a imagem de sensoriamento remoto cuja resolução espacial é da ordem de 1 metro (valores no intervalo 0,5 e 5 m).

Altitude: distância vertical sobre o datum a um objeto ou ponto no espaço sobre a superfície terrestre (o datum geralmente é o nível médio dos mares).

Calibração: é o ato ou processo de determinar determinados valores específicos medidos em uma câmara.

Calibração de câmara: é a determinação da distância focal calibrada, da posição do ponto principal em relação às marcas fiduciais, do ponto de simetria, da resolução das lentes, do grau de achatamento do plano focal, da distorção efetiva das lentes no plano focal da câmara, referente à efetiva distância focal calibrada.

Centro da fotografia: centro da imagem fotográfica que é indicado com precisão pela interseção das linhas fiduciais. Em uma câmara perfeita coincide com o ponto principal.

Centro de perspectiva: o ponto de origem ou convergência de conjuntos de raios perspectivos.

Controle: um sistema de medições relativamente exato para determinar as distâncias e as direções de diferenças na elevação entre pontos sobre a Terra.

Controle de campo: controle obtido por levantamentos de campo.

Controle fotogramétrico: controle obtido por processos fotogramétricos.

Coordenadas cartesianas: valores que representam a localização de um ponto num plano ou espaço tridimensional, em relação a dois ou três eixos, respectivamente, perpendiculares entre si.

Coordenadas geocêntricas: coordenadas que definem a posição de um ponto em relação ao centro da Terra, podendo ser cartesianas (x,y,z) ou esféricas (latitude e longitude geocêntricas).

Coordenadas geográficas: o mesmo que coordenadas terrestres e coordenadas geodésicas.

Coordenadas geodésicas: valores de latitude e longitude que definem a posição de um ponto da superfície terrestre em relação ao elipsóide de referência; o mesmo que coordenadas geográficas (Oliveira, 1993).

Datum local: datum que define um elipsóide para uma área limitada da Terra.

Datum: referência física da Terra que define o elipsóide de referência ao qual a superfície física da Terra é aproximada.

Distância focal calibrada: um valor ajustado de distância focal equivalente que distribui o efeito de distorção das lentes sobre toda a área imageada pela câmara.

Distância focal equivalente: a distância medida ao longo dos eixos das lentes desde o ponto nodal ao plano de melhor definição média sobre toda a área imageada pela câmara.

Distância principal: a distância perpendicular do centro interno de perspectiva ao plano da imagem.

Distorção: uma aberração que afeta a posição das imagens fora dos eixos.

Eixo fiducial: as linhas que ligam marcas fiduciais opostas em uma fotografia aérea.

Geóide: uma representação da Terra, cuja superfície é considerada coincidente ao nível dos mares, continuamente estendida através dos continentes. O geóide real é uma superfície equipotencial em que, para cada ponto, o vetor da gravidade é perpendicular.

Marca fiducial: um sinal de referência, rigidamente acoplado ou registrado nas lentes da câmara fotográfica, que forma uma imagem correspondente no negativo, permitindo definir-se o ponto principal da fotografia a partir da geometria do conjunto.

Nadir: ponto imaginário localizado diretamente sobre o observador, diretamente oposto ao zênite.

Orientação: é o processo de reconstituição (ou a determinação) dos parâmetros relacionados à câmara, à geometria espacial da câmara e ao terreno, que permite reconstituir a geometria de imageamento no instante de formação do quadro.

Orientação absoluta: é o ajuste da escala, da posição e da orientação do modelo em relação às coordenadas reais dos objetos localizados numa imagem.

Orientação exterior ou externa: um conjunto de parâmetros que ajustam a posição espacial e angular da câmara em relação ao terreno.

Orientação interior: é a determinação dos parâmetros de reconstituição da fotografia em relação ao cone perspectivo.

Ortofoto: o mesmo que ortoimagem, neste contexto.

Ortofoto verdadeira: o mesmo que ortoimagem verdadeira, neste contexto.

Ortofotografia: o mesmo que ortofoto.

Ortoimagem: imagem em projeção ortográfica, em que cada ponto é visto ortogonalmente.

Ortoimagem verdadeira: ortoimagem ideal, retiradas todas as paralaxes referentes ao relevo.

Paralaxe: deslocamento aparente da posição da imagem de um objeto estacionário causado por um deslocamento do ponto de observação.

Ponto de controle: qualquer estação em um sistema de controle horizontal e/ou vertical, identificado em uma imagem fotográfica e usada para correlacionar os dados mostrados na imagem. Também chamado de ponto de controle de foto, ponto de controle de imagem, e ponto de controle de campo.

Ponto principal: a interseção da perpendicular do centro de perspectiva interior com o plano da fotografia (imagem).

Projeção cartográfica: “traçado sistemático de linhas numa superfície plana, destinado à representação de paralelos de latitude e meridianos de longitude da Terra ou de parte dela”. (Oliveira, 1993)

Sistema cartesiano: sistema retangular de coordenadas geométricas lineares, cujos eixos de referência são perpendiculares entre si.

Projeção perspectiva: a projeção de raios perspectivados em um plano no espaço; o mesmo que projeção geométrica.

Raio perspectivo: uma linha que une o centro perspectivo a um ponto objeto.

APÊNDICE A'

PROGRAMA DE ORTORRETIFICAÇÃO EM IDL

1.1 PROGRAMA DE ORTORRETIFICAÇÃO

```
//=====
// Ortorretificação de Imagens Fotográficas .
// Dissertação de Mestrado INPE .
//=====

//Aluno: Carlos Frederico de Sá Volotão .

//=====

//Contém as seguintes funções e procedimentos:

// JAN0
// CRIAMNE (não utilizando JAN0, utiliza valores para teste)
// MATRIZDAT
// VZOOMWINDOW
// V_ORIENT_MMQ
// ORTOGEN
// DADOS

//=====
PRO JAN0
//=====

//Define o ambiente de janelas

base = WIDGET_BASE(/column, title='JANELA PRINCIPAL')

//janela0=WIDGET_DRAW(base, XSIZE=600, YSIZE=300)

base2 = WIDGET_BASE(base,/row)
  entradaX = CW_FIELD(base2, TITLE='Xo: ',/row,/float, XSIZE=20)
  base3 = WIDGET_BASE(base,/row)
  entradaY = CW_FIELD(base3, TITLE='Yo: ',/row,/float, XSIZE=20)
base4 = WIDGET_BASE(base,/row)
  entradaX = CW_FIELD(base4, TITLE='Tamanho do Pixel: ',/row,/float, XSIZE=20)
base5 = WIDGET_BASE(base,/row)
  entradaCol = CW_FIELD(base5, TITLE='Num de colunas: ',/row,/float, XSIZE=20)
base6 = WIDGET_BASE(base,/row)
  entradaSai = CW_FIELD(base6, TITLE='arquivo de saída MNE: ',/row, XSIZE=20)
base7 = WIDGET_BASE(base,/row)
  vok = WIDGET_BUTTON(base6,VALUE='Ok!')
  vFIM= WIDGET_BUTTON(base6,VALUE='FIM')
  resultados = WIDGET_TEXT(base,YSIZE=30)

//Desenha a janela definida acima

WIDGET_CONTROL, base, /REALIZE

//Executa os comandos

Flag = 1

//Imprime o Texto (resultado) na tela

while Flag do begin
  evento = WIDGET_EVENT(base, BAD_ID = id_mau)
  if id_mau NE 0 then RETURN
```

```

//Se o botão "OK" for pressionado...

if evento.id EQ vok then begin

    //Obtém valor X, Y, Z

    WIDGET_CONTROL, entradaX, get_value = valorX0
    WIDGET_CONTROL, entradaY, get_value = valorY0
    WIDGET_CONTROL, entradaX, get_value = valordx
    WIDGET_CONTROL, entradaCol, get_value = valorcol
    WIDGET_CONTROL, entradasai, get_value = valorsai

    //454863.46, 7386341.62, 1, 2000

    //Define variável do Texto a ser apresentado

    temp2 = ['Xo='+strcompress(valorX0,/rem),$
    ' Yo='+strcompress(valorY0,/rem),$
    ' Tam do Pixel='+strcompress(valordx,/rem),$
    ' Num de Pixel='+strcompress(valorcol,/rem),$
    ' Arquivo saída='+strcompress(valorsai,/rem)]

    //Imprime na tela o Texto

    WIDGET_CONTROL, resultados, set_value = temp2

    //Cria

    CRIAMNE, valorX0, valorY0, valordx, valorcol, ""+valorsai+""

endif

//Se o botão "FIM" for pressionado...

if evento.id EQ vFIM then WIDGET_CONTROL, base, /DESTROY

endwhile

//Fim de JAN0

end

//*****
PRO CRIAMNE
//*****
//Grava um MNE a partir do arquivo DAT

//Cria uma matriz usando parâmetros <Xo,Yo, dx=2.5m, lin=501pixels>

matriz = MATRIZDAT(454863.46, 7386341.62, 2.5, 501)

//Exibe a matriz float como byte

TVSCL, matriz

//Grava a matriz em arquivo TIFF float

texto = 'Crie nome de arquivo para gravar o MNE do tipo TIFF float'
arq = DIALOG_PICKFILE(PATH='c:\Mestrado\Orto',TITLE=texto, FILTER='*.tif' )
write_tiff, arq, matriz, 0, /float

//Fim de CRIAMNE

end

```

```

/*****
FUNCTION MATRIZDAT, X0, Y0, dxy, lin_col
/*****
//Lê arquivo DAT criado pelo ERDAS.
//X0,Y0: coordenadas do centro de projecao
//DX_Y: resolução/espacamento do MNE
//LIN_COL:número de linhas (=colunas) ímpar
//X,Y pode estar em sistema local de coordenadas.
//Esta função devolve 'matriz'

//Garante dxy GE 0

dxy = (dxy GT 0) ? (dxy) : (-dxy)

//Garante que a é ímpar

a = lin_col + 1 - (lin_col MOD 2)

//Cria matriz

matriz = FLTARR(a,a)

//Calcula x,y máximos e mínimos

xmax = X0 + dxy*(a-1d)/2.
xmin = X0 - dxy*(a-1d)/2.
ymax = Y0 + dxy*(a-1d)/2.
ymin = Y0 - dxy*(a-1d)/2.

//Define variáveis iniciais e abre o arquivo de pontos

ponto = [double(0),double(0),double(0)]
texto='Selecione o arquivo DAT existente, contendo os pontos de controle'
arq_mne = DIALOG_PICKFILE(PATH='c:\Mestrado\Orto', TITLE=texto, FILTER='*.dat')
OPENR, Unit, arq_mne, /GET_LUN
flag = 0

//Lê os pontos até o final do arquivo

while not EOF(Unit) do begin
  readf, Unit, ponto
  E = ponto[0]
  N = ponto[1]
  h = ponto[2]
  x = E
  y = N
  z = h
  cond = (x le xmax)*(x ge xmin)*(y le ymax)*(y ge ymin)

//Cria vetor "internos"

  if cond * flag then $
    internos = transpose([[transpose(ponto),transpose(internos)])]
  if cond and not flag then begin
    internos = ponto
    flag = 1
  endif
endwhile

//Termina fechando o arquivo

close, /all
if N_ELEMENTS(internos) eq 0 then print, 'ERRO'

//Para calcular as distâncias aos pontos

x = internos[0,*]
y = internos[1,*]
z = internos[2,*]

```

```

//Calcula linha/col correspondente aos n pontos

m = N_ELEMENTS(x)
print, 'm=', m
i = ROUND(2d/dxy*(x - X0))
j = ROUND(2d/dxy*(y - Y0))

//Define constante de 1/r^k, ponderação pelo inverso da k-ésima pot.

k = 2.0D

//Sendo (g,h) o pixel de cada elemento

for h = 0, a-1 do for g = 0, a-1 do begin
  r = SQRT(((i-g)^2D)+((j-h)^2D))

//Método do Vizinho-mais-próximo

  minr = min(r)
  for t = 0, m-1 do $
    if r[t] eq minr then $
      matriz[g,h]=z[t]

//Método Opcional de ponderação pelo inverso da k-ésima potência

  // Verifica se alguma distância é nula e atribui o valor de Z
  //   flag = 0
  //   numer = 0D
  //   denom = 0D
  //   for t = 0, m-1 do begin
  //     if r[t] eq 0 then begin
  //       matriz[g,h]=z[t]
  //       flag = 1
  //     endif else if not flag then begin
  //       numer=numer+z[t]/(r[t]^k)
  //       denom=denom+1D/(r[t]^k)
  //     endif
  //   endfor
  // Se r <> 0 define a matriz
  //   if not flag then matriz[g,h]=numer/denom

//Fim do Opcional

  endfor

RETURN, matriz

//Término de MATRIZDAT

end

//*****
FUNCTION VZOOMWINDOW, matriz, f=f, n, ttt
//*****
//Se não foi fornecido, o zoom será "1"

  if N_ELEMENTS(f) eq 0 then f=1

//Valores-limite (1 a 8)

  if f le 0 then f = 0.01
  if f gt 8 then f = 8

//Arredonda f para menor inteiro ("floor")

  if f ge 1 then f = BYTE(f) else inv_f = BYTE(1./f)

```

```

//Determina número de linhas e colunas

dimensao = (SIZE(matriz))(0) - 2
if dimensao eq 0 then begin
  lin = N_ELEMENTS(matriz[*,0])
  col = N_ELEMENTS(matriz[0,*])
endif else if dimensao eq 1 then begin
  lin = N_ELEMENTS(matriz[0,*,0])
  col = N_ELEMENTS(matriz[0,0,*])
endif else begin
  Print, "ERRO NA MATRIZ-IMAGEM"
  EXIT
endelse

//Cálculo de parâmetros de normalização
//kz=contraste// ktz=brilho

kz = (MAX(matriz)-MIN(matriz))
kz = 255/ (kz gt 0 ? kz : 1)
ktz= MIN(matriz)

//Dimensiona nova matriz ("zoom")

if NOT dimensao then $
  zoom = (f GE 1) ? $
  BYTARR(f*lin, f*col) : $
  BYTARR(LONG(lin/inv_f),LONG(col/inv_f)) $
else $
  zoom = (f GE 1) ? $
  BYTARR(3,f*lin, f*col) : $
  BYTARR(3,LONG(lin/inv_f),LONG(col/inv_f))

//Amplia f(*f) vezes para cada pixel

if f GE 1 then for i=0, lin-1 do for j=0, col-1 do $
  for l=0, f-1 do for m=0, f-1 do $
    if NOT dimensao then $
      zoom[f*i+l,f*j+m] = BYTE(kz * matriz[i,j] + ktz) $
    else $
      zoom[*,f*i+l,f*j+m] = BYTE(kz * matriz[*,i,j] + ktz) $

//...Ou... Reduz janela inv_f vezes

else for i=0, LONG(lin/inv_f)-1 do $
  for j=0, LONG(col/inv_f)-1 do $
    if NOT dimensao then $
      zoom[i,j]=BYTE(kz *matriz[inv_f*i,inv_f*j] + ktz) $
    else $
      zoom[*,i,j]=BYTE(kz *matriz[*,inv_f*i,inv_f*j] + ktz)

//Apresenta a imagem

WINDOW, n, XSIZE=LONG(f*col), YSIZE=LONG(f*lin), RETAIN=2, TITLE=ttt
WSET, n
TV, zoom, TRUE = (dimensao eq 3 ? 1:0)
FREE_LUN, n
RETURN, 1

//Fim de VZOOMWINDOW

end

```

```

//*****
FUNCTION V_ORIENT_MMQ, csi, eta, X, Y, Z, MP
//*****

//Número de pontos para o MMQ

n = N_ELEMENTS(X)

//Avisa se a unidade parecer ser o pixel,cm ou mm (preferir o metro)

if max(abs(csi) gt 1) then print, "Unidades (csi,eta) não SI"

//Formação da matriz M1 pela concatenação de "M1"

//Cria vetor com 2n elementos nulos (n=número de pontos)

A0 = DINDGEN(n*2)*0.D

//Gera a matriz M1

A0(INDGEN(n)*2)=csi*X
A0(INDGEN(n)*2 + 1)=eta*X
M1 = TRANSPOSE(A0)
A0(INDGEN(n)*2)=csi*Y
A0(INDGEN(n)*2 + 1)=eta*Y
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=csi*Z
A0(INDGEN(n)*2 + 1)=eta*Z
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=X
A0(INDGEN(n)*2 + 1)=0.D
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=Y
A0(INDGEN(n)*2 + 1)=0.D
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=Z
A0(INDGEN(n)*2 + 1)=0.D
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=1.D
A0(INDGEN(n)*2 + 1)=0.D
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=0.D
A0(INDGEN(n)*2 + 1)=X(INDGEN(n))
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=0.D
A0(INDGEN(n)*2 + 1)=Y(INDGEN(n))
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=0.D
A0(INDGEN(n)*2 + 1)=Z(INDGEN(n))
M1 = [M1,TRANSPOSE(A0)]
A0(INDGEN(n)*2)=0.D
A0(INDGEN(n)*2 + 1)=1.D
M1 = [M1,TRANSPOSE(A0)]

//Cria vetor de coordenadas de imagem L0 (matriz-coluna)

Lb=DINDGEN(n*2)
Lb(INDGEN(n)*2)=csi
Lb(INDGEN(n)*2+1)=eta
Lb=TRANSPOSE(Lb) //Coluna

```



```

//Cálculo da matriz de PESOS

WP = IDENTITY(2*n)
for i=0, n-1 do begin
    WP[2*i,2*i] = MP[i]
    WP[2*i+1,2*i+1] = MP[i]
endfor

//M.M.Q considerando a matriz dos pesos relativos (não var-covar)

P = invert( transpose(M1) ## WP ## M1 ) ## transpose(M1) ## WP ## Lb

//Coluna

P = TRANSPOSE(P)

//Cálculo do Erro Médio Quadrático EMQ:

//V = AX-L

V = M1 ## P - Lb
EMQ = TOTAL( (V-TOTAL(V)/N_ELEMENTS(V))^2 )/N_ELEMENTS(V)

//Resultados dos erros

Print,FORMAT='(TR3,"DesvPad[mm]:",1(TR3,F20.3))',SQRT(EMQ)*1000
Print,FORMAT='(TR3,"Resíduos [mm] ",1(TR3,F20.3))',TRANSPOSE(V)*1000

//endif

RETURN, TRANSPOSE(P)

end

```

```

//=====
PRO ORTOGEN
//=====
//Programa principal que faz a orientação e a ortorretificação

//Início do programa

//Inicia a medição do tempo do programa

medetempo = SYSTIME(1)
medetempot= medetempo

//Imprime mensagem de início do programa

Print,FORMAT='(40("=")/,"Início do Processamento"/,40("="))'

//Define nomes e pastas de arquivos a serem lidos

arquivo = 'c:\Mestrado\Orto\invPontos.txt'
arq_mne='c:\Mestrado\Orto\aR_mne.tif'
arquivo_foto = 'c:\Mestrado\Foto.tif'
caminho = 'c:\Mestrado\Orto'

//Define nomes e pastas de arquivos a serem gravados

arquivo_dist = 'c:\Mestrado\Orto\bR_dist.tif'
arquivo_ortor = 'c:\Mestrado\Orto\bOrtor.tif'
arquivo_ortog = 'c:\Mestrado\Orto\bOrtog.tif'
arquivo_ortob = 'c:\Mestrado\Orto\bOrtob.tif'

```

```

//O MNE de entrada, definido em "CRIAMNE" tem o espaçamento::
//FE (Fator de Espaçamento)

FE = 2.50 //Fator de Espaçamento do MNE em metros (centro X0,Y0)

// Abre o arquivo que contém os valores do cód_ponto, csi, eta, X, Y, Z
//cód_ponto do primeiro elemento é o número de elementos do arquivo
//ou seja, p.ex., se for "7" significa que o arquivo precisa ter 7 linhas.

//Abre o arquivo para saber o número de elementos armazenado em (0,0)

OPENR, 1, arquivo
READF, 1, i //Primeiro elemento é o "número de linhas" (pontos)
CLOSE, 1

//Cria um vetor com dimensão dos dados do arquivo

vetor=DBLARR(6,i,/NOZERO) //6 é o número de colunas (variáveis)

//Agora abre e lê o arquivo

OPENR, 1, arquivo
READF, 1, vetor
CLOSE, 1

//"vetor" é o vetor dos pontos

n = vetor[0,0]

//MP é a matriz dos pesos// HV pesa 'PESO' vezes os demais

PESO = 20
MP = TRANSPOSE (vetor[0,*])
MP = MP/n*(PESO-1)+1

//csi,eta são coordenadas de imagem (orientado origem: superior-esquerda)

csi=TRANSPOSE (vetor[1,*])
eta=TRANSPOSE (vetor[2,*])

//Inverte Eixo Y e translada para centro = 0,0

ok = QUERY_TIFF(arquivo_foto, s)
col0 = s.DIMENSIONS[0]
lin0 = s.DIMENSIONS[1]
TR_X = (col0-1)/2
TR_Y = (lin0-1)/2
csi = csi - TR_X
eta = TR_Y - eta

//Novos (XnYnZ) com Xn,Yn translados para ter média ZERO

unitX = MEAN(vetor[3,*])
unitY = MEAN(vetor[4,*])

//Serão utilizados nos cálculos (XnYnZ) e depois somados unitX e unitY

Xn = TRANSPOSE (vetor[3,*]-unitX)
Yn = TRANSPOSE (vetor[4,*]-unitY)
Z = TRANSPOSE (vetor[5,*])

//Unidade (csi,eta) em metros// passar pixel para metros (21 micra=0.000021)

RES_SCAN_FOTO = 0.000028D
csi = RES_SCAN_FOTO*csi
eta = RES_SCAN_FOTO*eta

//Calcula o MMQ e define as variáveis conforme teoria

P= V_ORIENT_MMQ (csi, eta, Xn, Yn, Z, MP)
print

```

```

print, TRANSPOSE(P)
print

//Define novos nomes para as variáveis

A = P[0] & B = P[1] & C = P[2]
D1= P[3] & D2= P[7]
E1= P[4] & E2= P[8]
F1= P[5] & F2= P[9]
G1= P[6] & G2= P[10]

//Calcula os Parâmetros de Orientação (permite angulos de até 90 graus)

ABC2 = DOUBLE(A*A+B*B+C*C)
ABC = SQRT(ABC2)
d=-1./ABC

//(csi,eta)

csi0= -(A*D1+B*E1+C*F1)/ABC2
eta0= -(A*D2+B*E2+C*F2)/ABC2

//fi,w,k

fi = ASIN(A/(ABC)) //impossivel que ABC = 0
w = (C ne 0) ? ATAN(-B/C) : !PI/2 //C=0 => w=+-90°
k = (D1+A*csi0 ne 0) ? ATAN((-D2-A*eta0),(D1+A*csi0)) : !PI/2

// Considerando que fi e w são menores que 90°, k pode variar de -180 a 180

//Então, se o valor calculado da distancia focal for negativo, i.e.:

t1= -d*(D1+A*csi0)/(cos(fi)*cos(k))

//O ângulo k está do outro lado do círculo trigonométrico:

if t1 lt 0 then k = k + !PI

//Cálculo da Matriz R (transposta porque o IDL usa coluna-linha)

R = DBLARR(3,3)
R[0,0] = cos(fi)*cos(k)
R[1,0] = cos(w) *sin(k) + sin(w)*sin(fi)*cos(k) // coluna 1, r11, r12, r13
R[2,0] = sin(w) *sin(k) - cos(w)*sin(fi)*cos(k)
R[0,1] = -cos(fi)*sin(k)
R[1,1] = cos(w) *cos(k) - sin(w)*sin(fi)*sin(k) // coluna 2
R[2,1] = sin(w) *cos(k) + cos(w)*sin(fi)*sin(k)
R[0,2] = sin(fi)
R[1,2] = -sin(w) *cos(fi) // coluna 3
R[2,2] = cos(w) *cos(fi)

r11=R[0,0]
r21=R[1,0]
r31=R[2,0]
r12=R[0,1]
r22=R[1,1]
r32=R[2,1]
r13=R[0,2]
r23=R[1,2]
r33=R[2,2]

//Transpondo R para seguir as convenções utilizadas

R = TRANSPOSE(R)

//Dist focal

cf=(A*csi0+D1)/((COS(fi)*COS(k))*ABC)

//Calcula novamente t1 e os demais valores de cf

```

```

t1= -d*(D1+A*csi0)/r11
t2= -d*(D2+A*eta0)/r12
t3= -d*(E1+B*csi0)/r21
t4= -d*(E2+B*eta0)/r22
t5= -d*(F1+C*csi0)/r31
t6= -d*(F2+C*eta0)/r32

//Cálculo de cf (média dos ti acima)

cf = (t1+t2+t3+t4+t5+t6)/6

//Apresenta os valores de distancia focal ao usuário

print, 'Valores de "c" devem estar coerentemente próximos'
print, t1,t2,t3,t4,t5,t6 & print

//Variável auxiliar para o cálculo de Xo, Yo, Zo

sp1 = (csi0-G1) / (cf*ABC)
sp2 = (eta0-G2) / (cf*ABC)
sp3 = 1D / ABC
SP=[sp1,sp2,sp3]

//Calcula Xo,Yo,Zo

PP = R ## SP

//Dá outro nome às variáveis

X0=PP[0]+unitX & Y0=PP[1]+unitY & Z0=PP[2]

//Apresenta valores obtidos

//Utilizando as unidades da câmara

Print, "Utilizando resolução de 28 micra"
Print, "      f[mm]      =", cf*1000D
Print, "      csi0[mm]=", csi0*1000D
Print, "      eta0[mm]=", eta0*1000D

//Apresenta os ÂNGULOS da orientação

Print, 'Orientação exterior, ângulos de rotação entre sistemas:
Print, FORMAT = ' (TR5,A5,F14.4/,TR5,A5,F14.4/,TR5,A5,F14.4)', "fi = ", $
      fi*180/!PI, "w = ", w*180/!PI, "k = ", k*180/!PI

//Xo,Yo,Zo:

Print, FORMAT=' (TR5,"Xo =",F20.4/TR5,"Yo =",F20.4/TR5,"Zo =",F20.4)', $
      X0,Y0,Z0

//Se ocorrer X ou Y ou Z negativo

if ((X0<Y0<Z0) lt 0) then Print, "Valor não deveria ser Negativo"

//Armazenar o resultado apresentado acima em uma matriz

resultado=strarr(6)
resultado[0]=' Dimensão do pixel = 28 micra'
resultado[1]=' f[mm]=' +string(cf*1000D)
resultado[2]=' csi0[mm]=' +string(csi0*1000)
resultado[3]=' eta0[mm]=' +string(eta0*1000)
resultado[4]=strcompress('fi=' +string(fi*180/!PI) + $
      'w=' +string(w*180/!PI) + 'k=' +string(k*180/!PI))
resultado[5]=string(X0) +string(Y0) +string(Z0)

//Sinaliza o final da parte de orientação

Print,FORMAT=' (40("=")/,"PARTE 1",F10.2," seg (da fase)"/,40("="))', $
      (SYSTIME(1) - medetempo) &      medetempo = SYSTIME(1)

```

```

//=====
//ORTORRETIFICAÇÃO
//=====

//O pixel central do MAT_MNE (e, conseqüentemente, de MAT_ORTO e MAT_DIST)
//deve ser exatamente o ponto X0,Y0.
//Deve-se executar a rotina,
//verificar o valor de X0,Y0,Z0, e
//gerar o modelo do terreno.

//Abre a foto (MAT_FOTO a ser ortorretificada)

MAT_FOTO = READ_TIFF(arquivo_foto)

//Verifica o número de linha e colunas e se é colorida ou não

colorida = (SIZE(MAT_FOTO))(0) - 2

//Se for colorida, ler as bandas (vermelha,verde,azul)

if colorida then begin
MAT_FOTOr = TRANSPOSE(MAT_FOTO[0,*,*])
MAT_FOTOg = TRANSPOSE(MAT_FOTO[1,*,*])
MAT_FOTOb = TRANSPOSE(MAT_FOTO[2,*,*])
MAT_FOTOr = ROTATE(MAT_FOTOr,3)
MAT_FOTOg = ROTATE(MAT_FOTOg,3)
MAT_FOTOb = ROTATE(MAT_FOTOb,3)
endif

//Obtém número de linhas e colunas da imagem original

col0 = N_ELEMENTS(MAT_FOTOr[*,0])
lin0 = N_ELEMENTS(MAT_FOTOr[0,*])

//Abre o modelo numérico de elevação

MAT_MNE = READ_TIFF(arq_mne) //Deve ser FLOAT
col = N_ELEMENTS(MAT_MNE[*,0])
lin = N_ELEMENTS(MAT_MNE[0,*])

//Informa execução terminada

Print,FORMAT='(40("=")/,"PARTE 2 =",F10.2," s (fase)"/,40("="))', $
(SYSTIME(1) - medetempo) & medetempo = SYSTIME(1)

// Dimensiona as matrizes da ortoimagem e distancia

//Cria Matriz nula com mesmas dimensoes de 'MAT_MNE'

MAT_ORTOr = BYTARR(col,lin)
MAT_ORTOg = BYTARR(col,lin)
MAT_ORTOb = BYTARR(col,lin)
MAT_DIST = FLTARR(col,lin)

//Constantes

//Cria variaveis auxiliares k01, k02, k03

X00=X0-unitX
Y00=Y0-unitY

K01 = r11*X00 + r21*Y00 + r31*Z0
K03 = r13*X00 + r23*Y00 + r33*Z0
K02 = r12*X00 + r22*Y00 + r32*Z0

//1/Escala (significativa se unidades Z0 e cf uniformes [m])

Zscf = Z0 / cf

//Define o valor central de (xi0,yi0) e desloc nulo p/ pixel central

```

```

xi0 = LONG( ((lin < col) - 1) / 2 ) //Equivale a X0
yi0 = LONG( ((lin < col) - 1) / 2 ) //Equivale a Y0
MAT_DIST[xi0,yi0] = 0.

//O primeiro ponto é o ponto NADIR (se denominador não nulo)

if r33 ne 0 then begin
  csi = csi0 - cf * r31/r33
  eta = eta0 - cf * r32/r33
  icsi= TR_X + csi/RES_SCAN_FOTO //em pixel
  ieta= TR_Y - eta/RES_SCAN_FOTO //em pixel
  MAT_ORTOr[xi0,yi0] = MAT_FOTOr[icsi,ieta]
  MAT_ORTOg[xi0,yi0] = MAT_FOTOg[icsi,ieta]
  MAT_ORTOb[xi0,yi0] = MAT_FOTOb[icsi,ieta]
endif

//Informa execução terminada

Print,FORMAT='(40("=")/,"PARTE 3 =",F10.2," s (fase)"/,40("="))', $
  (SYSTIME(1) - medetempo) &      medetempo = SYSTIME(1)

//=====
//OCCLUSÃO
//=====
//Considerando as áreas ocultas, verificar cada pixel:
//ka é a distancia minima do quadrado a cada lado

for ka = 1, LONG(((lin < col) - 1) / 2) do begin

  nivel = 0
  i = ka
  j = -ka + 1
  while (nivel LT 4) do begin

    //Pontos de imagem do quadrado externo

    xie = xi0 + i
    yie = yi0 + j

    //Pontos de imagem do quadrado interno

    //Caso ponto sobre o lado superior ou inferior

    if nivel MOD 2 EQ 0 then begin
      xii = xie - i/ABS(i)
      yii = yie - ROUND(FLOAT(j)/ka)

    //Caso ponto sobre o lado superior ou inferior

    end else begin
      xii = xie - ROUND(FLOAT(i)/ka)
      yii = yie - j/ABS(j)
    endelse

    //Cálculo de Xt,Yt,Zt (coordenadas do terreno correspond a xie,yie)
    //OBS: sem unitX

    Xt = (xie - xi0) * FE + (X00 + unitX)
    Yt = (yie - yi0) * FE + (Y00 + unitY)
    Zt = MAT_MNE [ xie, yie ]

    //Para calcular a distância ao ponto nadiral (metros)

    d0x = Z0 * (Xt - X0)/(Z0 - Zt)
    d0y = Z0 * (Yt - Y0)/(Z0 - Zt)

```

```

//Distância do ponto nadiral (m) ao pixel contendo valor desejado

distn = SQRT(d0x^2 + d0y^2)

//Caso de não ocorrer oclusão, desenha pixel se visível
// (ie, se a distn do ponto interno corresp é menor que a do outro)

if distn GE MAT_DIST[xii,yii] then begin

//Valor da distância é o armazenado na matriz

MAT_DIST[xie,yie] = distn

//Calcula (csi,eta) do ponto donde se obtém a informação
// do ponto (xie,yie)

Xt0 = Xt - unitX
Yt0 = Yt - unitY

csi= csi0 - cf*((r11*Xt0+r21*Yt0+r31*Zt-K01)/$
(r13*Xt0+r23*Yt0+r33*Zt-K03))
eta= eta0 - cf*((r12*Xt0+r22*Yt0+r32*Zt-K02)/$
(r13*Xt0+r23*Yt0+r33*Zt-K03))

//Transformar para coordenadas de imagem (e unidades para pixel)

icsi= TR_X + csi/RES_SCAN_FOTO //em pixel
ieta= TR_Y - eta/RES_SCAN_FOTO //em pixel
csi=round(icsi)
eta=round(ieta)

//Se (csi,eta) estiver contido na MAT_FOTO, ler

if (csi GE 0) and (csi LT col0) and $
(eta GE 0) and (eta LT lin0) then $
begin
MAT_ORTOOr[xie,yie] = MAT_FOTOOr[csi,eta]
MAT_ORTOG[xie,yie] = MAT_FOTOG[csi,eta]
MAT_ORTOB[xie,yie] = MAT_FOTOB[csi,eta]
endif

//Caso de ocorrer oclusão

endif else begin

//Valor de fundo arbitrado como "0"

MAT_ORTOOr[xie,yie] = 0
MAT_ORTOG[xie,yie] = 0
MAT_ORTOB[xie,yie] = 0
MAT_DIST[xie,yie] = MAT_DIST[xii,yii]

endelse

//Dentro do "while", determinar se muda o nivel
case nivel of
0: if j LT ka then j = j + 1 else nivel = 1
1: if i GT -ka then i = i - 1 else nivel = 2
2: if j GT -ka then j = j - 1 else nivel = 3
3: if i LT ka then i = i + 1 else nivel = 4
endcase
endwhile
endfor

//Informa execução terminada

Print,FORMAT='(40("=")/,"PARTE 4 =",F10.2," s (fase)"/,40("="))',$(
(SYSTIME(1) - medetempo) & medetempo = SYSTIME(1)

//Grava os arquivos MNE, ORTO em um único e grava RESULTADO

```

```

WRITE_TIFF, arquivo_dist, MAT_DIST, COMPRESSION=0 //Float
WRITE_TIFF, arquivo_ortor, MAT_ORTOR, COMPRESSION=0 //Byte
WRITE_TIFF, arquivo_ortog, MAT_ORTOG, COMPRESSION=0 //Byte
WRITE_TIFF, arquivo_ortob, MAT_ORTOB, COMPRESSION=0 //Byte

//Exibe a imagem

//ok = VZOOMWINDOW(MAT_ORTOR, f=1, 2, 'ORTO')

//Mensagem final
print, FORMAT='(40("=")/"Ortorretificação concluída"/40("="))'
print, FORMAT='(40("=")/"Término NORMAL do programa"/40("="))'

//Avisa do término

Print, "Usados os dados de ", arquivo
Print, FORMAT='(40("=")/"FIM ", F10.2, "s fase")', (SYSTIME(1) - medetempo)
Print, FORMAT='(F10.2, "s total"/, 40("="))', (SYSTIME(1) - medetempot)
BEEP

//Fim do Programa
stop
end

//*****
PRO DADOS
//*****

//Descreve os dados da imagem fornecida pela Funcate (utilizada acima)

Texto = ['Imagem fotogramétrica de CARAGUATATUBA']
Texto = [[Texto], [' Data do voo: ']]
Texto = [[Texto], [' Escala da foto -> 1 / 8000']]
Texto = [[Texto], [' Resolução de leitura no scanner -> 28 micrometros']]
Texto = [[Texto], [' Arquivo "11-016.tif", fornecido pela FUNCATE']]
Texto = [[Texto], [' Tamanho do arquivo em kB -> 206433 kB']]
Texto = [[Texto], [' Fotografia colorida 3 bandas (3 x 8-bits)']]
Texto = [[Texto], [' Dados voo: BASE S.A., Faixa 11, num 16']]
Texto = [[Texto], [' Dados da camara: marca ZEISS']]
Texto = [[Texto], [' Distancia focal aferida: [mm]']]
Texto = [[Texto], [' F = 152.755']]
Texto = [[Texto], [' Posicao aproximada do Nadir:']]
Texto = [[Texto], [' Lon = 045°26'35"W']]
Texto = [[Texto], [' Lat = 23°37'59"S']]

Print, Texto

Orien = ['Parametros de Orientacao']
Orien = [[Orien], [' 1) "HV-24" E=454230.54 N=7386866.59 h=13.75']]
Orien = [[Orien], [' 2) "HV-32" E=455582.04 N=7386506.25 h=03.18']]
Orien = [[Orien], [' 3) "HV-23" E=454093.23 N=7386241.19 h=08.37']]
Orien = [[Orien], [' 4) "PT1532" E=455898.24 N=7385742.28 h=04.84']]
Orien = [[Orien], [' 5) "PT1530" E=454649.04 N=7386344.19 h=10.88']]
Orien = [[Orien], [' 6) "PT1525" E=454411.08 N=7385396.69 h=04.54']]
Orien = [[Orien], [' 7) "PT2546" E=455251.86 N=7387197.00 h=13.53']]
Orien = [[Orien], ['Projeção South American Datum 1969 zona 23']]
Orien = [[Orien], ['Valores relacionados na imagem em pixels (respect):']]
Orien = [[Orien], [' 1) x= 919 y=6861']]
Orien = [[Orien], [' 2) x=4349 y=1167']]
Orien = [[Orien], [' 3) x=3578 y=7762']]
Orien = [[Orien], [' 4) x=7854 y= 843']]
Orien = [[Orien], [' 5) x=3857 y=5241']]
Orien = [[Orien], [' 6) x=7555 y=7442']]
Orien = [[Orien], [' 7) x=1043 y=1634']]

```



```

Orien = [[Orien],['Parametros de Orientacao']]
Orien = [[Orien],[' Orientação Interior']]
Orien = [[Orien],[' f = 152.755 mm']]
Orien = [[Orien],[' dcsi= 0.005 mm']]
Orien = [[Orien],[' deta= -0.001 mm']]
Orien = [[Orien],[' Orientação Absoluta']]
Orien = [[Orien],[' fi = -1.6610']]
Orien = [[Orien],[' w = -0.2062']]
Orien = [[Orien],[' k = -73.2049']]
Orien = [[Orien],[' Xo = 454863.459']]
Orien = [[Orien],[' Yo = 7386341.624']]
Orien = [[Orien],[' Zo = 1253.707']]

Print, Orien
Print, '(dcsi,deta) é o erro em relação ao centro "fiducial", medido:'
Print, ' centro fiducial = (4205.5, 4189.5) [em pixels]'
Print, ' erro = (5,-1) [em micrometros]'
Print, ' erro = (0.18,0.04) [em pixels]'
Print, ' erro ~ (0,0) [em pixels]'
Print, ' centro da imagem = (4205,4188) [em pixels]'
end

```

1.2 IDENTIFICAÇÃO DOS PONTOS NA IMAGEM

Acrescentando-se as linhas abaixo ao programa em IDL, no início do procedimento “ORTO”, cria-se uma janela em que a imagem é apresentada e pode-se identificar os pontos de controle com o uso do *mouse*.

```
//Verifica se o usuario quer selecionar os pontos ou usar os antigos valores
    repeat READ, opcao, PROMPT='Opcao 1=gravar//2=ler' until opcao eq 1 or opcao eq 2
//Deseja selecionar os pontos na tela e fornecer XYZ de terreno
    if opcao eq 1 then begin
        vetor = DBLARR(6)
        X = 0. & Y = 0. & Z = 0.
//Contador de pontos mínimos para saída da rotina
        flag = 0
        while flag lt 6 do begin
            repeat begin
                //Obtém segundo ponto
                CURSOR, cur_x, cur_y, /DOWN, /DEVICE
                if (!MOUSE.button ne 4) then $
                //Se ponto pertence à imagem
                if (cur_x ge 0 and cur_y ge 0) and (cur_x lt col and cur_y lt lin) $
                then begin
                    Print, 'Para o ponto',flag+1, ' :'
                    //Solicita ao usuário o valor de X, Y e Z
                    READ, X, PROMPT= 'Digite X terreno'
                    READ, Y, PROMPT=' Digite Y terreno '
                    READ, Z, PROMPT=' Digite Z terreno '
                    if flag eq 0 then vetor = [flag, cur_x, cur_y, X, Y, Z] else $
                    vetor = [[vetor],[flag, cur_x, cur_y, X, Y, Z]]
                    Print, 'Escolha outro ponto ou pressione botão direito'
                    flag = flag + 1
                    endif
                //Até pressionar o botão direito
                endrep until (!MOUSE.button eq 4)
                if flag lt 6 then Print, 'Faltam, pelo menos,',6 - flag,'pontos.' $
                else Print, 'Foram obtidos',flag,'pontos'
            endwhile
            vetor[0,0]=flag
            OPENW, 1, 'vetores.fre'
            PRINTF, 1,vetor
            CLOSE, 1
            endif $
```

```
        //Caso deseje utilizar um arquivo já criado...

        else begin
OPENR, 1, arquivo
READF, 1, i
CLOSE, 1
vetor=DEBLARR(6,i,/NOZERO)
OPENR, 1, arquivo
READF, 1, vetor
CLOSE, 1
Endelse
```