

# UERJ

Dissertação de Mestrado em Engenharia de Computação  
Área de concentração em Geomática

**Uma Métrica para Dimensionamento de *Software* Científico  
Aplicada à Fotogrametria Digital**

Autor: Daniel Lessa Bastos

Orientador: Orlando Bernardo Filho

Co-orientador: Jorge Luís Nunes e Silva Brito

Programa de Pós Graduação em Engenharia de Computação  
Área de Concentração em Geomática

Março – 2007



Faculdade de Engenharia

# **Uma Métrica para Dimensionamento de *Software* Científico Aplicada à Fotogrametria Digital**

**Daniel Lessa Bastos**

Dissertação submetida ao corpo docente da Faculdade de Engenharia da Universidade do Estado do Rio de Janeiro – UERJ, como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia de Computação.

Orientador: Orlando Bernardo Filho

Co-orientador: Jorge Luís Nunes e Silva Brito

Programa de Pós Graduação em Engenharia de Computação – Área de Concentração em Geomática

Rio de Janeiro  
Março de 2007

BASTOS, DANIEL LESSA

Uma Métrica para Dimensionamento de  
*Software* Científico Aplicada à Fotogrametria  
Digital [Rio de Janeiro] 2007

xix, 103 p. 29,7 cm, (FEN/UERJ, M.Sc.,  
Engenharia de Computação – Área de Concentração  
Geomática, 2007)

Dissertação – Universidade do Estado do Rio de  
Janeiro, FEN

1. Engenharia de *Software*
2. Métricas de *Software*
3. Análise de Pontos de Função
4. Cartografia
5. Fotogrametria Digital

I. FEN/UERJ II. Título

## Folha de Julgamento

Título: Uma Métrica para Dimensionamento de *Software* Científico Aplicada à Fotogrametria Digital

Candidato: Daniel Lessa Bastos

Programa: Pós-Graduação em Engenharia de Computação – Área de Concentração em Geomática

Data da Defesa: 15/03/2007

Aprovada por:

---

Orientador: Orlando Bernardo Filho, D.Sc., UERJ

---

Co-orientador: Jorge Luís Nunes e Silva Brito, Ph.D., UERJ

---

Guilherme Lúcio Abelha Mota, D.Sc., UERJ

---

Asterio Kiyoshi Tanaka, Ph.D., UNIRIO

Aos meus pais

À meu irmão

À minha namorada

Aos meus amigos

## **AGRADECIMENTOS**

Aos meus pais, Alda e Sérgio por todo o carinho, amor e apoio para seguir com este projeto.

A Universidade do Estado do Rio de Janeiro por me dar a oportunidade de realizar este projeto sem qualquer ônus financeiro.

Ao meu Orientador, professor Orlando, pela atenção, compreensão e otimismo com que conduziu este trabalho.

Ao meu co-orientador, professor Nunes, pelo apoio e presteza.

À toda equipe do projeto E-FOTO, em especial aos pesquisadores Marcelo e Guilherme, pela compreensão e ajuda na obtenção de resultados.

Aos meus amigos, meu irmão e namorada pelo apoio e compreensão nos momentos difíceis.

Resumo da dissertação apresentada à FEN/UERJ como parte dos requisitos necessários para obtenção do grau de Mestre em Ciências (M. Sc.)

**Uma Métrica para Dimensionamento de *Software* Científico  
Aplicada à Fotogrametria Digital**

Daniel Lessa Bastos

MARÇO/2007

Orientador: Orlando Bernardo Filho

Co-orientador: Jorge Luís Nunes e Silva Brito

Programa de Pós Graduação em Engenharia de Computação –  
Área de Concentração em Geomática

Este trabalho consiste na elaboração de uma proposta de métrica para dimensionamento de esforço de desenvolvimento de *software* voltado para a área da Fotogrametria Digital. Esta proposta de métrica procura contribuir para que se possa estimar, de maneira mais precisa, o tempo de criação de *software* que possua funcionalidades focadas no processamento de cálculos científicos, o que geralmente ocorre em programas da área da fotogrametria digital. Utilizando como base a métrica de pontos de função, mantida e divulgada pelo *International Function Point Users Group* (IFPUG) e largamente utilizada atualmente pelo mercado de desenvolvimento de *software* brasileiro, este trabalho propõe a criação de funções que possuam elementos típicos do cálculo numérico. Ao final do trabalho, a métrica proposta é aplicada em um projeto de *software* da área da fotogrametria digital com base em uma relação, e um modelo de custo. Os resultados de sua aplicação são analisados ao final do trabalho.

Palavras-Chave: Métrica, Engenharia de *Software*, Fotogrametria Digital, Cálculo Numérico, Cálculo Científico, Pontos de Função, Métricas de *Software*.

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

## **A Metric to Measure Scientific Software Applied to Digital Photogrammetry Field**

Daniel Lessa Bastos

MARCH/2007

Advisor: Orlando Bernardo Filho

Co-Advisor: Jorge Luís Nunes e Silva Brito

Computer Engineering Program – Geomatic

This work presents a metric proposal for sizing the software development effort in the digital photogrammetry field. This metric proposal aims at contributing to estimate, in a more precise way, the development effort for implementing software focused on scientific calculus processing, which normally occurs in the digital photogrammetry software. Based on the function point metric, which is maintained and divulged by IFPUG (International Function Point Users Group) and largely used by the Brazilian software development market, the present work considers the creation of functions that have typical elements of numeric calculus. By the end of this work, the metric proposal is applied, following two different cost models, on a software project in the area of digital photogrammetry, the results are shown, being debated in the conclusion section.

Keywords: Digital Photogrammetry, Software Engineering, Software Metrics, Point Function, Scientific Calculus.



# SUMÁRIO

ÍNDICE DE TABELAS .....	XV
LISTA DE SIGLAS .....	XVIII
<b>CAPÍTULO I - INTRODUÇÃO .....</b>	<b>1</b>
I.1 APRESENTAÇÃO .....	1
I.2 OBJETIVO.....	2
I.3 ORGANIZAÇÃO.....	3
<b>CAPÍTULO II – FUNDAMENTOS TEÓRICOS .....</b>	<b>4</b>
II.1 INTRODUÇÃO .....	4
II.2 MÉTRICAS E FORMAS DE MEDIÇÃO DE CUSTO PARA DESENVOLVIMENTO DE SOFTWARE.....	4
II.2.1 LINHAS DE CÓDIGO .....	5
II.2.2 MARK II.....	6
II.2.3 COSMIC FFP .....	6
II.2.4 COCOMO II .....	7
II.2.5 MEDIÇÃO DE CUSTO POR RELAÇÃO HOMEM X HORA.....	13
II.2.6 CONSIDERAÇÕES SOBRE AS MÉTRICAS DE SOFTWARE .....	14
II.3 ANÁLISE DE PONTOS DE FUNÇÃO .....	14
II.3.1 CONTAGEM DOS PONTOS DE FUNÇÃO .....	15
II.3.2 CONSIDERAÇÕES SOBRE A ANÁLISE DE PONTOS DE FUNÇÃO .....	28
II.4 FOTOGRAMETRIA .....	29
II.4.1 UM BREVE HISTÓRICO ATÉ A FOTOGRAMETRIA DIGITAL.....	30
II.4.2 PORQUE UTILIZAR A FOTOGRAMETRIA.....	33
II.4.3 AQUISIÇÃO DE IMAGENS .....	34
II.4.4 UTILIZAÇÃO DE SENSORES ORBITAIS.....	35
II.4.5 ORIENTAÇÃO INTERIOR .....	36
II.4.6 ORIENTAÇÃO EXTERIOR .....	38
II.4.7 FOTOTRIANGULAÇÃO.....	39
II.4.8 OBTENÇÃO DO MODELO DIGITAL DE SUPERFÍCIE.....	39
II.4.9 RETIFICAÇÃO DE IMAGENS.....	41
II.4.10 ORTORETIFICAÇÃO DE IMAGENS.....	42
II.4.11 RESTITUIÇÃO DIGITAL .....	43
II.4.12 CONSIDERAÇÕES SOBRE FOTOGRAMETRIA .....	45
II.5 CÁLCULO MATEMÁTICO APLICADO À FOTOGRAMETRIA.....	45
II.5.1 OPERAÇÕES BÁSICAS SOBRE MATRIZES .....	46

II.5.2 TRANSFORMAÇÕES.....	49
II.5.2.1 TRANSFORMAÇÃO AFIM.....	49
II.5.2.2 TRANSFORMAÇÃO AFIM ISOGONAL.....	51
II.5.2.3 TRANSFORMAÇÃO PROJETIVA.....	52
II.5.3 CONSIDERAÇÕES SOBRE O CÁLCULO APLICADO A FOTOGRAMETRIA .....	53
II.6 COMENTÁRIOS.....	53
<b>CAPÍTULO III – PROJETO E-FOTO .....</b>	<b>54</b>
III.1 INTRODUÇÃO.....	54
III.2 ESTAÇÃO FOTOGRAMÉTRICA DIGITAL .....	55
III.2.1 ITENS DE HARDWARE.....	56
III.2.2 MÓDULOS DE SOFTWARE DEDICADOS A OPERAÇÕES FOTOGRAMÉTRICAS.....	57
III.3 METODOLOGIA UTILIZADA PARA IMPLEMENTAÇÃO DO E-FOTO.....	58
III.4 SITUAÇÃO ATUAL DO PROJETO .....	65
III.5 MÓDULOS IMPLEMENTADOS.....	66
III.5.1 DEFINIÇÃO DE PARÂMETROS DO E-FOTO.....	67
III.5.2 CALIBRAÇÃO DE CÂMARA .....	67
III.5.3 ORIENTAÇÃO INTERIOR.....	68
III.5.4 MEDIÇÃO DIGITAL.....	69
III.5.5 MEDIÇÃO E VISUALIZAÇÃO ESTEREOSCÓPICA .....	70
III.5.6 RETIFICAÇÃO DE IMAGENS DIGITAIS.....	71
III.6 COMENTÁRIOS .....	72
<b>CAPÍTULO IV – PROPOSTA DE MÉTRICA .....</b>	<b>73</b>
IV.1 INTRODUÇÃO.....	73
IV.2 A ANÁLISE DE PONTOS DE FUNÇÃO E PROJETOS DE SOFTWARE FOCADOS NA ELABORAÇÃO DE CÁLCULOS CIENTÍFICOS.....	73
IV.3 DADOS UTILIZADOS NA PROPOSTA DE MÉTRICA .....	75
IV.4 UMA PROPOSTA DE MÉTRICA PARA MEDIÇÃO DE SOFTWARE FOCADO NA IMPLEMENTAÇÃO DE PROJETOS DE SOFTWARE NA ÁREA DA FOTOGRAMETRIA DIGITAL.....	77
IV.4.1 CRIAÇÃO DE FUNÇÕES TIPO CÁLCULO MATEMÁTICO .....	77
IV.4.2 REGRAS DE CONTAGEM PARA AS FUNÇÕES PROPOSTAS .....	78
IV.4.3 TABELAS DE COMPLEXIDADE.....	78
IV.4.4 DETERMINAÇÃO DA CONTRIBUIÇÃO .....	79
IV.4.5 O PROCESSO DE CONTAGEM.....	81
<b>CAPÍTULO V – APLICAÇÃO E APRESENTAÇÃO DE RESULTADOS .....</b>	<b>83</b>
V.1 INTRODUÇÃO.....	83
V.2 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE RETIFICAÇÃO DE ACORDO COM A PROPOSTA DO IFPUG.....	83

V.3 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE RETIFICAÇÃO COM BASE NA PROPOSTA DESTA DISSERTAÇÃO .....	84
V.4 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE ORIENTAÇÃO INTERIOR DE ACORDO COM A PROPOSTA DO IFPUG .....	86
V.5 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE ORIENTAÇÃO INTERIOR COM BASE NA PROPOSTA DESTA DISSERTAÇÃO .....	87
V.6 APLICAÇÃO DO MODELO DE CUSTO COCOMO II SOBRE A PROPOSTA DO IFPUG.....	88
V.7 APRESENTAÇÃO DE RESULTADOS .....	90
V.8 COMENTÁRIOS .....	93
<b>CAPÍTULO VI – CONCLUSÃO .....</b>	<b>94</b>
<b>REFERÊNCIAS .....</b>	<b>98</b>

## **ÍNDICE DE FIGURAS**

<b>FIGURA II.1 – INTERFACE DA FERRAMENTA USC COCOMO II 1999.0 ....</b>	<b>12</b>
<b>FIGURA II.2 – PROCESSO DE CONTAGEM DOS PONTOS DE FUNÇÃO.....</b>	<b>15</b>
<b>FIGURA II.3 – RELACIONAMENTO ENTRE OS TIPOS DE CONTAGEM....</b>	<b>17</b>
<b>FIGURA II.4 – AQUISIÇÃO DE IMAGENS ATRAVÉS DE SENSORES ÓPTICOS IMPLANTADOS EM UM AVIÃO – AEROFOTOGRAMETRIA.....</b>	<b>30</b>
<b>FIGURA II.5 – RESTITUIDOR ANALÓGICO FONTE: UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO.....</b>	<b>31</b>
<b>FIGURA II.6 – PROCESSO DE RECONSTRUÇÃO TRIDIMENSIONAL DO TERRENO ATRAVÉS DO COMPUTADOR.....</b>	<b>33</b>
<b>FIGURA II.7 – SUPERPOSIÇÃO DE FOTOS AÉREAS.....</b>	<b>35</b>
<b>FIGURA II.8 – AQUISIÇÃO DE IMAGENS POR SENSORES ORBITAIS.....</b>	<b>36</b>
<b>FIGURA II.9 – ORIENTAÇÃO INTERIOR – RECONSTRUÇÃO DA GEOMETRIA DO FEIXE PERSPECTIVO .....</b>	<b>37</b>
<b>FIGURA II.10 – PARÂMETROS A SEREM OBTIDOS NA ORIENTAÇÃO EXTERIOR.....</b>	<b>38</b>
<b>FIGURA II.11 – EXTRAÇÃO ATRAVÉS DA UTILIZAÇÃO DE MESA DIGITALIZADORA.....</b>	<b>40</b>
<b>FIGURA II.12 – COMPARAÇÃO ENTRE IMAGEM ORIGINAL E IMAGEM RETIFICADA.....</b>	<b>41</b>
<b>FIGURA II.13 – IMAGEM ADQUIRIDA ATRAVÉS DE CÂMARA – DEVE-SE NOTAR A DIFERENÇA ENTRE AS DISTÂNCIAS MEDIDAS NA BASE E NO TOPO DO EDIFÍCIO, QUE NA VERDADE SÃO IGUAIS. ....</b>	<b>42</b>
<b>FIGURA II.14 – PROJEÇÃO ORTOGONAL DA MESMA IMAGEM MOSTRADA NA FIGURA II.12. ....</b>	<b>42</b>

<b>FIGURA II.15 – EXEMPLO DE DOIS NÍVEIS DIFERENTES(HIDROGRAFIA E ALTIMETRIA) QUE JUNTO COM OUTROS COMPÕEM A REGIÃO RESTITUÍDA, REPRESENTADA ABAIXO DELES. ....</b>	<b>44</b>
<b>FIGURA III.1 – PONTOS CHAVE PARA A IMPLEMENTAÇÃO DO PROJETO E-FOTO.....</b>	<b>55</b>
<b>FIGURA III.3 – PILARES PARA IMPLEMENTAÇÃO DO E_FOTO .....</b>	<b>58</b>
<b>FIGURA III.4 – OS TRÊS NÍVEIS DE APRENDIZADO OBTIDAS COM O E-FOTO.....</b>	<b>60</b>
<b>FIGURA III.5 – FLUXO DE ATIVIDADES PARA MAPEAMENTO DE TERRENO .....</b>	<b>61</b>
<b>FIGURA III.6 – DIAGRAMA DE MÓDULOS DO E-FOTO .....</b>	<b>62</b>
<b>FIGURA III.7 – DIAGRAMA DE CASOS DE USO DO E-FOTO.....</b>	<b>63</b>
<b>FIGURA III.8 – DIAGRAMA DE CLASSES DO PROJETO E-FOTO. ....</b>	<b>64</b>
<b>FIGURA III.9 – INTERFACES DE CONFIGURAÇÃO DO PROJETO E-FOTO .....</b>	<b>67</b>
<b>FIGURA III.10 – INTERFACE DE CALIBRAÇÃO DE CÂMARA DO E-FOTO .....</b>	<b>68</b>
<b>FIGURA III.11 – INTERFACE DO MÓDULO DE ORIENTAÇÃO INTERIOR DO PROJETO E-FOTO.....</b>	<b>69</b>
<b>FIGURA III.12 – INTERFACE DO MÓDULO DE MEDIÇÃO DIGITAL .....</b>	<b>69</b>
<b>FIGURA III.13 INTERFACE DO MÓDULO DE MEDIÇÃO E VISUALIZAÇÃO ESTEROSCÓPICA.....</b>	<b>70</b>
<b>FIGURA III.14 – INTERFACE DE RETIFICAÇÃO DE IMAGENS DIGITAIS DO E-FOTO.....</b>	<b>71</b>
<b>FIGURA IV.1 – SISTEMA DE VERIFICAÇÃO DE DADOS DO ALUNO - EXEMPLO DE SISTEMA DE INFORMAÇÃO GERENCIAL .....</b>	<b>74</b>

<b>FIGURA IV.2 – NOVO PROCESSO DE CONTAGEM.....</b>	<b>81</b>
<b>TOTAL .....</b>	<b>85</b>
<b>TOTAL .....</b>	<b>88</b>

## ÍNDICE DE TABELAS

TABELA II.1 - COMPLEXIDADE FUNCIONAL DE ALI E AIE .....	20
TABELA II.2 – CONTRIBUIÇÃO DE P.F. NÃO AJUSTADOS EM FUNÇÕES DO TIPO DADO.....	21
TABELA II.3 – COMPLEXIDADE FUNCIONAL PARA ENTRADAS EXTERNAS .....	23
TABELA II.4 – COMPLEXIDADE FUNCIONAL PARA SAÍDAS E CONSULTAS EXTERNAS.....	24
TABELA II.5 – CONTRIBUIÇÃO DOS P.F. NÃO AJUSTADOS EM FUNÇÕES DO TIPO TRANSAÇÃO .....	24
TABELA II.6 – CARACTERÍSTICAS GERAIS DE SISTEMA PARA CÁLCULO DO FATOR DE AJUSTE .....	25
TABELA II.7 – NÍVEIS DE INFLUÊNCIA DAS CARACTERÍSTICAS GERAIS DO SISTEMA .....	25
TABELA III.1 – MÓDULOS IMPLEMENTADOS DO PROJETO E-FOTO .....	66
TABELA IV.1 – ESFORÇO DE IMPLEMENTAÇÃO DE COMPONENTES RELACIONADOS A MATRIZES .....	76
TABELA IV.2 – ESFORÇO DE IMPLEMENTAÇÃO DE COMPONENTES DE TRANSFORMAÇÕES.....	76
TABELA IV.1 – COMPLEXIDADE DE FUNÇÕES DO TIPO MATRIZ .....	79
TABELA IV.2 – COMPLEXIDADE DE FUNÇÕES DO TIPO TRANSFORMAÇÕES.....	79
TABELA IV.3 – CONTRIBUIÇÃO DE FUNÇÕES DO TIPO MATRIZ .....	79

<b>TABELA IV.4 – CONTRIBUIÇÃO DE FUNÇÕES DO TIPO TRANSFORMAÇÕES.....</b>	<b>80</b>
<b>TABELA V.1 – CONTAGEM DE FUNÇÕES TIPO DADOS –RETIFICAÇÃO - IFPUG.....</b>	<b>84</b>
<b>TABELA V.2 - CONTAGEM DE FUNÇÕES TIPO TRANSAÇÃO – RETIFICAÇÃO - IFPUG.....</b>	<b>84</b>
<b>TABELA V.3 – CONTAGEM DE FUNÇÕES TIPO DADOS – RETIFICAÇÃO - PROPOSTA .....</b>	<b>85</b>
<b>TABELA V.4 - CONTAGEM DE FUNÇÕES TIPO TRANSAÇÃO – RETIFICAÇÃO.....</b>	<b>85</b>
<b>TABELA V.5 - CONTAGEM DE FUNÇÕES TIPO CÁLCULO MATEMÁTICO – RETIFICAÇÃO.....</b>	<b>85</b>
<b>TABELA V.6 – CONTAGEM DE FUNÇÕES TIPO DADOS – O.I. - IFPUG.....</b>	<b>86</b>
<b>TABELA V.7 - CONTAGEM DE FUNÇÕES TIPO TRANSAÇÃO – O.I. - IFPUG .....</b>	<b>86</b>
<b>TABELA V.8 – CONTAGEM DE FUNÇÕES TIPO DADOS - O. I. – PROPOSTA DISSERTAÇÃO .....</b>	<b>87</b>
<b>TABELA V.9 – CONTAGEM DE FUNÇÕES TIPO TRANSAÇÃO - O. I. – PROPOSTA DISSERTAÇÃO .....</b>	<b>87</b>
<b>TABELA V.10 - CONTAGEM DE FUNÇÕES TIPO CÁLCULO MATEMÁTICO – O. I. – PROPOSTA DISSERTAÇÃO.....</b>	<b>88</b>
<b>TABELA V.11 – VALORES DE CALIBRAÇÃO DA FERRAMENTA USC COCOMO II.1999.0 .....</b>	<b>89</b>
<b>TABELA V.12 – HORAS REAIS UTILIZADAS PARA IMPLEMENTAÇÃO DE MÓDULOS DO E-FOTO .....</b>	<b>90</b>



<b>TABELA V.13 – RESULTADOS DE ESFORÇO E TEMPO REAL DE IMPLEMENTAÇÃO.....</b>	<b>91</b>
---	-----------

## LISTA DE SIGLAS

AIE – Arquivo de Interface Externa

ALI – Arquivo Lógico Interno

APF – Análise de Pontos de Função

AR – Arquivo Referenciado

BFPUG – *Brazilian Function Point Users Group*

CAD – *Computer Aided Design*

CE – Consulta Externa

CGS – Características Gerais do Sistema

COCOMO – *Constructive Cost Model*

COSMIC FFP – *Common Software Measurement International Consortium – Full Function Points*

CPM – *Counting Practices Manual*

CPU – *Central Processing Unit*

EE – Entradas Externas

E-FOTO – Estação Fotogramétrica Digital Educacional Livre

FDL – *Free Documentation License*

FFP – *Full Functions Point*

GNU – *GNU's not Unix*

GPL – *General Public License*

GPS – *Global Positioning System*

IFPUG – *International Function Point Users Group*

ISP – *International Society for Photogrammetry*

ISPRS – *International Society for Photogrammetry and Remote Sensing*

LOC – *Lines Of Code*

MKII – *Mark II*

MDS – Modelo Digital de Superfícies

MDE – Modelo Digital de Elevações

OMG – *Object Management Group*

OMT – *Object Modeling Technique*

OOD – *Object Oriented Design*

OOSE – *Object Oriented Software Engineering*

PF – Pontos de Função

SAD-69 – *South American Datum 1969*

SE – Saída Externa

SIRGAS – Sistema de Referência Geocêntrico para a América do Sul

TD – Tipo de Dado

TDI – *Total Degree of Influence*

TI – Tecnologia da Informação

UKSMA - *United Kingdom Software Metrics Association*

UML – *Unified Modeling Language*

USC – *University of Southern California*

UTM – *Universal Transverse Mercator*

VAF – *Value Adjustment Factor*

XML – *Extensible Markup Language*

## CAPÍTULO I - INTRODUÇÃO

### I.1 APRESENTAÇÃO

Em um contexto mundial onde cada vez mais as organizações, sejam elas públicas ou privadas, preocupam-se com o controle de seus gastos e focam na redução contínua de custos operacionais, a área da tecnologia da informação (TI) não pode deixar de fornecer sua contribuição. Solicita-se sempre a todas as áreas das organizações, que façam mais com menos.

No que tange especificamente à tecnologia da informação, uma das atividades onde se busca cada vez mais economizar é na área de desenvolvimento de *software*. Projetos de desenvolvimento de sistemas de informação têm sido demasiado onerosos para as organizações, não trazendo, muitas vezes, o retorno esperado, ou necessário, para pagar seus custos de implementação.

De acordo com (LEFFINGWELL *et al.*, 2000), um estudo do *Standish Group* realizado em 1994 sobre o mercado americano de TI, cita que são gastos a cada ano 250 bilhões de dólares para a implementação de 175.000 projetos de desenvolvimento de *software*. Segundo esse estudo, 31% dos projetos futuros de *software* serão cancelados antes mesmo de seu término. No que diz respeito ao custo, o estudo informa que 52,7% dos projetos irão custar 189% de sua estimativa original e as organizações pagarão US\$ 59 bilhões por projetos que serão completados mas irão exceder sua estimativa inicial de custo.

Os dados fornecidos pelo estudo do *Standish group* deixam claro que há espaço para redução de custos no que tange ao desenvolvimento de *software*. Essa melhora pode ser obtida através de uma gestão adequada dos recursos necessários para a implementação de projetos dessa natureza.

Segundo (VAZQUEZ *et al.*, 2001), as atividades relativas à gestão de projetos podem ser agrupadas em três grandes passos: Planejamento, Execução e Controle. Durante a etapa de planejamento, faz-se necessário estimar o esforço a ser empreendido para a implementação do projeto, de forma a dimensionar quantidade de recursos e orçamento para o mesmo. A etapa de execução, por sua vez, deve ser constantemente monitorada e medida, para que seja possível controlar o andamento das atividades e verificar se o projeto está de acordo com o orçamento e prazo planejados.

Para que as atividades de gestão de um projeto sejam realizadas de forma satisfatória, faz-se necessária a utilização de algum mecanismo que permita mensurar seu tamanho. Visando atender a essa necessidade foram criados diversos modelos de estimativa para o fornecimento de métricas que permitam dimensionar, com a menor margem de erro possível, a real dimensão de um projeto de *software*. Mas diante dos diversos modelos para estimativa de *software* fica a questão: Qual modelo deve-se utilizar?

A resposta a essa pergunta pode estar relacionada com a natureza da aplicação a ser implementada. Cada modelo de métrica possui características próprias e podem ser aplicados, com melhor ou pior precisão, em determinado tipo de sistema de informação. Sistemas que possuam como foco o processamento matemático complexo, como o *software* desenvolvido para as áreas da cartografia e fotogrametria digital, podem ter seu tamanho medido com menor precisão por um modelo de métrica considerado excelente para sistemas comerciais.

Visando encontrar uma métrica adequada para medir o esforço de implementação de *software* na área da fotogrametria digital, este trabalho realiza um estudo sobre a análise de pontos de função, uma das métricas mais utilizadas atualmente no Brasil, e sua aplicabilidade sobre sistemas de informação com foco em processamento matemático complexo. O objetivo do estudo e a estruturação do trabalho encontram-se descritos nas seções seguintes.

## I.2 OBJETIVO

Este trabalho tem como objetivo a formulação e apresentação de uma proposta de métrica para mensurar sistemas de informação aplicados à área da fotogrametria digital.

A proposta apresentada busca aprimorar a precisão das medidas de *software* obtidas pela análise de pontos de função quando aplicada sobre sistemas da área da fotogrametria digital. Sistemas dessa área possuem como característica principal o foco no processamento de cálculos matemáticos de elevada complexidade. Essa característica costuma fazer com que medições realizadas com a análise de pontos de função não tenham a precisão desejada.

Para alcançar o objetivo proposto, é realizado um estudo sobre os conceitos da análise de pontos de função, elaborada e aplicada uma proposta de especialização dessa métrica sobre módulos do projeto E-FOTO (Estação Fotogramétrica Digital). Finalmente, o resultado da aplicação da nova métrica é comparado com o resultado obtido pelo modelo original de pontos de função e com o tempo real de implementação dos módulos. A comparação é analisada, sendo o resultado desta análise exposto na seção de conclusão..

### I.3 ORGANIZAÇÃO

Esta dissertação está organizada em 6 (seis) capítulos. O presente capítulo apresenta uma introdução sobre o objeto do estudo, descrevendo o objetivo do trabalho e a sua organização.

O capítulo II (dois) apresenta os fundamentos teóricos utilizados no trabalho. Esse capítulo fornece todos os conceitos necessários para o correto entendimento da proposta dessa dissertação.

O capítulo III (três) fornece uma explicação sobre o projeto E-FOTO. São fornecidos os objetivos do projeto, os módulos já implementados e os módulos a serem desenvolvidos.

O quarto capítulo apresenta a proposta de métrica para dimensionamento de *software* aplicado à fotogrametria digital. Nesse capítulo, são apresentados ainda os elementos da proposta de métrica, assim como, uma breve análise sobre a aplicação da métrica de pontos de função em projetos de *software* com foco em processamento matemático complexo.

No quinto capítulo, a proposta de métrica formulada no decorrer deste trabalho é aplicada sobre dois módulos do projeto E-FOTO, de acordo com a métrica proposta, com o modelo padrão de análise de pontos de função utilizado atualmente e com modelo de custos COCOMO II. Os resultados obtidos são apresentados de forma que se possa compará-los facilmente. Ainda nesse capítulo, são feitas considerações sobre o perfil da equipe do projeto.

O sexto e último capítulo corresponde à conclusão do trabalho. No decorrer desse capítulo são analisados os resultados obtidos da aplicação da proposta de métrica e fornecidos os comentários finais da dissertação.

## CAPÍTULO II – FUNDAMENTOS TEÓRICOS

### II.1 INTRODUÇÃO

A seguir é apresentado um panorama geral dos fundamentos teóricos e conceituais necessários para o correto entendimento do objetivo desta dissertação.

A parte conceitual deste trabalho se inicia com a apresentação dos fundamentos teóricos de métricas e formas de medição de custo de *software*, tendo como foco em seguida, a métrica de Pontos de Função e sua importância nos projetos de desenvolvimento de *software*. Posteriormente, apresentam-se conceitos de fotogrametria e cálculo numérico, complementos fundamentais para o entendimento deste trabalho.

### II.2 MÉTRICAS E FORMAS DE MEDIÇÃO DE CUSTO PARA DESENVOLVIMENTO DE *SOFTWARE*

Atualmente, toda organização que mantenha projetos voltados à contratação, desenvolvimento e manutenção de sistemas se depara com o mesmo problema: Como implantar um projeto de *software* dentro de um horizonte de tempo, prazo e custo definidos?

A resposta para esta pergunta passa, invariavelmente, pela gerência adequada de todo o projeto, através da definição e execução de passos básicos como:

- Planejamento;
- Execução;
- Controle.

Durante a etapa de planejamento, por exemplo, será definido o orçamento para o projeto, ou seja, qual o volume de recursos necessários para sua implementação. Para o correto dimensionamento desses recursos, é fundamental que se tenha uma estimativa do tamanho do *software* a ser implementado.

Uma vez planejado e iniciado, o projeto deve, durante sua execução, ser constantemente controlado, para evitar, entre outras coisas, desperdício de recursos e possíveis atrasos de execução. Durante a etapa de controle, faz-se necessário rever as

estimativas iniciais do projeto durante seu andamento e, caso necessário, corrigir seu rumo.

Tanto o planejamento quanto o controle do projeto, somente serão factíveis se for possível determinar o volume de recursos utilizados e o que foi produzido ao longo do ciclo de vida do projeto. Isto pode ser conseguido através da utilização de uma métrica que permita quantificar o esforço realizado na implementação do projeto. Neste contexto, surge a necessidade da adoção das métricas de *software*.

As métricas surgem como uma forma de quantificar o tamanho do *software* a ser desenvolvido. De acordo com (RAMOS, 2004) durante o processo de medição de software, números e símbolos do mundo real são atribuídos a entidades através de regras claras e definidas visando caracterizá-las. Desta forma, as métricas podem ser definidas como “*uma definição matemática algorítmica ou função utilizada para obter uma avaliação quantitativa de um processo ou produto*”(RAMOS, 2004), mas sua utilização leva a uma segunda pergunta: Qual medida representa melhor o tamanho de um *software*? Essa pergunta é atualmente debatida, tanto no meio acadêmico quanto no mercado de *software*, sem que haja unanimidade. Como fato, existem atualmente vários modelos de métricas, cada um com pontos a favor e contra.

Estabelecidas como ferramenta fundamental para medição do tamanho do software produzido, as métricas por si só não conseguem mensurar o custo do desenvolvimento do software. Para atender a este ponto, são estabelecidas formas de medições de custo, como a utilização de modelos de custo e a relação Homem x Hora. Recebendo como entrada os valores provenientes do resultado da aplicação das métricas, as medições de custo geram como saída um valor que representa o esforço necessário para a implementação do *software*.

De acordo com este contexto, como base teórica para este trabalho, são apresentados a seguir alguns modelos de métrica, assim como, de formas de medição de *software* usualmente encontrados na literatura acadêmica.

## II.2.1 LINHAS DE CÓDIGO

Uma das mais antigas modalidades de mensuração de *software* é a técnica de medição por linhas de código (LOC — *Lines of Code*) que consiste basicamente na



contagem da quantidade de linhas de código que um programa possui. É uma técnica de fácil implementação, no entanto, possui algumas desvantagens.

Dentre as desvantagens da utilização da LOC, pode-se citar: o fato de que é inexato ter que medir a produtividade de um projeto de desenvolvimento com a saída de somente uma das fases (fase de codificação); a dependência no que diz respeito à experiência do desenvolvedor, pois o número de linhas de código pode variar de pessoa a pessoa; a diferença entre linguagens (o número de linhas de código de uma aplicação desenvolvida em Basic pode ser diferente da quantidade de linhas de código da mesma aplicação desenvolvida em Cobol); a ausência de padronização de contagem, já que não há uma definição sobre a contagem de certas características como comentários e declarações de dados; a dificuldade de medição nas fases iniciais do ciclo de vida do *software*. Outro ponto de extrema importância reside na ausência de significado para o usuário final do *software*. Para esse usuário, normalmente leigo em computação, o termo linhas de código não possui um significado tangível.

## II.2.2 MARK II

Também conhecida como MK II, esta técnica foi criada por Charles Symons para a empresa KPMG nos anos oitenta, sendo atualmente de domínio público e mantida pela *United Kingdom Software Metrics Association* (UKSMA). Esta técnica procura melhorar a escala de tamanho funcional, contando de forma mais precisa a complexidade de processamento interno de Sistemas de Informações Gerenciais. Tem como uma das principais diferenças em relação a análise de pontos de função (APF), que será definida na seção II.3 desta dissertação, o fato de que esta conta “Arquivos Lógicos” uma vez para cada parte de *software* sendo mensurada, enquanto que a MK II conta “tipos de entidade” sempre que elas são referenciadas em cada transação lógica.

## II.2.3 COSMIC FFP

A *Common Software Measurement International Consortium – Full Function Points* (COSMIC-FFP) foi amplamente divulgada para a comunidade científica em 2000, através do consórcio chamado COSMIC, liderado pelos pesquisadores Alain

Abran e Charles Symons. Possuindo como pilar a técnica de FFP (*Full Functions Point*), criada por um grupo de pesquisadores da Universidade de Québec em Montréal, Canadá, a COSMIC-FFP surgiu como alternativa à APF para contagem de aplicações em tempo real e multiplataforma.

Embora esta técnica venha contribuindo de forma significativa para a correta medição de sistemas de tempo real, segundo (VOGELEZANG, 2005) ainda há o que melhorar para evitar a ambigüidade de alguns conceitos da COSMIC-FFP. De qualquer forma, existem muitos estudos acerca dessa métrica e é possível que futuramente haja um crescimento da utilização da mesma.

#### II.2.4 COCOMO II

O modelo de métrica construtivo (da sigla em inglês COCOMO – *Constructive Cost Model*) pode ser definido como um método que procura mensurar prazo, esforço, tamanho de equipe e custo voltados para o desenvolvimento de *software* (LÓPEZ, 2005). Sua utilização somente faz-se possível após o dimensionamento do tamanho do *software* através das métricas de pontos de função ou linhas de código.

O modelo COCOMO II representa uma evolução do modelo original COCOMO. Esse último, atualmente é considerado como obsoleto devido à idade dos projetos utilizados como base para seu desenvolvimento e a incapacidade de utilizar componentes de software.

Atualmente mantido e divulgado pela *University of Southern California*, o COCOMO II estima o custo e tempo baseado em pessoas/mês e meses, para a conclusão de uma atividade ou conjunto de atividades. O modelo prevê ainda um adicional de 20% ao tempo computado, como margem de erro (análise de risco).

O modelo em questão divide-se em 3 (três) sub-modelos que abordam as diferentes fases em que o projeto ou atividade que está sendo realizada se encontra. A utilização de cada sub-modelo aumenta a precisão e da estimativa ao longo do processo de planejamento e execução do projeto. Os sub-modelos utilizados são os seguintes:

*Application Composition* utilizado para estimar esforços de prototipação de *software* (USC, 1999). Normalmente empregado quando é necessário construir um protótipo com parte(s) do software a ser desenvolvido para analisar

questões de risco como integrações entre sistemas, *performance* ou maturidade da tecnologia empregada.

*Early Design* é apropriado quando as exigências são conhecidas e as alternativas de arquitetura do software foram exploradas. Normalmente utilizado para realizar estimativas de custo durante a etapa de análise, quando ainda não se tem definida com exatidão a arquitetura do sistema a ser implementado.

*Post-Architecture* é o sub-modelo o mais detalhado e envolve as etapas de construção real do software e de manutenção. Normalmente utilizado quando se tem uma arquitetura de sistema definida.

Este trabalho considerou a utilização do sub-modelo *Post – Architecture*, uma vez que os módulos de software a serem estimados já se encontravam implementados. A seção a seguir explicita a utilização do COCOMO II para mensurar custo e esforço de aplicações segundo o *Post – Architecture*. O leitor que desejar aprofundar seus conhecimentos em *Application Composition* ou *Early Design* consultar o manual de referência do modelo em questão (USC,1999).

#### II.2.4.1 PROCESSO DE DETERMINAÇÃO DE MEDIÇÕES SEGUNDO SUBMODELO POST-ARCHITECTURE

O processo de determinação de medições através do COCOMO II segue alguns passos básicos, mas antes de explicitá-los faz-se necessária a definição dos seguintes conceitos:

- a) **Calibração do Modelo:** A calibração do modelo constitui-se do ajuste de uma variável específica que indica o percentual de reuso do código. A calibração tem como objetivo obter a distribuição da produtividade e atividade de desenvolvimento para determinado ambiente. O modelo nominal vem calibrado com dados de projetos que podem ser totalmente distintos aos que se deseja estimar. Por esta razão é recomendável calibrar o modelo antes de sua utilização, com dados de projetos de natureza similar ao que se está estimando.

- b) Fatores de escala: O COCOMO II possui um fator exponencial para considerar despesas e economias obtidas em projetos de software de tamanhos diferentes. O sub-modelo *Post-Architecture* possui cinco fatores de escala: precedência, flexibilidade de desenvolvimento, resolução do risco/arquitetura, coesão da equipe e maturidade do processo. A cada fator deve ser aplicado um valor de cinco (baixo) a zero (alto), aplicá-los a uma equação específica e obter como resultado um valor que influenciará o modelo de custo indicando se há economia ou gasto no projeto.
- c) Multiplicadores de Esforço: São utilizados para capturar características do desenvolvimento e ajustar o resultado da medição de forma a refletir de maneira mais precisa o esforço real para implementação do projeto. Vêm pré-ajustados com valores nominais no modelo COCOMO II, podendo ter seus valores alterados para refletir as características específicas do projeto que se está estimando. Podem ter seus valores ajustados segundo classificação que vai de Extra baixo a Extra alto. Como exemplos de fatores de multiplicação de esforço, temos fatores de plataforma, fatores de produto e de pessoal.

Tomando-se como base os conceitos apresentados nesta seção, pode-se aplicar o COCOMO II de acordo com os seguintes passos:

Primeiro Passo: Efetuar a calibração do modelo com uma base histórica de projetos, preferencialmente similares ao projeto que se deseja estimar;

Segundo Passo: Aplicar a seguinte fórmula (Equação II.1), para cálculo do esforço do desenvolvimento, obtida em (LÓPEZ, 2005):

$$E = A * K * SLOC^b * (II EM_i)$$

II.1

Onde:

E - Esforço de desenvolvimento em pessoas-mês;

A – Fator que determina percentual de reutilização do código, sendo pré-ajustada pelo COCOMO II (USC, 1999) para o valor de 2,94 (dois vírgula noventa e quatro), também chamado de valor nominal;

KSLOC – Número de linhas de código do sistema a ser estimado;

EM – Multiplicadores de esforço;

b – expoente derivado da utilização dos cinco fatores de escala sobre a fórmula da equação II.2 a seguir:

$$b = 1,01 + 0,01 * \sum SF_i$$

II.2

Onde:

SF – fator de escala com valor no intervalo de cinco a zero.

Terceiro Passo: Aplicar a fórmula (Equação II.3) extraída de (LÓPEZ, 2005) para realizar o cálculo do tempo de desenvolvimento e estimar o prazo.

$$TDEV = C * (E^F)$$

II.3

Onde:

TDEV – Tempo de desenvolvimento expresso em meses;

C – Constante calibrada através de dados históricos. Segundo (USC, 1999) possui valor nominal igual a 3,67 (três vírgula sessenta e sete);

E – Esforço calculado anteriormente nesta seção;

F – Fator exponencial oriundo da aplicação da fórmula:

$$F = [D + 0,2 * (b - 1,01)]$$

II.4

Onde:

D – constante para tempo de desenvolvimento, calibrada através de dados históricos. Conforme (USC, 1999) possui valor nominal de 0,28 (zero vírgula vinte e oito);

b - expoente derivado da utilização de 5(cinco) fatores de escala previamente calculados nesta seção.

Quarto passo: Determinação da equipe média para o projeto, que pode ser obtida através da divisão do esforço pelo prazo. Segundo o (USC, 1999) o COCOMO II

considera um mês de trabalho equivalente a 152 (cento e cinquenta e duas) horas de trabalho excluído os fins de semana e feriados. Desta forma, tem-se:

$$P = E/TDEV$$

II.5

Onde:

P – quantidade média de profissionais alocados no projeto;

E – esforço previamente calculado nesta seção;

TDEV – Tempo de desenvolvimento, previamente calculado nesta seção.

Quinto Passo: Estimar o custo utilizando-se a equipe média P e o tempo desenvolvimento, TDEV, previamente calculados. Para a realização deste passo é necessário conhecer o valor hora dos integrantes do projeto.

A aplicação dos passos descritos nesta seção pode ser automatizada através do uso de uma ferramenta de *software* específica. A seção a seguir descreve de forma sucinta sua utilização.

#### II.2.4.2 FERRAMENTA DE *SOFTWARE* PARA REALIZAÇÃO DE ESTIMATIVAS BASEADAS NO COCOMO II

De forma a facilitar a utilização do COCOMO II como modelo de custo para realização de estimativas, a USC disponibiliza para *download* gratuito, uma ferramenta de *software* para aplicação do modelo. Denominada USC COCOMO II 1999.0, a ferramenta pode ser obtida em <http://sunset.usc.edu/research/cocomosuite/index.html>, estando disponível em versões *Windows* e *SUNOS*.

Uma vez instalada e acionada, a ferramenta possui uma interface que permite ao usuário configurar o modelo e executar a estimativa. Através da utilização de menus específicos é possível configurar os fatores de escala, multiplicadores de esforço e inserir parâmetros de calibração do modelo. Após a configuração e calibração do modelo, é possível inserir a quantidade de linhas de código ou o número de pontos de função do sistema a ser estimado e verificar os resultados de esforço, equipe envolvida e

tempo de desenvolvimento. A figura a seguir ilustra a interface da ferramenta de *software* em questão.

Untitled - USC-COCOMO II.1999.0

File Edit View Parameters Calibrate Phase Maintenance Help

Project Name:  Scale Factor:  Schedule:

Development Model:

X	Module Name	Module Size	LABOR Rate (\$/month)	EAF	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Exemplo 1	S:45300	0.00	1.00	194.8	194.8	232.6	0.00	0.0	9.9	0.0

Total Lines of Code:

Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
Optimistic	155.8	18.3	290.7	0.00	0.0	8.5	
Most Likely	194.8	19.6	232.6	0.00	0.0	9.9	0.0
Pessimistic	243.5	21.1	186.0	0.00	0.0	11.6	

Ready

**Figura II.1 – interface da ferramenta USC COCOMO II 1999.0**

A interface apresenta em sua parte superior um menu onde é possível calibrar o modelo e configurar os multiplicadores de esforço. No canto superior direito, pode-se fazer a opção pelo modelo a ser utilizado (*Early Design*, *Post- Architecture* etc). A ferramenta permite a inserção de vários módulos de um mesmo projeto, e para cada um deles é possível configurar parâmetros como: nome do módulo, tamanho do módulo (em linhas de código ou pontos de função não ajustados) e grau de risco. Uma vez inseridos os valores, na parte inferior central são apresentados os resultados divididos em três categorias: Otimista, mais provável e pessimista.

Neste trabalho buscou-se utilizar a interface acima descrita para realizar os cálculos de estimativas através do modelo COCOMO II. Os resultados e parâmetros utilizados na calibração do modelo encontram-se descritos nas seções de aplicação do modelo apresentação de resultados desta dissertação. Durante a seção de apresentação de resultados, a estimativa oriunda do modelo é apresentada em tempo de

desenvolvimento, que também está diretamente relacionada com a estimativa da métrica de pontos de função relacionadas a uma taxa de produtividade aferida em homem x hora. A seção a seguir apresenta o conceito da relação homem x hora para a estimativa de custo de desenvolvimento de projetos de *software*.

## II.2.5 MEDIÇÃO DE CUSTO POR RELAÇÃO HOMEM X HORA

A relação homem x hora é cada vez mais utilizada para explicitar o custo de um projeto de software. Este tipo de relação é muito comum no mercado de desenvolvimento de software, sendo conhecida, como uma modalidade de contratação (VAZQUEZ *et al.*, 2005). Nesta modalidade atribui-se um valor a cada hora utilizada de um profissional para executar determinada tarefa. Este valor normalmente encontra-se dentro de faixas praticadas pelo mercado de acordo com as qualificações do profissional envolvido na execução do serviço.

Normalmente, em negociações de projetos de software, a relação homem x hora é empregada em conjunto com uma métrica e uma taxa de produtividade. As taxas de produtividade são determinadas, em geral, através da consulta à bases históricas de projetos. Com uma base de dados históricos bem estruturada é possível determinar, por exemplo, que profissionais de um dado perfil são capazes de criar 10(dez) linhas de código em 1(uma) hora, utilizando uma determinada linguagem de programação. O mesmo raciocínio vale para outras métricas de *software*, como a métrica de pontos de função que será detalhada mais adiante neste trabalho.

Uma base de dados detalhada, pode contribuir de forma decisiva para a correta precificação de um projeto. Caso determinada instituição não possua uma base de dados detalhada o suficiente para aferir uma produtividade precisa, existem algumas organizações que publicam (ou vendem) bases históricas e informações de taxa de produtividade. Uma das mais conhecidas organizações internacionais provedoras de informações deste nível é o *International Software Group Benchmark* (ISBSG). Esta organização oferece dados históricos gratuitos para trabalhos acadêmicos, bem como, fornece publicações com taxas de produtividade com base nas principais métricas do mercado.

Este trabalho utilizou uma publicação do ISBSG para aferir uma taxa de produtividade média a métrica de software proposta (HILL, 2004). No decorrer desta



dissertação será possível visualizar qual taxa de produtividade foi considerada e sua aplicação sobre a métrica de pontos de função apresentada nas próximas seções.

## II.2.6 CONSIDERAÇÕES SOBRE AS MÉTRICAS DE *SOFTWARE*

As métricas apresentadas nas seções anteriores representam as técnicas de medida de *software* mais conhecidas atualmente, sendo utilizadas tanto no mercado mundial de *software* quanto no âmbito acadêmico. No entanto, embora possuam forte representação no cenário atual do mercado de desenvolvimento de *software*, nenhuma delas se popularizou tanto nos últimos anos quanto a Análise de Pontos de Função.

A análise de pontos de função vem sendo largamente utilizada, tanto no exterior quanto no Brasil, em projetos de desenvolvimento de *software* em empresas e instituições governamentais. Este fato por si só, além de representar um fator motivador para a elaboração deste trabalho de dissertação, torna obrigatória a exposição (e compreensão) dessa métrica de forma detalhada, conforme descrito a seguir.

## II.3 ANÁLISE DE PONTOS DE FUNÇÃO

Segundo (BFPUG, 2007), pontos de função podem ser definidos como uma medida funcional de tamanho de *software*. Assim como quilômetros representam unidade de medida para distância e horas representam unidade de medida para o tempo, os pontos de função são utilizados para determinar o tamanho do *software* em termos de funcionalidades.

Nascida na IBM no início da década de 70 (setenta), a análise de pontos de função se consolidou como um padrão de métrica de *software* a ser seguido pela indústria. Atualmente, a métrica vem sendo largamente empregada em contratos entre grandes empresas e até mesmo em licitações públicas para auxiliar na determinação do preço dos serviços de desenvolvimento de *software*.

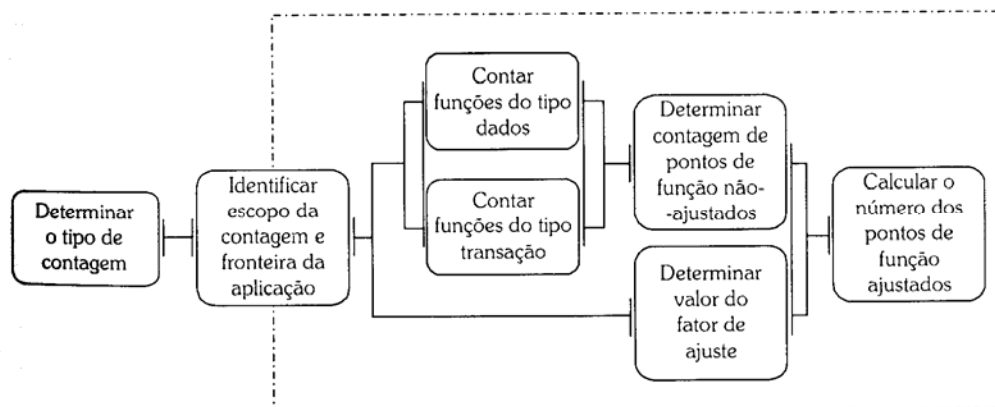
Após o surgimento da métrica de pontos de função, o crescimento acelerado de usuários acarretou na criação do *International Function Points Users Group* (IFPUG), em 1986. Essa organização tem como objetivo promover um melhor gerenciamento dos processos de desenvolvimento de *software*, através da utilização da métrica de pontos

de função. Além de difundir a utilização da métrica, o IFPUG se responsabiliza também por padronizar as regras de contagem, através da criação e publicação do manual de práticas de contagem, também conhecido como *Counting Practices Manual* (CPM). Atualmente, esse manual é reconhecido pela indústria de *software* como padrão para contagem de pontos de função.

O IFPUG busca promover o uso da métrica de pontos de função através da realização constante de seminários, *Workshops* Educacionais e conferências. A organização promove também a criação de associações locais, denominadas *chapters* (capítulos), para troca de experiências e estabelecimento de redes de contatos entre indivíduos que tenham como objetivo comum o estudo e utilização da análise de pontos de função. No Brasil, o *Brazilian Function Point Users Groups* (BFPUG) é o *chapter* do IFPUG. Constituído em 1998 e localizado na cidade do Rio de Janeiro, RJ, esse *chapter* promove a troca de experiências e contato entre profissionais, a realização de palestras e o exame de certificação do IFPUG no Brasil.

### II.3.1 CONTAGEM DOS PONTOS DE FUNÇÃO

O processo de contagem dos pontos de função pode ser resumidamente descrito de acordo com a gravura abaixo, adaptada de (VAZQUEZ *et al.*, 2005):



**Figura II.2 – Processo de Contagem dos Pontos de Função**

Conforme ilustrado na figura II.2, o processo de contagem inicia-se com a determinação do tipo de contagem. Após esta etapa, identifica-se o escopo da contagem e até onde vai a fronteira da aplicação a ser contada. A etapa seguinte constitui-se da contagem de funções de dois tipos: dados e transação. Encerrada essa etapa, deve-se determinar

quantos pontos de função não ajustados foram obtidos através do somatório dos pontos de função obtidos na contagem de cada tipo de função. Em paralelo a esta determinação, calcula-se um fator de ajuste e, finalmente, utilizando o fator de ajuste obtido calcula-se o número de pontos de função ajustados.

O detalhamento de cada etapa do processo de contagem pode ser encontrado nas seções a seguir.

#### II.3.1.1 DETERMINAÇÃO DO TIPO DE CONTAGEM

A etapa de determinação do tipo de contagem corresponde à etapa inicial do processo de contagem dos pontos de função. Durante essa etapa é estabelecido o tipo de contagem que será utilizado para medir o *software*.

A contagem é dividida em três tipos, a saber:

- Contagem de uma aplicação;
- Contagem de um projeto de melhoria;
- Contagem de um projeto de desenvolvimento.

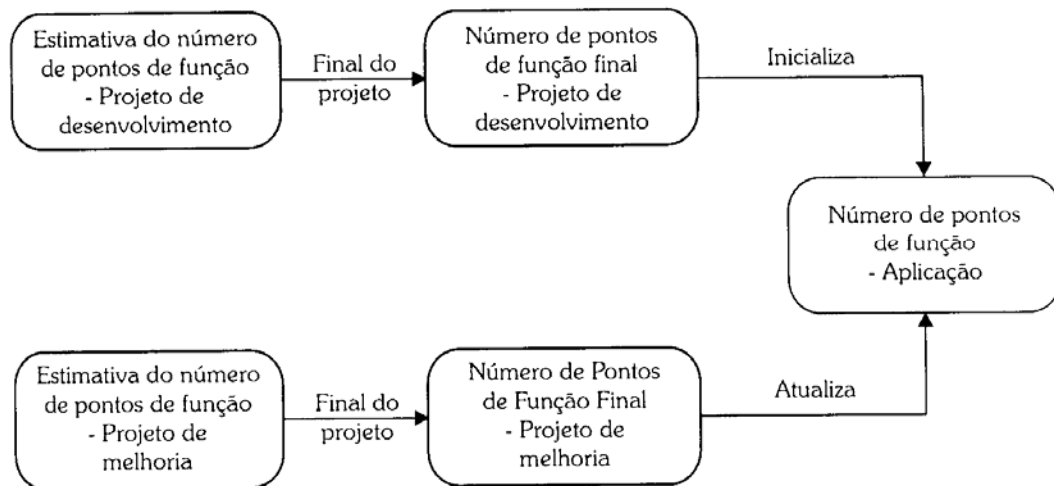
A contagem de um projeto de desenvolvimento irá medir a funcionalidade entregue aos usuários finais no momento da sua instalação inicial. Esse tipo de contagem contempla também a substituição de eventuais sistemas, assim como, as funcionalidades de conversão de dados necessárias para a migração de um sistema para o outro. Se, por exemplo, o sistema X será substituído pelo sistema Y, a contagem de desenvolvimento realizada sobre esse último irá contemplar todas as funções de conversão de dados necessárias para a importação dos dados do sistema X para o sistema Y.

No que diz respeito à contagem de um projeto de melhoria, essa será utilizada para medir funcionalidades adicionadas, excluídas ou alteradas em um sistema já implantado. Essa contagem, assim como a contagem de um projeto de desenvolvimento, contempla a medição de funções de conversão de dados necessárias para sua implantação.

Iniciada ao final da contagem dos pontos de função de um projeto de desenvolvimento, a contagem de uma aplicação (também conhecida como *baseline*) fornece a medida atual das funcionalidades obtidas pelo usuário da aplicação. Essa

contagem deve ser atualizada ao término de todo projeto de melhoria que venha a alterar funcionalidades da aplicação.

O relacionamento entre os tipos de contagem pode ser melhor compreendido com a ajuda da gravura abaixo, adaptada de (VAZQUEZ *et al.*, 2005).



**Figura II.3 – Relacionamento entre os tipos de contagem**

A figura II.3 ilustra o relacionamento entre os tipos de contagem. A estimativa do número de pontos de função para projetos de desenvolvimento é realizada no início do projeto. A contagem de aplicação é inicializada no final da contagem de desenvolvimento e atualizada no término de todo projeto de melhoria que altera funcionalidade da aplicação. Uma aplicação pode ser visualizada como um conjunto de funcionalidades que fazem parte de um escopo. A seção seguinte detalha o que se considera identificação de escopo e fronteira de aplicação.

#### II.3.1.2 IDENTIFICAÇÃO DO ESCOPO DA CONTAGEM E FRONTEIRA DA APLICAÇÃO

Uma vez identificado o tipo de contagem a ser utilizado, faz-se necessário identificar o escopo da contagem, assim como, definir a fronteira da aplicação. Essa etapa é fundamental no processo de contagem dos pontos de função. Caso a definição de escopo ou da fronteira seja feita de forma incorreta, a contagem final poderá ser seriamente comprometida.

O escopo da contagem deverá informar qual(is) sistema(s) ou parte de qual(is) sistema(s) será(ão) contemplado(s) na contagem. O escopo de uma contagem poderá conter ainda todas as funcionalidades disponíveis, podendo inclusive ser restrito apenas a funcionalidades específicas como emissão de relatórios e cadastro de dados.

A fronteira da aplicação pode ser entendida como uma “*fronteira conceitual*” entre o *software* a ser medido e o mundo exterior. Ela separa o *software* dos elementos externos que interagem (usuários, outras aplicações ou sistemas) com ela.

A identificação da fronteira da aplicação é fundamental para a medição funcional do *software*. Uma má identificação da fronteira pode acarretar em consequências indesejáveis para a contagem, como:

- Contagem duplicada de transações. A mesma transação é contada em duas ou mais aplicações;
- Duplicidade na contagem de arquivos;
- Dificuldade na determinação do tipo de transação.

Visando orientar a correta identificação da fronteira da aplicação, o IFPUG especificou um conjunto de regras (VAZQUEZ *et al.*, 2005):

- 1 – Sua determinação deve ser feita com base no ponto de vista do usuário. O foco deve estar sempre no que ele pode entender e descrever;
- 2 – A fronteira entre aplicações deve ser baseada na separação das funções conforme estabelecido pelos processos de negócio, não em considerações tecnológicas;
- 3 – Em projetos de melhoria, a fronteira estabelecida no início do projeto deve estar de acordo com a fronteira já estabelecida para a aplicação que está sendo modificada.

A adoção desse conjunto de regras facilita em muito a identificação da fronteira, além de servir como um padrão para evitar interpretações errôneas sobre o escopo de contagem. Somente após a correta identificação do escopo da contagem e da fronteira da aplicação é possível passar ao próximo passo do processo de contagem, a contagem de funções do tipo dados.

### II.3.1.3 CONTAGEM DE FUNÇÕES TIPO DADOS

Funções do tipo dados procuram representar as funcionalidades relativas ao processamento de dados internos e externos à aplicação. Essas funções são classificadas como: Arquivos Lógicos Internos (ALI) e Arquivos de Interface Externa (AIE). Cabe esclarecer que nessa classificação, o termo arquivo não implica necessariamente em um arquivo do sistema operacional, mas sim em um ou mais grupos de dados que possuam alguma relação lógica e possam ser reconhecidos pelo usuário.

#### II.3.1.3.1 ALI's X AIE's

Os Arquivos Lógicos Internos têm como principal função o armazenamento de dados mantidos por um ou mais processos elementares da aplicação que está sendo contada. Fundamentalmente, os ALI se constituem de um conjunto de dados identificável pelo usuário que estão logicamente relacionados e mantidos dentro da fronteira da aplicação. Como exemplos de ALI podemos citar: tabelas do banco de dados que armazenam dados mantidos pela aplicação, arquivos de configuração da aplicação e arquivos de *help*, mensagens de erro e segurança de acesso.

Os Arquivos de Interface Externa (AIE) têm uma função complementar à dos ALI. Eles são responsáveis pelo armazenamento de dados referenciados em processos elementares dentro da fronteira da aplicação. Todo AIE é obrigatoriamente um ALI de outra aplicação. A principal diferença entre os dois é que o AIE não é mantido pela aplicação que está sendo contada, estando fora de sua fronteira.

Como exemplos de AIE, podem-se destacar: dados de referência mantidos por outra aplicação e consultados pela aplicação que está sendo submetida à contagem, arquivos de ajuda mantidos por aplicação externa, arquivos de mensagens de erro gerados e mantidos por uma aplicação externa.

#### II.3.1.3.2 A DETERMINAÇÃO DA COMPLEXIDADE

De acordo com o manual de práticas de contagem do IFPUG, o próximo passo da contagem é a classificação de cada AIE e cada ALI em três níveis de complexidade:

Alta, Média e Baixa. Essa classificação é feita com base na contagem dos tipos de dados e registros existentes em cada ALI ou AIE.

Uma vez aferida a quantidade de tipos de registros e de tipos de dados existentes em cada ALI/AIE, utiliza-se a tabela de complexidade fornecida pelo IFPUG para a determinação da complexidade.

**Tabela II.1 - Complexidade Funcional de ALI e AIE**

Tipos de Dados				
Tipos de Registros		<20	20-50	>50
	1	Baixa	Baixa	Média
	2-5	Baixa	Média	Alta
	>5	Média	Alta	Alta

Desta forma, segundo a tabela 2.1, um AIE contendo 32 tipos de Dados e 3 tipos de registros é classificado como de complexidade média.

#### II.3.1.3.3 TIPOS DE DADO E REGISTRO — DEFINIÇÕES E REGRAS DE CONTAGEM

Um tipo de dado pode ser compreendido como um campo único, que não seja repetido e possa ser reconhecido pelo usuário. Como padrão para sua contagem, o IFPUG estabelece as seguintes regras:

- 1- Deve ser contado um tipo de dado para cada campo único reconhecido pelo usuário e não repetido, mantido ou recuperado de um AIE ou de um ALI;
- 2- Caso duas ou mais aplicações mantenham ou referenciem o mesmo ALI/AIE, apenas os campos utilizados pela aplicação em análise devem ser contados;
- 3- Caso seja necessário estabelecer relacionamento com outros arquivos, deve ser contado apenas um tipo de dado para cada campo utilizado para formar o relacionamento.

Um tipo de registro pode ser definido como uma subclasse de tipo de dados, reconhecido pelo usuário e componente de um AIE ou ALI. Essas subclasses são divididas em opcionais e obrigatórias. As opcionais facultam ao usuário a entrar com

seu valor em um processo elementar que crie ou adicione dados no arquivo. As subclasses obrigatórias requerem sua utilização pelo processo elementar de adição ou criação de dados no arquivo.

Como regras para contagem do tipo do registro, têm-se:

- 1- Deve ser contado um tipo de registro para cada subclasse, obrigatória ou opcional, de um AIE ou ALI;
- 2- Caso não haja nenhuma subclasse, o próprio ALI ou AIE deverá ser contado como um tipo de registro.

#### II.3.1.3.4 DETERMINAÇÃO DA CONTRIBUIÇÃO

Uma vez determinada a complexidade dos arquivos envolvidos, deve-se obter a quantidade de pontos de função não ajustados que os mesmos representam. Essa quantidade é obtida através da tabela a seguir, fornecida pelo IFPUG como referência.

**Tabela II.2 – Contribuição de P.F. não ajustados em funções do tipo dado**

<b>Tipo de Função</b>	<b>Baixa</b>	<b>Média</b>	<b>Alta</b>
<b>ALI</b>	7 PF	10 PF	15 PF
<b>AIE</b>	5 PF	7 PF	10 PF

Como exemplo, tem-se um AIE de complexidade baixa que contribui com 5 (cinco) pontos de função não ajustados enquanto que um ALI de complexidade baixa contribui com 7 (sete) pontos de função não ajustados. A diferença entre a contribuição de ambos ocorre não somente por uma eventual diferença de complexidade mas também devido à classificação do tipo de função considerado.

#### II.3.1.4 CONTAR FUNÇÕES DO TIPO TRANSAÇÃO

Este tipo de função tem como objetivo representar a funcionalidade que é oferecida ao usuário. As funções de transação são classificadas em: Entradas Externas, Saídas Externas e Consultas Externas.



As Entradas Externas (EE) podem ser definidas como um processo elementar que processa dados ou informações de controle recebidos de fora da fronteira da aplicação cuja meta é manter um ou mais ALI's ou alterar o comportamento do sistema. Como exemplos de Entradas Externas, podem-se citar transações que recebam dados externos utilizados na manutenção da aplicação, interfaces que permitam adicionar, excluir e remover registros em arquivos.

As Saídas Externas (SE), por sua vez, podem ser entendidas como um processo elementar que enviam dados ou informações de controle para fora da fronteira da aplicação e possui como objetivo apresentar informação ao usuário por meio de lógica de processamento. Essa lógica de processamento deve conter pelo menos uma fórmula matemática. Pode também manter um ou mais ALI's e/ou alterar o comportamento do sistema. Como exemplos de Saídas Externas, têm-se relatórios com totalização dos dados, relatórios de atualização de arquivos e informações em formato gráfico.

Já as Consultas Externas (CE) são processos elementares que enviam dados ou informação de controle para fora da fronteira da aplicação e têm como principal objetivo apresentar informação através da recuperação de dados ou informação de controle de um ALI ou AIE. Esse tipo de função não altera o comportamento da aplicação e não possui nenhum cálculo matemático em sua lógica de processamento. Como exemplos de CE, podem-se destacar telas de ajuda, informação sobre a versão do aplicativo e as telas de *logon*.

#### II.3.1.4.1 A DETERMINAÇÃO DA COMPLEXIDADE

A determinação da complexidade de funções do tipo transação é dada através de uma consulta às tabelas fornecidas pelo IFPUG. A complexidade é dividida em três níveis: alta, média e baixa, sendo determinadas através da quantidade do número de arquivos referenciados e tipos de dados que estão relacionados com determinada função de transação.

Um arquivo referenciado (AR) pode ser entendido como um ALI que é lido ou mantido pela função do tipo de transação, ou um AIE que é lido por ela. A contagem do número de arquivos referenciados deve seguir as seguintes regras:

- 1 - Deve ser contado um arquivo referenciado para cada ALI mantido;

- 2 - Deve ser contado um arquivo referenciado para cada ALI que seja tanto mantido quanto lido;
- 3 - Deve ser contado um arquivo referenciado para cada ALI ou AIE lido durante o processamento.

No que diz respeito a um tipo de dado, pode-se entendê-lo como um campo único, reconhecido pelo usuário e não repetido. A contagem de tipos de dados deve seguir as seguintes regras:

- 1 Deve ser contado um tipo de dado para cada campo, não repetido e reconhecido pelo usuário, que entra ou sai pela fronteira da aplicação, sendo necessário à conclusão do processo;
- 2 Caso um campo entre e saia da fronteira da aplicação, este deve ser contado uma única vez;
- 3 Campos que durante o processo elementar são recuperados ou derivados pelo sistema e armazenados em um ALI, sem, contudo atravessar a fronteira da aplicação, não devem ser contados;
- 4 Deve ser contado um único tipo de dado para a capacidade de especificar uma ação a ser tomada;
- 5 Deve ser contado um único tipo de dado para a funcionalidade de envio para fora da fronteira da aplicação de uma mensagem de resposta do sistema indicando erro durante processamento;
- 6 Literais não devem ser contados como tipo de dado;
- 7 Variáveis de paginação e campos automáticos não devem ser contados como tipo de dado.

Uma vez determinadas as quantidades de arquivos referenciados e tipos de dado, utiliza-se as tabelas a seguir para obter a complexidade das funções.

**Tabela II.3 – Complexidade funcional para entradas externas**

Tipos de Dados				
Arquivos referenciados		<5	5-15	> 15
	< 2	Baixa	Baixa	Média
	2	Baixa	Média	Alta
	> 2	Média	Alta	Alta

**Tabela II.4 – Complexidade funcional para saídas e consultas externas**

Tipos de Dados				
Arquivos referenciados		<6	6-19	> 19
	< 2	Baixa	Baixa	Média
	2-3	Baixa	Média	Alta
	> 3	Média	Alta	Alta

Como exemplo, uma SE com 12 tipos de dados e 3 arquivos referenciados é considerada de média complexidade, enquanto que uma entrada externa que possui 2 arquivos referenciados e 16 tipo de dados possui alta complexidade.

#### II.3.1.4.2 DETERMINAÇÃO DA CONTRIBUIÇÃO EM PONTOS DE FUNÇÃO NÃO AJUSTADOS

Uma vez obtida a complexidade da função tipo dado, o próximo passo é determinar sua contribuição em pontos de função não ajustáveis. Este passo é realizado utilizando-se a tabela a seguir, fornecida pelo IFPUG.

**Tabela II.5 – Contribuição dos P.F. não ajustados em funções do tipo transação**

Tipo de Função	Baixa	Média	Alta
<b>EE</b>	3 PF	4 PF	6 PF
<b>SE</b>	4 PF	5 PF	7 PF
<b>CE</b>	3 PF	4 PF	6 PF

Dessa forma, uma Entrada Externa de complexidade baixa contribui com 3 (três) pontos de função não ajustados, enquanto que uma Saída Externa de complexidade alta contribui com 7(sete) pontos de função não ajustados.

#### II.3.1.5 DETERMINAÇÃO DO FATOR DE AJUSTE

Uma vez determinada a contribuição das funções de tipo de dados e transação em termos de pontos de função não ajustados, o próximo passo a ser considerado passa a ser o cálculo do fator de ajuste. Sendo conhecido como *Value Adjustment Factor*

(VAF), o fator de ajuste atualmente é considerado como um passo opcional pelo IFPUG no processo de cálculo de pontos de função.

O VAF de uma aplicação é calculado com base em 14 Características Gerais do Sistema (CGS). Essas características refletem funções que afetam, de forma geral, a aplicação que está sendo contada. A tabela apresentada a seguir fornece as características gerais a serem consideradas.

**Tabela II.6 – Características Gerais de Sistema para cálculo do Fator de Ajuste**

<b>Características Gerais do Sistema</b>	
1 – Comunicação de Dados	8 – Atualização <i>On line</i>
2 – Processamento Distribuído	9 – Processamento Complexo
3 – Performance	10 – Reutilização
4 – Configuração Altamente Utilizada	11 – Facilidade de Instalação
6 – Entrada de Dados <i>On line</i>	13 – Múltiplos Locais
7 – Eficiência do Usuário Final	14 – Modificações Facilitadas

Para o cálculo do fator de ajuste, a cada uma das características da tabela II.6, deve ser identificado o nível de influência que a característica geral exerce sobre o sistema. Esse nível de influência deve ser classificado pelo responsável pela contagem de pontos de função, tomando por base a tabela a seguir.

**Tabela II.7 – Níveis de influência das características gerais do sistema**

<b>Nível de Influência</b>	<b>Peso</b>
Nenhuma Influência	0
Influência Mínima	1
Influência Moderada	2
Influência Média	3
Influência Significativa	4
Grande Influência	5

Uma vez determinados os níveis de influência para cada característica geral, o fator de ajuste pode ser obtido através da utilização da seguinte fórmula (equação II.6), extraída de (VAZQUEZ *et al.*, 2005):

$$\text{VAF} = (\text{TDI} * 0,01) + 0,65$$

II.6

Onde:

TDI (*Total Degree of Influence*) – corresponde ao somatório dos níveis de influência das características gerais.

#### II.3.1.6 CÁLCULO DOS PONTOS DE FUNÇÃO AJUSTADOS

Último passo do processo de contagem dos pontos de função, o cálculo dos pontos de função ajustados considera o tipo de contagem, determinado no início do processo, para determinar a fórmula a ser utilizada. A seguir são apresentadas e explicadas as fórmulas de cálculo para cada um dos tipos de contagem.

##### II.3.1.6.1 CÁLCULO DE PONTOS DE FUNÇÃO EM PROJETOS DE DESENVOLVIMENTO

Como componentes para o cálculo dos PF de um projeto de desenvolvimento, tem-se: a funcionalidade da aplicação requisitada pelo usuário, a funcionalidade de conversão requisitada pelo usuário e o valor do fator de ajuste da aplicação. Considerando esses três componentes, tem-se a fórmula apresentada a seguir (equação II.7), extraída de (VAZQUEZ *et al.*, 2005).

$$\text{DFP} = (\text{UFP} + \text{CFP}) * \text{VAF}$$

II.7

Onde:

DFP - Número de pontos de função do projeto de desenvolvimento;

UFP - Número de pontos de função não ajustados das funções disponíveis após a instalação;

CFP - Número de pontos de função não ajustados das funções de conversão;

VAF - Valor do fator de ajuste.

#### II.3.1.6.2 CÁLCULO DOS PONTOS DE FUNÇÃO EM PROJETOS DE MELHORIA

De acordo com o IFPUG, o projeto de melhoria envolve apenas manutenções evolutivas na aplicação, ou seja, alterações realizadas para atender a novos requisitos de negócio. Não estão contempladas manutenções corretivas e preventivas.

Dessa forma, os componentes a serem considerados para o cálculo de PF para projetos de melhoria são: funcionalidade da aplicação requisitada pelo usuário para o projeto, a funcionalidade de conversão para converter dados para a correta instalação da aplicação, e o valor do fator de ajuste da aplicação. Considerando esses três componentes, tem-se a fórmula (equação II.8), extraída de (VAZQUEZ *et al.*, 2005), apresentada a seguir.

$$\text{EFP} = [(\text{ADD} + \text{CHGA} + \text{CFP}) * \text{VAFA}] + (\text{DEL} * \text{VAFB}) \quad \text{II.8}$$

Onde:

EFP – Número de pontos de função do projeto de melhoria;

ADD – Número de pontos de função não ajustados das funções incluídas pelo projeto de melhoria;

CHGA – Número de pontos de função não ajustados das funções modificadas. Reflete as funções após sua modificação;

CFP – Número de pontos de função não ajustados adicionados pela conversão;

VAFA – Valor do fator de ajuste da aplicação depois do projeto de melhoria;

DEL – Número de pontos de função não ajustados das funções excluídas pelo projeto de melhoria;

VAFB – Valor do fator de ajuste da aplicação antes do projeto de melhoria.

#### II.3.1.6.3 CÁLCULO DOS PONTOS DE FUNÇÃO DE APLICAÇÃO

O cálculo dos pontos de função de aplicação envolve a utilização de duas fórmulas. A primeira é utilizada apenas no caso de uma primeira contagem de pontos de função, enquanto a segunda deve ser utilizada para redimensionamento de tamanho

após um projeto de melhoria ter alterado sua funcionalidade. A seguir apresentamos as duas fórmulas a serem consideradas dependendo da finalidade da contagem.

Fórmula para contagem inicial, segundo (VAZQUEZ *et al.*, 2005):

Equação II.9

$$\text{AFP} = \text{ADD} * \text{VAF}$$

II.9

Onde:

AFP - Número de pontos de função ajustados da aplicação;

ADD - Pontos de função não ajustados das funções instaladas;

VAF - Valor do fator de ajuste.

Fórmula, segundo (VAZQUEZ *et al.*, 2005), após projeto de melhoria:

$$\text{AFP} = [(\text{UFPB} + \text{ADD} + \text{CHGA}) - (\text{CHGB} + \text{DEL})] * \text{VAFA}$$

II.10

Onde:

AFP – Número de pontos de função ajustados da aplicação;

UFPB – Pontos de função não ajustados da aplicação antes do projeto de melhoria;

ADD – Pontos de função não ajustados das funções incluídas pelo projeto de melhoria;

CHGA – Pontos de função não ajustados das funções alteradas pelo projeto de melhoria depois de seu término;

CHGB – Pontos de função não ajustados das funções alteradas pelo projeto de melhoria antes de seu término;

DEL – Pontos de função não ajustados das funções excluídas pelo projeto de melhoria;

VAFA – Valor do fator de ajuste depois do projeto de melhoria.

### II.3.2 CONSIDERAÇÕES SOBRE A ANÁLISE DE PONTOS DE FUNÇÃO

Durante a seção anterior foi apresentada a análise de pontos de função como método de medição funcional de *software*. Foram abordados todos os passos necessários para a correta aplicação do método, assim como, citados alguns exemplos para ilustrar a aplicação dos conceitos expostos.

Apesar da análise de pontos de função ter sido apresentada de forma detalhada, o correto entendimento da proposta desta dissertação somente poderá ocorrer após o estudo de conceitos básicos das áreas de fotogrametria e cálculo numérico. Com base neste fato, a seção seguinte apresentará a conceituação necessária no que diz respeito à fotogrametria. Posteriormente serão apresentados os fundamentos teóricos de cálculo numérico.

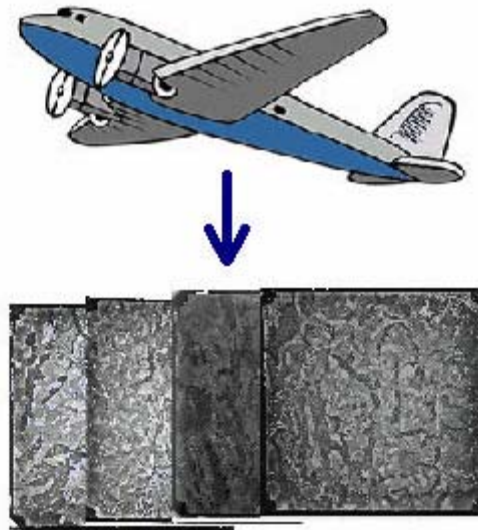
## II.4 FOTOGRAMETRIA

De acordo com (BRITO *et al.*, 2002), a palavra Fotogrametria tem sua origem proveniente de *photon*, que significa luz; *graphos*, escrita; e *metron*, medição. Desta forma, o termo Fotogrametria pode ser entendido como uma medição feita em fotografias.

Uma definição mais precisa para Fotogrametria, seria segundo (BRITO *et al.*, 2002): “ciência e tecnologia de se obter informação confiável, através de imagens adquiridas por sensores”.

A Fotogrametria pode ser entendida como uma ciência, uma vez que os processos de captação da energia eletromagnética, do qual os raios luminosos são formados, são estudados, bem como a análise dos resultados obtidos com os mesmos. No que diz respeito à tecnologia, pode-se dizer que a Fotogrametria utiliza o estado da arte da tecnologia existente para tornar os resultados mais rápidos e eficientes para o usuário final. A Informação obtida é extremamente confiável, de acordo com as normas e padrões vigentes e dentro de tolerâncias e precisões desejáveis. Quanto às imagens adquiridas, podemos dizer que ocorre a aquisição de imagens ópticas que podem ser classificadas como “a reprodução aparente de um objeto, formado por um sistema de lentes ou espelhos, a partir de ondas luminosas refletidas, refratadas ou difratadas” (BRITO *et al.*, 2002). Os Sensores Remotos utilizados para aquisição de imagens podem ser definidos de acordo com (NOVO, 1992), como: “qualquer equipamento capaz de transformar alguma forma de energia em um sinal passível de ser convertido em informação sobre o ambiente, sem contato físico entre este sensor e os alvos de interesse”. Como exemplo simples de um sensor fotogramétrico largamente utilizado, podemos citar a câmara fotográfica, que utiliza a sensibilização química de um filme para gerar imagens. A figura a seguir, obtida de (BRITO *et al.*, 2002), ilustra a aquisição de imagens através de sensores ópticos implantados em um avião.





**Figura II.4 – Aquisição de Imagens através de Sensores ópticos implantados em um avião – Aerofotogrametria**

#### II.4.1 UM BREVE HISTÓRICO ATÉ A FOTOGRAMETRIA DIGITAL

Após a descoberta da fotografia, o francês Arago, em 1840, foi um dos primeiros a propor a sua utilização em levantamentos topográficos. Em 1851, os primeiros princípios e técnicas fotogramétricas são criados por Aimé Laussedat. Esses começam a ser utilizados para a documentação de edifícios e prédios históricos. Durante esse período começam, de maneira ainda tímida, serem tiradas as primeiras fotos aéreas.

A invenção do avião e a criação do primeiro aparelho opto-mecânico para realização de “estereocomparações” permitiram um avanço significativo da fotogrametria. Este período marca o início da chamada fotogrametria analógica.

A fotogrametria analógica é marcada como um período onde, pela primeira vez na história, aparelhos optico-mecânicos substituem o homem na realização de cálculos matemáticos associados à fotogrametria. Pode-se citar como exemplo a criação dos retificadores analógicos, que logo em seguida passaram a ser substituídos pelos restituídos analógicos, pois estes últimos permitiam visão estereoscópica, através da

utilização de um par estereoscópico (um par de fotografias com áreas de superposição). A figura a seguir, extraída de (BRITO *et al.*, 2002) nos mostra um restituidor analógico implementado com base na tecnologia criada neste período.



**Figura II.5 – Restituidor analógico Fonte: Universidade do Estado do Rio de Janeiro**

Cabe ressaltar que, durante a fase da fotogrametria analógica são criadas e fortalecidas várias associações e entidades que buscam congregar todos os interessados na área. Como exemplo de associação criada na época, pode-se citar a ISP (*International Society for Photogrammetry*), atual ISPRS (*International Society for Photogrammetry and Remote Sensing*), fundada em 1910 por E. Dolezal, na Áustria.

O período da fotogrametria analógica tem seu encerramento marcado pelo início de uma transformação nos processos fotogramétricos da época. Esta transformação começa a surgir com a criação dos primeiros computadores, a partir dos anos 40 (quarenta). Inicia-se então um novo período denominado fotogrametria analítica.

A fotogrametria analítica visava a utilização dos computadores para a realização dos cálculos matemáticos, antes realizados pelos aparelhos opto-mecânicos. O primeiro estudo neste sentido, desenvolvido em 1953 por Helmut Schmidt, estabeleceu as bases da fotogrametria analítica. Neste estudo foram criados, por exemplo, o tratamento matricial, as soluções por mínimos quadrados (um tipo de método de ajustamento estatístico), a solução simultânea utilizando múltiplas imagens e uma análise completa de propagação de erros.

No ano de 1957, o conceito de restituidor analítico é desenvolvido pelo Finlandês Hiki Helava. Esse conceito se baseava na utilização de servomecanismos para medição das coordenadas de marcas fiduciais das imagens em conjunto com a utilização de computadores para realização de todos os demais cálculos necessários para a restituição da imagem. A partir da elaboração deste conceito, surgiram 1976 os primeiros restituidores analíticos.

A fotogrametria analítica permanece como estado da arte até a década de 80 (oitenta), quando surge uma nova transformação tecnológica. O surgimento das imagens digitais.

A utilização das imagens em formato digital iria permitir, pela primeira vez na história, o processamento e análise computacional de todo o processo fotogramétrico. Neste contexto, surge o conceito da Fotogrametria digital, onde, segundo (BRITO *et al.*, 2002), se tem como principal objetivo realizar a reconstrução automática espaço tridimensional (espaço objeto), a partir de imagens bidimensionais (espaço imagem).

A evolução da tecnologia atualmente permite que sejam realizadas, de forma automática e rápida, inúmeras operações e algumas etapas do processo fotogramétrico. Pode-se dizer que com a fotogrametria digital busca-se criar uma verdadeira “máquina de mapeamento automático” (BRITO *et al.*, 2002). Procura-se como ideal criar uma máquina que, sem a intervenção humana, possa reconhecer feições do terreno (prédios, pontes, dentre outros), assim como, para extrair as formas do relevo da região a ser mapeada.

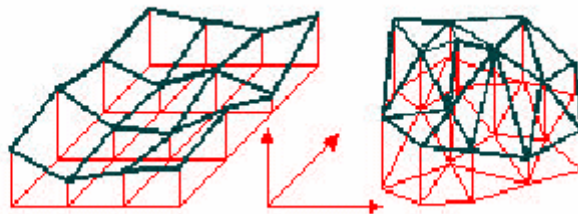
Apesar dos inúmeros avanços conquistados, deve-se esclarecer que o ideal da “máquina de mapeamento automático” ainda não foi alcançado. Como o relevo da superfície terrestre é extremamente descontínuo, invariavelmente, a interação do homem com a máquina faz-se necessária. Dessa forma, atualmente pode-se dizer que o estado da arte em fotogrametria digital está no mapeamento semi-automático do terreno, com a intervenção humana e através do uso de equipamentos computadorizados denominados estações fotogramétricas. Nas próximas seções deste documento, estarão sendo detalhados os conceitos básicos de fotogrametria, assim como, compreendida a definição e utilização de uma estação fotogramétrica digital.

## II.4.2 PORQUE UTILIZAR A FOTOGRAMETRIA

Embora a utilização da Fotogrametria traga uma série de benefícios para os cartógrafos (como redução do trabalho de campo para levantamento de coordenadas e obtenção de dados), segundo (BRITO *et al.*, 2002), o seu objetivo primordial é realizar “a reconstrução de um espaço tridimensional, chamado de espaço objeto, a partir de imagens bidimensionais, chamadas de espaço imagem”.

A reconstrução citada por (BRITO *et al.*, 2002), na realidade, representa uma transformação entre o sistema bidimensional de fotografias e o sistema em três dimensões (também conhecido como tridimensional) que representa o espaço objeto. Como base do sistema bidimensional utiliza-se normalmente o centro do quadro da câmara e coordenadas calibradas em laboratório. Já o sistema tridimensional usualmente tem como base coordenadas geodésicas do terreno sobre o qual se adquirem as imagens. Ocasionalmente, é possível criar um sistema de referência próprio, de origem arbitrária, dependendo do alvo em questão a ser mapeado.

Normalmente, o que se busca é realizar a reconstrução do espaço físico do tridimensional do terreno através de um computador. Isto pode ser feito através da conversão de dados dos sistemas, permitindo ao usuário realizar análises detalhadas no terreno, evitando custos e problemas oriundos da necessidade de deslocamento até o local. A figura a seguir, obtida de (BRITO *et al.*, 2002), ilustra um par de modelos tridimensionais de terreno reconstruídos digitalmente, o primeiro se baseia em elementos quadrangulares e o segundo em elementos triangulares.



**Figura II.6 – Processo de reconstrução tridimensional do terreno através do computador**

### II.4.3 AQUISIÇÃO DE IMAGENS

A aquisição de imagens fotogramétricas, na maior parte dos casos ocorre através da utilização da fotografia. Em geral, são realizadas fotos aéreas através da utilização de câmaras fotográficas acopladas a aviões que sobrevoam as áreas a serem mapeadas.

De acordo com (SILVEIRA, 2005), as câmaras fotográficas utilizadas, podem ser classificadas dentro dos seguintes tipos: analógicas, digitais, métricas e não métricas.

A câmara analógica grava a imagem obtida em uma película de filme por um período de tempo que varia de acordo com a quantidade de luz disponível no ambiente (quanto mais luz menor o tempo). O filme é então revelado, obtendo-se a imagem da cena impressa em um papel fotográfico.

Já a câmara digital adquire a imagem por meio de sensores fotoelétricos, que transformam a luz em informação elétrica armazenada em forma de dígitos binários. Estes dígitos seguem padrões de formatos de imagens largamente utilizados, como JPEG e BMP.

As câmaras métricas fornecem a possibilidade de aquisição das coordenadas dos objetos na foto, enquanto que as não métricas não permitem a captura de qualquer tipo de informação adicional, além da foto adquirida.

As fotografias aéreas são obtidas a partir de determinada altura e com base em escala a ser calculada.

O cálculo da escala é fornecido com base na altura e na distância focal. Sendo representada pela equação II.11 a seguir, obtida de (BRITO *et al.*, 2002):

$$E = f \times H$$

II.11

Onde:

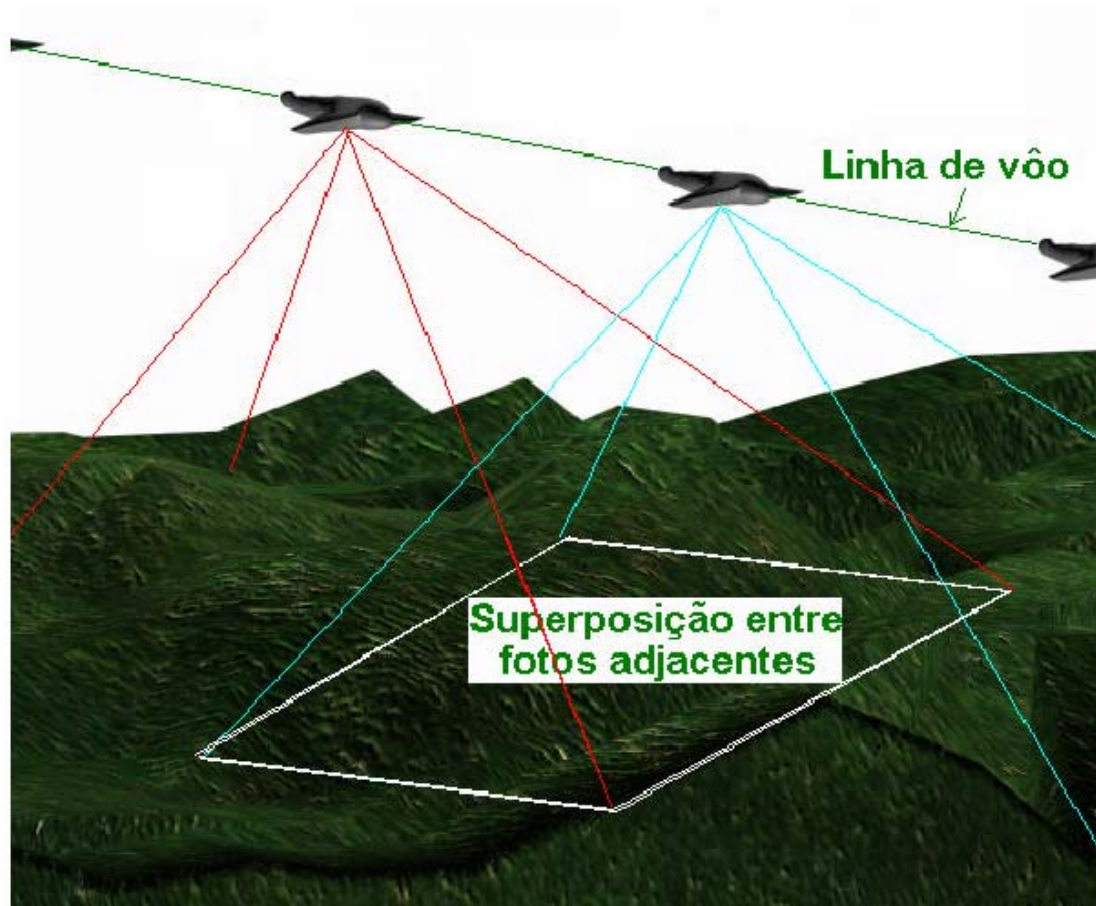
E - Escala.

f - Distância focal calibrada.

H - Altura de vôo.

Para a correta reconstrução do espaço tridimensional, é fundamental a utilização da equação descrita anteriormente, associada à sobreposição das imagens. O procedimento

de reconstrução do espaço tridimensional sugere que as imagens capturadas possuam uma superposição mínima de 60 (sessenta) % . A figura a seguir, extraída de (BRITO *et al.*, 2002), ilustra este requisito.



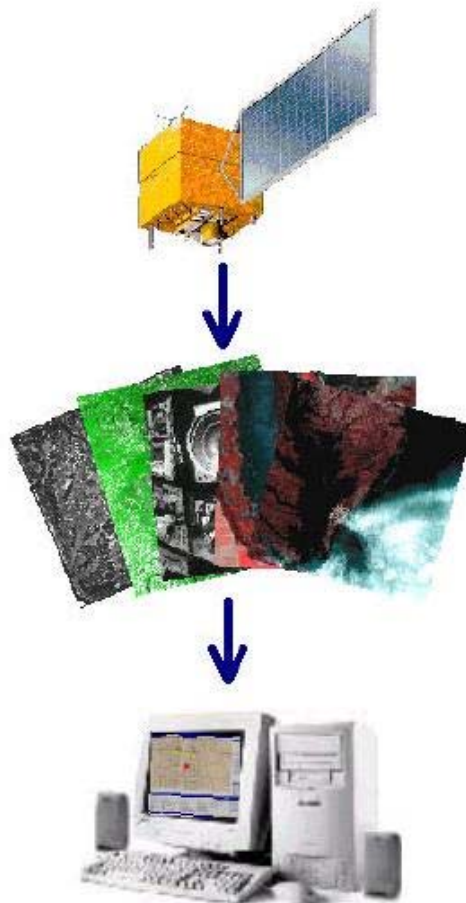
**Figura II.7 – Superposição de Fotos aéreas**

#### II.4.4 UTILIZAÇÃO DE SENSORES ORBITAIS

Atualmente, em paralelo à utilização das câmaras fotográficas, cada vez mais estão sendo utilizados para a aquisição de imagens, os chamados sensores orbitais. Como exemplos de sensores orbitais, podemos citar os satélites e os veículos espaciais que circulam na órbita do planeta capturando imagens de varias regiões e com várias escalas.

Funcionalmente, os sensores orbitais se enquadram na mesma posição das máquinas fotográficas no processo de geração tridimensional do terreno a ser mapeado. Eles capturam e enviam imagens de terreno para estações computadorizadas de

conversão de sistemas e tratamento destas imagens. Uma vez recebidas, as imagens passam pelo processo de transformação de sistemas e geração do terreno em três dimensões. A figura a seguir, extraída de (BRITO *et al.*, 2002), ilustra este processo.



**Figura II.8 – Aquisição de imagens por Sensores Orbitais**

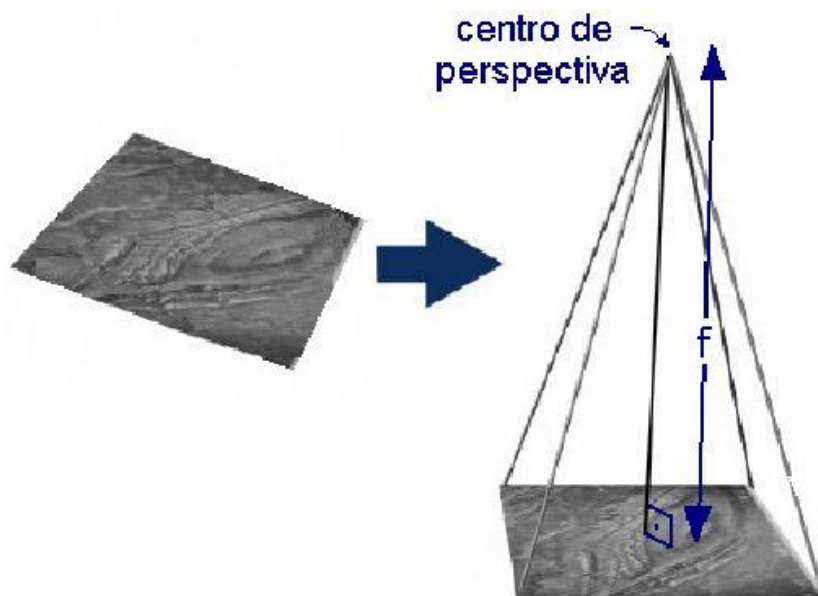
#### II.4.5 ORIENTAÇÃO INTERIOR

Uma vez adquiridas, as imagens devem passar uma etapa denominada processamento analítico. Durante este processo, a imagem será submetida a uma série de operações, para que ao final destas, seja reconstruído o espaço tridimensional (espaço objeto) desejado. A primeira operação a qual a imagem capturada deve sofrer é denominada de orientação interior.



Segundo (BRITO *et al.*, 2002), a orientação interior pode ser entendida como a reconstrução da posição dos feixes perspectivos em relação a câmara no momento da tomada da foto. Na prática, esta operação corresponde a referenciar a imagem em relação a câmara. Este procedimento é necessário, uma vez que as imagens inicialmente obtidas não possuem qualquer informação métrica, inviabilizando assim, a obtenção de qualquer medida precisa sobre elas.

Resumidamente, a orientação interior consiste na colocação das imagens adquiridas em posições semelhantes às que estavam dentro da câmara. A figura a seguir, extraída de (BRITO *et al.*, 2002), ilustra a reconstrução da geometria do feixe perspectivo.



**Figura II.9 – Orientação Interior – Reconstrução da geometria do feixe perspectivo**

A ferramenta utilizada para a realização da reconstrução da geometria do feixe perspectivo é denominada restituidor. Na fotogrametria digital, o restituidor responsável por referenciar a imagem em relação à câmara é um *software* fotogramétrico, que realiza a restituição de maneira automática ou com intervenção humana.

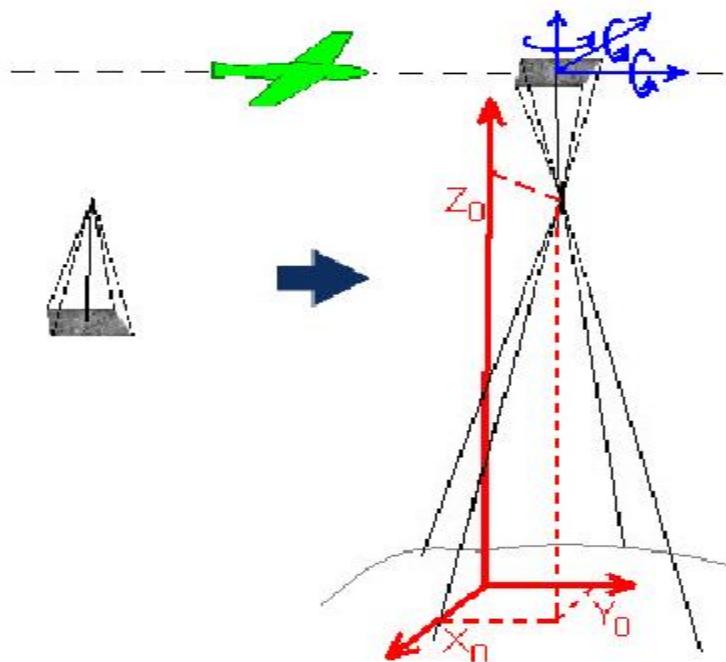


#### II.4.6 ORIENTAÇÃO EXTERIOR

Uma vez estabelecida a referência entre a imagem e a câmara utilizada, a próxima etapa a ser considerada no processo de reconstrução do espaço tridimensional é a orientação exterior. Através desta etapa, será estabelecida uma relação entre o sistema de câmara (obtido na orientação interior) e o terreno (espaço objeto) que está sendo mapeado. Pode-se dizer que a etapa de determinação da orientação exterior de uma imagem compreende a obtenção de seis parâmetros:

- Coordenadas no espaço objeto para o centro de perspectiva (obtenção das três coordenadas espaciais);
- Ângulos de rotação ou de atitude do sensor.

A figura a seguir, extraída de (BRITO *et al.*, 2002), ilustra os parâmetros a serem obtidos.



**Figura II.10 – Parâmetros a serem obtidos na orientação exterior**

Com a obtenção dos seis parâmetros anteriormente especificados e com o conhecimento da posição da imagem em relação à câmara (obtida na orientação interior), é possível deduzir as coordenadas no espaço-objeto (terreno) de qualquer

ponto representado no espaço-imagem, através da utilização de equações de colinearidade. Esse, segundo (BRITO *et al.*, 2002) é o principal objetivo da Fotogrametria.

O método de determinação dos seis parâmetros especificados recebe o nome de resseção espacial. Já o método de obtenção das coordenadas no espaço-objeto de um ponto representado no espaço imagem, é denominado interseção espacial.

#### II.4.7 FOTOTRIANGULAÇÃO

Os métodos de resseção e interseção espacial, apresentados anteriormente, apesar de possuírem enorme praticidade, por vezes deixam a desejar no que diz respeito a precisão das coordenadas obtidas. Neste contexto, surge como possível solução um método de obtenção de coordenadas denominado fototriangulação.

A fototriangulação, segundo (BRITO *et al.*, 2002), pode ser definida como “um conjunto de resseções espaciais realizado simultaneamente com um conjunto de interseções espaciais para um conjunto de imagens (bloco)”. O nome aerotriangulação vem da formação de triângulos no espaço, por ocasião da interseção espacial.

A utilização da aerotriangulação permite que sejam medidas as coordenadas “in loco” de apenas uns poucos pontos espaçados pelo bloco para calcular quantos pontos forem necessários para cada modelo. Este processo é normalmente conhecido como o “adensamento” de pontos de campo.

#### II.4.8 OBTENÇÃO DO MODELO DIGITAL DE SUPERFÍCIE

Atualmente, o conceito de mapeamento em geral envolve a implementação do conceito de modelo digital de superfícies. Também conhecido como MDS, o modelo digital de superfícies pode ser definido, segundo (BRITO *et al.*, 2002), como “qualquer representação numérica para uma determinada superfície física (relevo, por exemplo) do terreno a ser representada”.

Um MDS pode trazer várias informações como temperatura, pressão, declividade e altitude. Quando o MDS exprime alturas, este passa a ser conhecido como

Modelo Numérico de Elevações (MNE). Os processos fotogramétricos utilizam-se geralmente, apenas de altitudes ou alturas.

A obtenção do MNE envolve a aquisição de dados, e a utilização de alguns métodos. Existem vários métodos que podem ser utilizados, como:

- Digitação de coordenadas de ponto provenientes de aerotriangulação ou inspeção na carta;
- Digitalização de cartas em mesa digitalizadora;
- Aquisição via GPS;
- Extração fotogramétrica digital

A figura a seguir, obtida de (BRITO *et al.*, 2002), ilustra a extração de dados por meio da utilização de uma mesa digitalizadora.



**Figura II.11 – Extração através da utilização de mesa digitalizadora**

O último método citado considera inicialmente, a conversão das imagens a serem utilizadas para meio digital. Após a finalização desta etapa, deve ser realizada a orientação interior da imagem. A imagem obtida deve então passar pela obtenção da orientação exterior, com aquisição de seus parâmetros. Ao término desta etapa, caso o processo a ser seguido envolva a extração automática do MNE, deve ser realizado o processo denominado de normalização do par estereoscópico. Essa operação otimiza a localização automática de pontos homólogos. Ao final desta etapa é realizada a extração dos pontos tridimensionais propriamente ditos.

## II.4.9 RETIFICAÇÃO DE IMAGENS

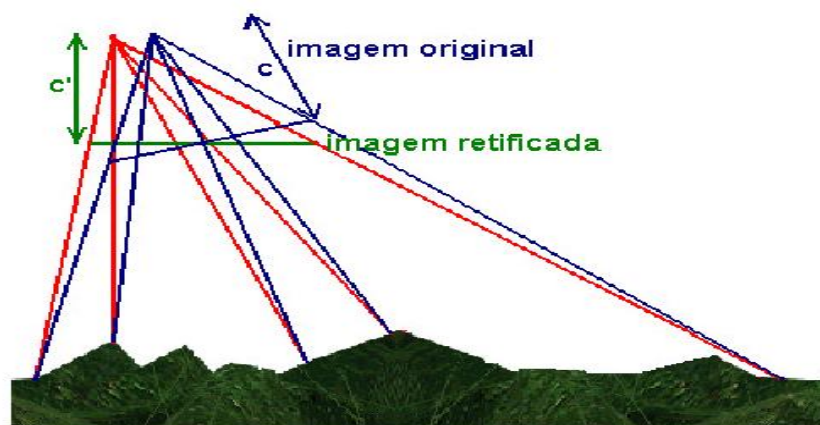
Segundo (ANDRADE, 1998), “retificar uma imagem consiste em projetá-la, segundo seu próprio feixe perspectivo, para um plano horizontal”. Através da utilização da retificação, é possível alterar os ângulos de atitude da câmara em relação a um referencial, assim como, a distância focal da imagem.

A retificação é utilizada pela fotogrametria aérea/orbital, para a geração de uma nova imagem vertical, sem as distorções introduzidas pela atitude do sensor durante a tomada da imagem. A imagem gerada poderá estar inclusive, livre de possíveis erros de deslocamento.

Para retificar uma imagem é necessária a utilização de modelos matemáticos para cálculo de suas novas coordenadas. Os modelos matemáticos aplicáveis para realização da retificação, são:

- Transformações (Afim, Polinomial, Projetiva, dentre outras);
- Princípio da colinearidade.

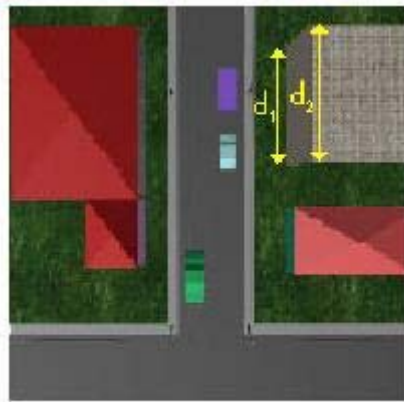
Os modelos matemáticos citados acima são abordados com mais detalhes na seção de cálculo numérico. A figura a seguir, extraída de (BRITO *et al.*, 2002), ilustra a modificação dos ângulos de atitude da câmara em decorrência da aplicação do processo de retificação sobre uma imagem.



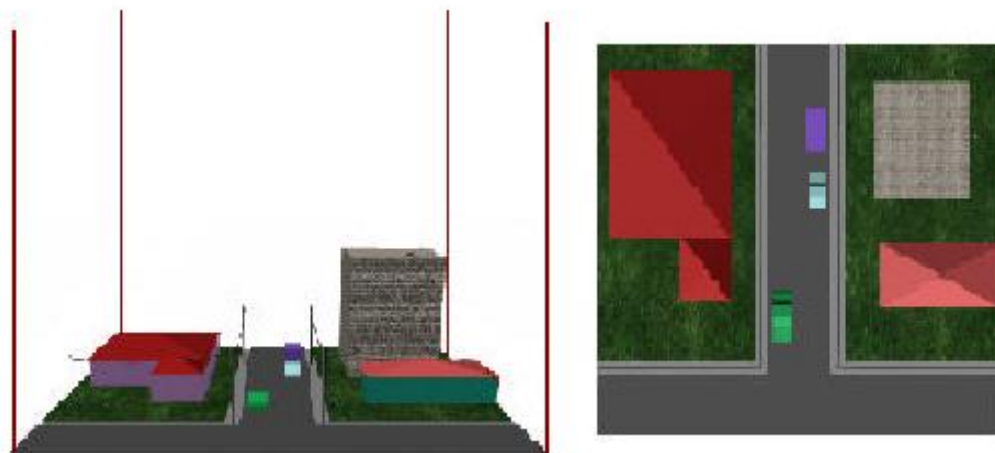
**Figura II.12 – Comparação entre imagem original e Imagem retificada**

#### II.4.10 ORTORETIFICAÇÃO DE IMAGENS

A ortorretificação de imagens pode ser entendida como um passo adicional a retificação para correção de distorções na imagem. A ortorretificação, além de corrigir as distorções relativas a rotação da câmara, possibilita a eliminação da distorção relativa ao relevo. Ela consiste na realização de transformações sobre as imagens adquiridas para obter uma projeção ortogonal das mesmas. As figuras a seguir, extraídas de (BRITO *et al.*, 2002), retratam a diferença entre imagens visualizadas sob perspectiva central (câmara) e imagens visualizadas em perspectiva ortométrica.



**Figura II.13 – Imagem adquirida através de câmara – deve-se notar a diferença entre as distâncias medidas na base e no topo do edifício, que na verdade são iguais.**



**Figura II.14 – Projeção Ortogonal da mesma imagem mostrada na figura II.12.**

Neste caso, a diferença entre as distâncias da base e do topo do edifício não é visualizada. Esta projeção permite a visualização da altura dos edifícios.

As Projeções ortogonais podem ser obtidas através da utilização de aparelhos denominados ortoprojetores ou, no caso da fotogrametria digital, através de transformações realizadas sobre números digitais dos *pixels* das imagens de origem. As transformações são implementadas através da utilização de modelos matemáticos abordados com mais detalhes na seção de cálculo numérico. Os principais modelos utilizados são a transformação afim, a projetiva e a retificação diferencial.

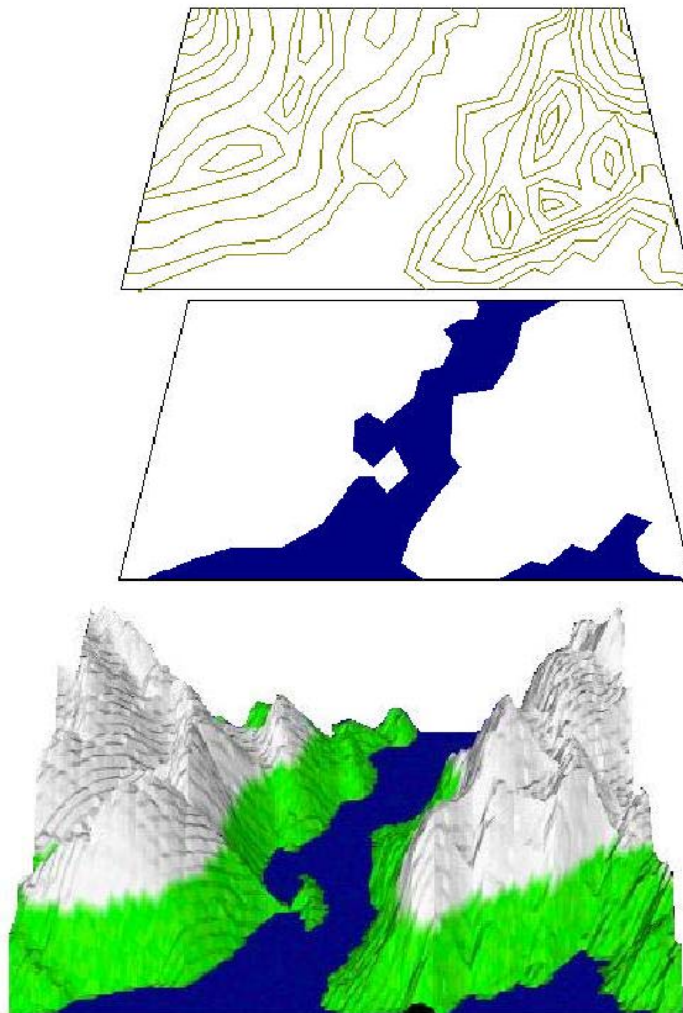
#### II.4.11 RESTITUIÇÃO DIGITAL

Os processos de extração de modelos digitais do terreno e (orto)retificação de imagens anteriormente apresentados possibilitam a obtenção de produtos cartográficos finais, que não necessitam de posterior processamento e/ou tradução. Entretanto, estes processos não conseguem alcançar um nível muito elevado no que diz respeito a interpretação de informações. Para que se consiga extrair o máximo de informações possíveis sobre o terreno em questão é necessária a utilização da restituição digital.

A restituição tem como objetivo reconhecer as diversas feições encontradas no terreno (por exemplo: rios, estradas, edificações etc.), obtendo suas coordenadas de forma a compor uma base cartográfica para a região, em uma dada escala. Tomando-se por base esta definição, segundo (BRITO *et al.*, 2002), deve-se entender como base cartográfica “o conjunto de objetos geograficamente referenciados a um determinado sistema de coordenadas”. Os objetos referenciados serão omitidos, ou representados de diversas formas, dependendo da escala utilizada.

Quando o processo de restituição ocorre em um ambiente digital, este passa a receber o nome de restituição digital. Atualmente, o estado da arte deste processo está na utilização de sistemas CAD (*Computer Aided Design*) com suporte a sistemas de coordenadas cartográficas. Esses sistemas utilizam a junção de um modelo da Terra (sistema geodésico) sobre o qual é aplicada uma projeção cartográfica. No Brasil utiliza-se o sistema de coordenadas UTM aplicado sobre um sistema geodésico que costuma ser o SIRGAS (Sistema de Referência Geocêntrico para a América do Sul), o SAD-69 (*South American Datum 1969*) ou o córrego alegre.

Atualmente, existem programas CAD que atuam em conjunto com outros de restituição digital. Em conjunto, eles podem obter as coordenadas do terreno, representando-o em arquivos digitais vetoriais. A representação do terreno ocorre com base em três tipos básicos de formas: pontos, linhas e áreas. A estas formas podem ser atribuídas outras características como cor, espessura, estilo e nível. Cabe destacar a importância da característica denominada nível. Esta característica permite uma melhor organização do resultado obtido por permitir a separação, em cada nível, das feições relacionadas entre si. Desta forma é possível visualizarmos diferentes feições como hidrografia, vegetação e transportes. Os níveis podem ou não ser exibidos, dependendo da preferência do usuário. A figura a seguir mostra os diferentes níveis obtidos com base em uma operação de restituição digital.



**Figura II.15 – Exemplo de dois níveis diferentes(hidrografia e altimetria) que junto com outros compõem a região restituída, representada abaixo deles.**

O resultado final de uma restituição, não é obtido automaticamente. Após a utilização do *software* de restituição, é necessária a intervenção humana para realizar um tratamento no resultado de forma a padronizá-lo de acordo com as normas vigentes na base cartográfica, assim como corrigir eventuais erros.

#### II.4.12 CONSIDERAÇÕES SOBRE FOTOGRAMETRIA

Conforme descrito nas seções anteriores, a fotogrametria utiliza uma série de técnicas e cálculos matemáticos para obtenção de informação confiável para obter um mapa de um determinado espaço-objeto estudado. A obtenção da informação é realizada com base em conceitos matemáticos, aplicados em conformidade com técnicas de fotogrametria, aliado a utilização da tecnologia computacional existente para sua execução. O estado da arte em fotogrametria está na utilização de estações fotogramétricas digitais que visam automatizar ao máximo possível a geração do mapa do terreno estudado.

Uma melhor compreensão do estado da arte na área de fotogrametria requer um entendimento mais preciso de como são compostas as estações fotogramétricas digitais e quais funções matemáticas são por elas processadas. Desta forma, as seções seguintes deste documento se propõem a abordar estes temas, inicialmente tratando da questão do cálculo matemático aplicado e posteriormente, no próximo capítulo, do projeto de montagem de uma estação fotogramétrica.

#### II.5 CÁLCULO MATEMÁTICO APLICADO À FOTOGRAMETRIA

A aplicação dos conceitos de Fotogrametria expostos nas seções anteriores está baseada em uma série de conceitos matemáticos. Esta seção se propõe a detalhar os principais conceitos utilizados na área da fotogrametria, tomando como base o projeto de implementação de uma estação fotogramétrica digital, que será descrita no capítulo seguinte.



## II.5.1 OPERAÇÕES BÁSICAS SOBRE MATRIZES

Fundamental para a representação, gerenciamento e tratamento de imagens, as matrizes constituem-se de um fundamento básico no estudo da Fotogrametria. Dessa forma, visando fundamentar o trabalho proposto nesta dissertação, apresenta-se a seguir os conceitos teóricos utilizados para a implementação das técnicas de fotogrametria utilizadas no projeto E-FOTO.

### II.5.1.1 DEFINIÇÃO DE MATRIZES

Tomando como base dois números  $m$  e  $n$  naturais e não nulos, denomina-se matriz de  $m$  por  $n$  ( $m \times n$ ) toda tabela  $M$  formada por números reais distribuídos em  $m$  linhas e  $n$  colunas. Como exemplo podemos citar a matriz:

$$A = \begin{bmatrix} 3 & 5 & -1 \\ 0 & 4/5 & 2 \end{bmatrix}$$

Que se constitui em uma matriz  $2 \times 3$ .

Os elementos de uma matriz são indicados através da notação  $a_{i,j}$ , onde  $i$  corresponde a linha e  $j$  corresponde a coluna do elemento  $a$ . Como exemplo, na figura anterior, o elemento  $a_{1,3}$  corresponde a  $-1$ .

### II.5.1.2 SOMA DE MATRIZES

Dadas duas matrizes  $A = (a_{ij})_{m \times n}$  e  $B = (b_{ij})_{m \times n}$ , diz-se que a soma  $A + B$  corresponde a matriz  $C = (c_{ij})_{m \times n}$ , tal que  $c_{ij} = a_{ij} + b_{ij}$ , para todo  $i$  e  $j$ . Este fato implica em que a soma de duas matrizes  $A$  e  $B$  do tipo  $m \times n$  seja uma matriz  $C$  do mesmo tipo, onde cada elemento é a soma dos elementos correspondentes em  $A$  e  $B$ .

Como exemplo podemos citar a soma a seguir:

$$C = \begin{bmatrix} 7 & 8 \\ 9 & 9 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 7+0 & 8+1 \\ 9+2 & 9+3 \end{bmatrix} = \begin{bmatrix} 7 & 9 \\ 11 & 12 \end{bmatrix}$$

A operação de soma de matrizes  $m \times n$  possui as seguintes propriedades:

- É associativa, ou seja,  $(A+B) + C = A + (B+C)$  para quaisquer matrizes  $A$ ,  $B$  e  $C$  do tipo  $m \times n$ ;
- É comutativa, ou seja,  $A + B = B + A$  para quaisquer matrizes  $A$  e  $B$  do tipo  $m \times n$ ;
- Tem elemento neutro;
- Todo elemento possui simétrico.

### II.5.1.3 PRODUTO DE MATRIZES

Dadas duas matrizes  $A = (a_{ij})_{m \times n}$  e  $B = (b_{jk})_{n \times p}$ , denomina-se produto  $AB$ , a matriz  $C = (c_{ik})_{m \times p}$  tal que:

$$c_{ik} = a_{i1} b_{1k} + a_{i2} * b_{2k} + a_{i3} * b_{3k} + \dots + a_{in} * b_{nk} \text{ para todo } i \in \{1, 2, \dots, m\} \text{ e todo } k \in \{1, 2, \dots, p\}.$$

A essa definição cabem algumas observações:

- O produto  $AB$  somente existe, se e somente se, o número de colunas de  $A$  for igual ao número de linhas de  $B$ , uma vez que  $A$  é do tipo  $m \times n$  e  $B$  é do tipo  $n \times p$ ;
- O produto  $AB$  será uma matriz com o número de linhas de  $A$  e o número de colunas de  $B$ , pois  $C = AB$  é do tipo  $m \times p$ ;

O elemento  $c_{ik}$  da matriz  $AB$  deve ser obtido através do seguinte procedimento:

- 1) Considera-se a linha  $i$  da matriz  $A$ ;
- 2) Considera-se a coluna  $k$  da matriz  $B$ ;
- 3) Coloca-se a linha  $i$  de  $A$  na vertical ao lado da coluna  $k$  de  $B$ ;
- 4) Calculam-se os  $n$  produtos dos elementos que ficaram lado a lado;
- 5) Somam-se esses  $n$  produtos, obtendo  $c_{ik}$ .

Como exemplo, pode-se citar a operação de produto entre as seguintes matrizes:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \text{ e } B = \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix}$$

Sendo A do tipo 2 x 3 e B do tipo 3 X 1, AB existe e é do tipo 2 x 1. Fazendo  $AB = C$ , devemos calcular  $c_{11}$  e  $c_{21}$ . Desta forma, calcula-se C:

$$C = \begin{bmatrix} (1*7) \\ (2*8) \\ (3*9) \\ (4*7) \\ (5*8) \\ (6*9) \end{bmatrix} = \begin{bmatrix} (7+16+27) \\ (28+40+54) \end{bmatrix} = \begin{bmatrix} 50 \\ 122 \end{bmatrix}$$

A operação de multiplicação de matrizes possui as seguintes propriedades:

- É associativa, ou seja,  $(AB)C = A(BC)$ ;
- É distributiva à direita em relação a adição, ou seja,  $(A+B)C = AC + BC$ ;
- É distributiva à esquerda, ou seja,  $C(A+B) = CA + CB$ ;
- Qualquer número k e as matrizes  $(a_{ij})_{m \times n}$  e  $B = (b_{jk})_{n \times p}$  temos que  $(kA)B = A(kB) = k(AB)$ .

#### II.5.1.5 MATRIZ TRANSPOSTA

Considerando-se uma matriz  $A = (a_{ij})_{m \times n}$ , denomina-se transposta de A, a matriz  $A^t = (a'_{ji})_{n \times m}$  tal que  $a'_{ji} = a_{ij}$ , para todo i e para todo j. Isto significa dizer que a primeira coluna de  $A^t$  é igual a primeira linha de A . Como exemplo, pode-se citar que:

$$\text{Se } A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \text{ então } A^t = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

Como propriedades da matriz transposta, pode-se verificar:

- $(A^t)^t = A$  para toda matriz  $A = (a_{ij})_{m \times n}$ ;
- Se  $A = (a_{ij})_{m \times n}$  e  $B = (b_{ij})_{m \times n}$ , então  $(A+B)^t = A^t + B^t$ ;
- Se  $A = (a_{ij})_{m \times n}$  e  $k \in \mathfrak{R}$ , então  $(kA)^t = kA^t$ ;
- Se  $A = (a_{ij})_{m \times n}$  e  $B = (b_{jk})_{n \times p}$ , então  $(AB)^t = B^t A^t$ .

#### II.5.1.6 INVERSÃO DE MATRIZES

Para se compreender a operação de inversão de matrizes, é necessário antes entender o conceito de matriz inversível. Diz-se que uma matriz  $A$  é inversível, se existir uma matriz  $B$  tal que  $AB = BA = I_n$ , onde  $I_n$  é uma matriz unidade de ordem  $n$  (matriz cujos elementos da diagonal principal são iguais a 1). Se  $A$  não é inversível, diz-se que  $A$  é uma matriz singular. Se  $A$  é inversível, então é única a matriz  $B$  tal que  $AB = BA = I_n$ .

Considerando-se uma matriz inversível  $A$ , denomina-se inversa de  $A$  a matriz  $A^{-1}$  (que é única) tal que  $AA^{-1} = A^{-1}A = I_n$ .

Como exemplo, pode-se citar:

$$A = \begin{bmatrix} 1 & 3 \\ 2 & 7 \end{bmatrix} \text{ e sua inversa } A^{-1} = \begin{bmatrix} 7 & -3 \\ -2 & 1 \end{bmatrix}$$

#### II.5.2 TRANSFORMAÇÕES

Importantíssimas para a fotogrametria, as transformações são fundamentais para a implementação de uma estação fotogramétrica, sendo utilizadas, por exemplo, nos procedimentos de orientação exterior e retificação. São apresentados a seguir os principais tipos de transformação, utilizados na implementação do projeto E-FOTO.

##### II.5.2.1 TRANSFORMAÇÃO AFIM

Segundo (BASTOS *et al.*, 2005), “a transformação afim consiste em, conhecendo-se as coordenadas de, no mínimo três pontos não-colineares no sistema

de coordenadas da imagem inicial e no sistema de coordenadas da imagem final, através de um ajustamento pelo método paramétrico, calcular os coeficientes de transformação entre ambos os sistemas”.

Considerando-se  $a_0, a_1, a_2, b_0, b_1, b_2$  como coeficientes de transformação, utiliza-se a seguinte formulação para a transformação afim geral:

$$\begin{aligned}x &= a_0 + a_1 \cdot x' + a_2 \cdot y' \\y &= b_0 + b_1 \cdot x' + b_2 \cdot y'\end{aligned}$$

Onde o par  $(x, y)$  representa o sistema de coordenadas de imagem final, enquanto que o par  $(x', y')$  representa o sistema de imagem inicial. Considerando-se que o objetivo da transformação é corrigir distorções de imagem causadas pela rotação da câmara em relação a um dado referencial, e que as imagens encontram-se em um eixo de coordenadas denominado linha e coluna, as equações anteriores podem ser reescritas como se segue:

$$\begin{aligned}\text{coluna} &= a_0 + a_1 \cdot \text{coluna}' + a_2 \cdot \text{linha}' \\ \text{linha} &= b_0 + b_1 \cdot \text{coluna}' + b_2 \cdot \text{linha}'\end{aligned}$$

A última condição para realizar o ajuste é fornecer pontos de controle. Devem ser fornecidos pelo menos as coordenadas de três pontos no objeto que sofrerá a transformação. Caso o objeto em questão seja retangular, é possível arbitrar seus cantos como 4 (quatro) pontos de controle. Organizando-se esta configuração na forma vetorial, considerando as coordenadas dos quatro pontos de controle, é possível obter:

$$\begin{bmatrix} \text{coluna1} \\ \text{linha1} \\ \text{coluna2} \\ \text{linha2} \\ \text{coluna3} \\ \text{linha3} \\ \text{coluna4} \\ \text{linha4} \end{bmatrix} = \begin{bmatrix} 1 & \text{coluna1}' & \text{linha1}' & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \text{coluna1}' & \text{linha1}' \\ 1 & \text{coluna2}' & \text{linha2}' & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \text{coluna2}' & \text{linha2}' \\ 1 & \text{coluna3}' & \text{linha3}' & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \text{coluna3}' & \text{linha3}' \\ 1 & \text{coluna4}' & \text{linha4}' & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \text{coluna4}' & \text{linha4}' \end{bmatrix} * \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ b_0 \\ b_1 \\ b_2 \end{bmatrix}$$

A partir desta estrutura podem-se obter os valores finais de **a0**, **a1**, **a2** e **b0**, **b1**, **b2** através do procedimento de ajustamento paramétrico. A estrutura matricial anterior pode ser expressa como:

$$L_b = AX^*$$

Onde:  $L_b$  é o vetor das observações,  $A$  é a matriz dos coeficientes dos parâmetros e  $X^*$  é o vetor dos parâmetros ajustados. Pode-se obter  $X^*$  através da equação a seguir:

$$X^* = (A^t A)^{-1} A^t P L_b$$

Onde  $P$  (normalmente igual a matriz identidade) é a matriz peso das observações.

#### II.5.2.2 TRANSFORMAÇÃO AFIM ISOGONAL

Variante da transformação afim, quando os eixos da imagem são ortogonais e o fator de escala é constante. Para este caso apresentam-se as equações lineares:

$$\begin{aligned} x &= a. \text{coluna} + b. \text{linha} + c \\ y &= -b. \text{coluna} + a. \text{linha} + d \end{aligned}$$

Os valores das equações anteriores, no entanto, não funcionarão com o caso da transformação entre coordenadas pixel e milímetros. Isto ocorre devido ao fato do eixo  $y$  do sistema analógico estar invertido em relação ao homólogo no sistema digital (pixels). Para corrigir este problema, faz-se um rebatimento no eixo  $y$ , obtendo-se as seguintes equações:

$$\begin{aligned} x &= a. \text{coluna} + b. \text{linha} + c \\ y &= b. \text{coluna} - a. \text{linha} + d \end{aligned}$$

Na forma matricial:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \text{coluna} & \text{linha} & 1 & 0 \\ \text{linha} & \text{coluna} & 0 & 1 \end{bmatrix} * \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

Os quatro parâmetros são então calculados por ajustamento linear obtendo-se a matriz A:

$$A = \begin{bmatrix} \text{coluna1} & \text{linha1} & 1 & 0 \\ \text{linha1} & \text{coluna1} & 0 & 1 \\ \text{coluna2} & \text{linha2} & 1 & 0 \\ \text{linha2} & \text{coluna2} & 0 & 1 \\ \text{coluna3} & \text{linha3} & 1 & 0 \\ \text{linha3} & \text{coluna3} & 0 & 1 \\ \text{coluna4} & \text{linha4} & 1 & 0 \\ \text{linha4} & \text{coluna4} & 0 & 1 \end{bmatrix}$$

### II.5.2.3 TRANSFORMAÇÃO PROJETIVA

Segundo (BRITO *et al.*, 2002), esta transformação requer, no mínimo, quatro pontos de controle para sua execução. Outro inconveniente, é que ela mapeia planos em planos, sendo desaconselhável para a retificação de superfícies tridimensionais (um terreno, por exemplo).

Para superfícies planas, ou aproximadamente planas (uma fachada, por exemplo), ela pode apresentar melhores resultados finais que a transformação afim.

A transformação projetiva é expressa com base nas seguintes equações:

$$x = (c_{11} x' * c_{12} y' * c_{13}) / (c_{31} x' * c_{32} y' * 1)$$

$$y = (c_{21} x' * c_{22} y' * c_{23}) / (c_{31} x' * c_{32} y' * 1).$$

### II.5.3 CONSIDERAÇÕES SOBRE O CÁLCULO APLICADO A FOTOGRAMETRIA

Os principais conceitos de fotogrametria atualmente em vigência baseiam-se em conceitos matemáticos. Dentre os fundamentos matemáticos existentes, os que mais se aplicam à área da fotogrametria são as operações com matrizes e as transformações.

### II.6 COMENTÁRIOS

Neste capítulo foram abordados os conceitos teóricos necessários para o entendimento deste trabalho. O capítulo seguinte apresentará o projeto E-FOTO, descrevendo seus objetivos e estágio atual de implantação.



## CAPÍTULO III – PROJETO E-FOTO

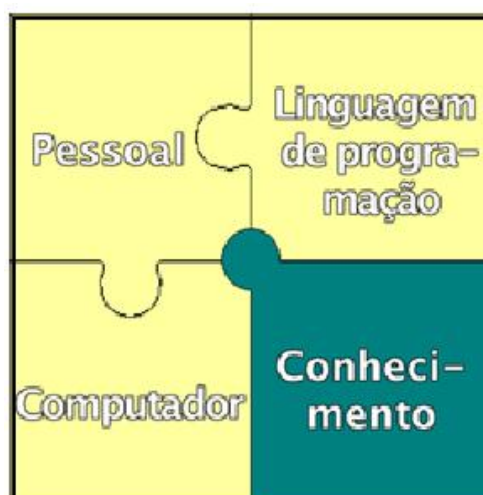
### III.1 INTRODUÇÃO

O estado da arte em fotogrametria está na utilização de estações fotogramétricas digitais que visam automatizar ao máximo possível a geração do mapa do terreno estudado.

As estações fotogramétricas começaram a ser utilizadas em larga escala a partir de 1995, sendo que no Brasil sua utilização começa a ser divulgada principalmente a partir de 1998. Hoje em dia, pode-se dizer que a maior parte das instituições produtoras de dados cartográficos emprega largamente estações fotogramétricas, ou pelo menos, pretende fazê-lo no mais curto período de tempo possível. Devido a esta realidade, a utilização e domínio dessas estações torna-se cada vez mais importante nos órgãos e instituições formadores de profissionais habilitados a trabalhar junto à produção cartográfica atual.

Atualmente, segundo (BRITO *et al.*, 2005), pode-se notar que, no Brasil, as instituições de ensino e pesquisa de uma maneira geral apresentam escassez de equipamentos modernos, causada devido à falta de recursos e elevado preço da tecnologia disponível no mercado. O que se percebe é que a maioria das instituições, devido à falta de recursos, termina por utilizar equipamentos defasados em seus laboratórios de ensino e pesquisa. Este fato compromete a formação dos profissionais da área da fotogrametria, que, em geral, concluem seus cursos sem nenhum treinamento formal em fotogrametria digital. Desta forma, com base no contexto apresentado, inicia-se o projeto E-Foto.

O E-Foto surge com o objetivo de criar, desenvolver e gerenciar uma estação fotogramétrica digital educacional. Com o intuito de facilitar o ensino e a pesquisa na área de fotogrametria, o projeto considera a implementação da estação sobre uma plataforma de hardware e *software* que demande o menor custo possível. Como Pontos-chave para a implementação do projeto foram considerados: pessoal (pessoas envolvidas), equipamento (computadores), tecnologia (linguagem de programação) e conhecimento para atuar de forma integradora aos três primeiros pontos-chave. A figura a seguir, obtida de (BRITO *et al.*, 2003), procura ilustrar os pontos fundamentais para o projeto.



**Figura III.1 – Pontos Chave para a implementação do Projeto E-Foto**

Como anteriormente citado, a implementação do projeto considera o estado da arte na fotogrametria digital. Desta forma, a implementação de uma estação fotogramétrica deve levar em consideração os conceitos atualmente vigentes no âmbito do mercado no que diz respeito aos componentes da mesma. Visando esclarecer este ponto, a seção a seguir objetiva fornecer conceitos básicos sobre o que é, e como é composta uma estação fotogramétrica digital.

### III.2 ESTAÇÃO FOTOGRAMÉTRICA DIGITAL

Conforme citado anteriormente, uma estação fotogramétrica digital pode ser entendida como um equipamento computadorizado que visa automatizar ao máximo o processo de mapeamento digital de um terreno estudado. Estes equipamentos constituem-se de estações de trabalho inteiramente dedicadas à fotogrametria.

O padrão atual de estações fotogramétricas digitais envolve a utilização de uma série de itens de hardware e *software*. Estes itens podem variar de acordo com o fabricante e os modelos de estação existentes. Entretanto, de uma maneira geral, uma estação fotogramétrica digital deve considerar os seguintes itens:

### III.2.1 ITENS DE HARDWARE

- Monitor(es) de vídeo;
- Dispositivo de visão estereoscópica – Óculos, monitores especiais, dentre outros;
- CPU gráfica, ou placa de vídeo especial;
- CPU geral;
- Periféricos de Entrada/Saída – Teclado, *mouse*, *scanner*, mesa digitalizadora, *plotter* e/ou impressora;
- Dispositivo de medição estereoscópica – *Trackball*, topo-*mouse*.

A figura a seguir, ilustra uma estação fotogramétrica digital, extraída de (BRITO et al., 2002).



**Figura III.2 – Estação fotogramétrica *Image Station 2001* (fonte: *ImageStation factsheet*)**

No que diz respeito aos monitores de vídeo, costuma-se utilizar como padrão telas com dimensões entre 19 e 21 polegadas. Algumas estações podem apresentar monitores de vídeo de dimensões diferentes das usuais, com o objetivo de maximizar a visualização estereoscópica.

Quanto às máquinas utilizadas, há modelos de vários fabricantes. As mais populares são estações *Silicon Graphics* com sistema operacional *Unix* ou *PC's* com sistema operacional *Windows*.

Os sistemas de visão estereoscópica possuem quatro principais tipos de solução:

- **Separação espacial** – Nesta solução, a tela do monitor exibe o par lado a lado e utiliza-se um estereoscópio de espelhos para visualizá-las estereoscopicamente;
- **Anaglifo** – Este tipo de solução contempla a exibição de ambas as imagens uma sobre a outra, porém em cores diferentes. É necessária a utilização de um óculos especial com uma lente;
- **Separação radiométrica** – Solução semelhante a do anaglifo, apresenta polarizações de luz diferentes;
- **Separação temporal** – Esta modalidade exibe alternadamente as imagens do modelo, a altas frequências.

De acordo com (BRITO *et al.*, 2002), “Os sistemas mais modernos utilizam uma combinação do princípio da polarização (passiva, se a tela de polarização é montada no monitor e ativa, se a tela de polarização é montada nos óculos) e da separação temporal”.

### III.2.2 MÓDULOS DE *SOFTWARE* DEDICADOS A OPERAÇÕES FOTOGRAMÉTRICAS

- Orientação interior;
- Orientação relativa e orientação absoluta, ou Orientação exterior (relativa + absoluta simultaneamente);
- Aerotriangulação (medição e cálculo);
- Restituição, retificação e reamostragem de imagens (geometria epipolar);
- Extração de modelos digitais do terreno;
- Geração de orto-imagens;
- Integração com sistemas de informação geográfica.

O advento das estações fotogramétricas digitais e dos programas de fotogrametria digital trouxe alguns fatos bastante positivos para a área da fotogrametria. Como exemplo, há a possibilidade de, mesmo sem o equipamento completo, serem executadas várias operações (ainda que de maneira não-otimizada). Outro ponto a favor

é a concentração de todas os procedimentos e rotinas em um só computador, poupando espaço e ganhando em versatilidade e tempo de operação. Antes da existência das estações fotogramétricas digitais, as fotos precisavam passar de aparelho para aparelho, de especialista a especialista, em diferentes áreas (e até em diferentes organizações) para finalmente ter a restituição do terreno concluída.

Outro ponto bastante facilitado foi a entrada de dados. Como se utiliza rotinas genéricas, o *software* aceita imagens em quaisquer formatos, inclusive provenientes de câmaras não-fotogramétricas (cujas fotos não têm um sistema de coordenadas a elas associado). Antes da utilização das estações fotogramétricas digitais, os aparelhos restituidores somente aceitavam fotos de um determinado padrão, provenientes de câmaras de distâncias focais de valores constantes.

Com base no exposto acima, o projeto E-FOTO procura implementar uma estação fotogramétrica digital educacional com o máximo de funções possível, visando a automatização do processo de mapeamento digital do terreno a ser estudado. A seguir, será mostrado como o E-FOTO foi estruturado para alcançar seu objetivo.

### III.3 METODOLOGIA UTILIZADA PARA IMPLEMENTAÇÃO DO E-FOTO

Conforme citado anteriormente, o projeto E-FOTO tem como objetivo a construção de uma estação fotogramétrica digital educacional. Como pilares para a obtenção deste objetivo estão a gratuidade e o auto-aprendizado. A figura a seguir, obtida de (BRITO *et al.*, 2003), ilustra os pilares do projeto E-FOTO.



**Figura III.3 – Pilares para implementação do E\_FOTO**

No que diz respeito à gratuidade, de forma a diminuir ao máximo o impacto do custo para sua utilização por parte de instituições de ensino e pesquisa, todo o projeto foi definido e construído sobre plataforma de *software* livre, sendo portátil para utilização em várias plataformas. Dentre as licenças de *software* que se enquadram na categoria de “*software* livre”, foi adotada a GNU GPL (GNU *General Public License*) para os programas e a GNU FDL (GNU *Free Documentation License*) para a documentação elaborada. A linguagem de programação adotada para a implementação dos algoritmos fotogramétricos foi a linguagem C++. Esta escolha ocorreu devido a alguns fatos de fundamental importância: Ser o C++ uma linguagem orientada a objeto – facilitando assim o trabalho de programação; possuir extrema popularidade no meio acadêmico facilitando a manutenção e expansão do projeto; possuir o C++ um *kit* de ferramentas para construção de interfaces gráficas inteiramente gratuito e multiplataforma, denominado Qt.

O segundo pilar do E-FOTO está diretamente relacionado com o âmbito educacional do projeto. A filosofia do auto-aprendizado é fundamental, não somente para a manutenção e o contínuo desenvolvimento da estação fotogramétrica, mas também para a formação de melhores profissionais nas áreas de fotogrametria digital e, por que não, informática. Com este intuito, o projeto busca atender a diferentes perfis de usuários, no que diz respeito a sua interação com o sistema e nível de aprendizado:

Usuários de primeiro nível de aprendizado: Neste perfil se encaixam os usuários que desejam apenas realizar alguma tarefa fotogramétrica, utilizando os módulos em execução. Para atender a este tipo de usuário, há uma ajuda *online* de cada um dos módulos, que abrange sua utilização e conceitos básicos de fotogrametria.

Usuários de segundo nível de aprendizado: Usuários com este perfil, além de aprender como utilizar os programas desejam entender o seu funcionamento. Para isto, foi criado um tutorial denominado *e-book*, que cobre todos os principais conceitos da fotogrametria. Este tutorial possui enfoque auto-explicativo, e procura ensinar não apenas os princípios teóricos, mas também a montagem das equações e algoritmos, além de comparações entre diferentes métodos e resultados. O tutorial prioriza a abordagem da auto-aprendizagem, desta forma, a disciplina é apresentada de modo que não é necessário conhecimento prévio em fotogrametria.

Usuários de terceiro nível de aprendizado: Possuem interesse no desenvolvimento do código do sistema e, as informações e conhecimento obtido no

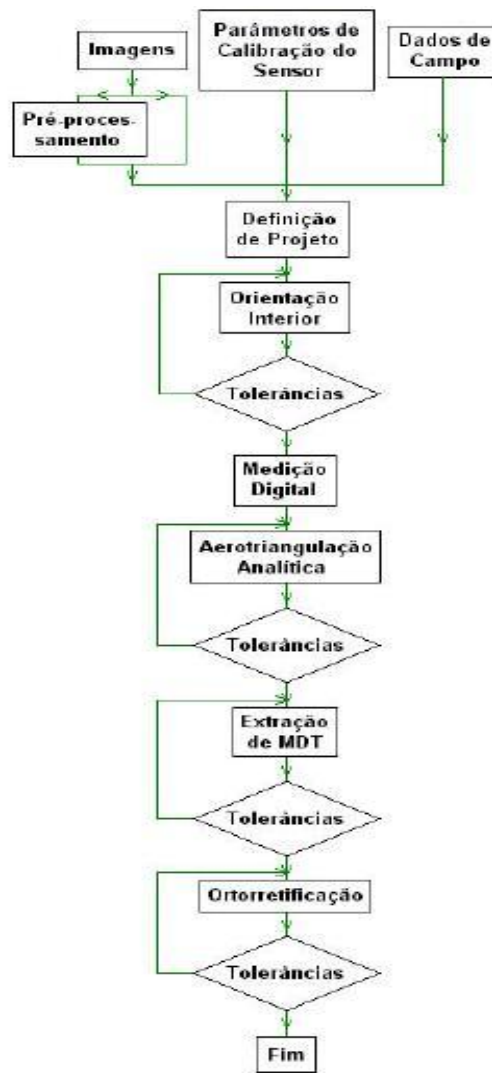
material disponibilizado para os usuários do primeiro e segundo nível. Podem participar ativamente do projeto, melhorando o código, enviando sugestões e desenvolvendo novos módulos e textos.

A figura a seguir, obtida de (BRITO *et al.*, 2003), ilustra os níveis de aprendizado que dividem os perfis de usuários da estação fotogramétrica construída no E-FOTO.



**Figura III.4 – Os três níveis de aprendizado obtidas com o E-FOTO**

A elaboração do E-FOTO compreende o desenvolvimento de módulos que implementem funcionalidades necessárias para a geração do mapeamento digital de um dado espaço-objeto (terreno). Este mapeamento deve ser realizado com base em um fluxo, que possui atividades estruturadas segundo uma ordem necessária a conclusão do trabalho. A figura a seguir, extraída de (COELHO, 2002), ilustra o fluxo de atividades considerado pelo E-FOTO.



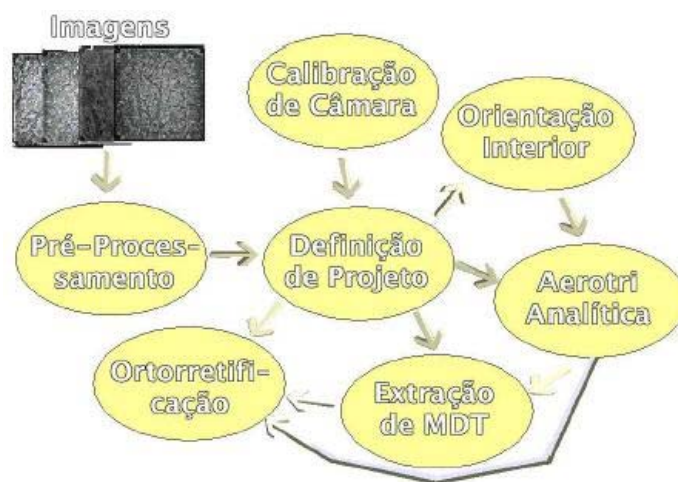
**Figura III.5 – Fluxo de atividades para mapeamento de terreno**

Desta forma, com base nos conceitos e fluxo de trabalho anteriormente expostos, para a elaboração da estação fotogramétrica foram definidos os seguintes módulos:

- Pré-processamento de imagens;
- Definição de parâmetros de projeto;
- Calibração de câmara;
- Orientação interior monoscópica;
- Medição digital e Aerotriangulação analítica (resseção espacial + interseção espacial de um bloco de imagens);
- Extração de um modelo digital do terreno;
- Ortorretificação;
- Restituição digital (em ambiente CAD);



A figura a seguir, retirada de (BRITO *et al.*, 2003), ilustra os módulos considerados para o projeto E-FOTO, assim como o relacionamento que possuem entre si.

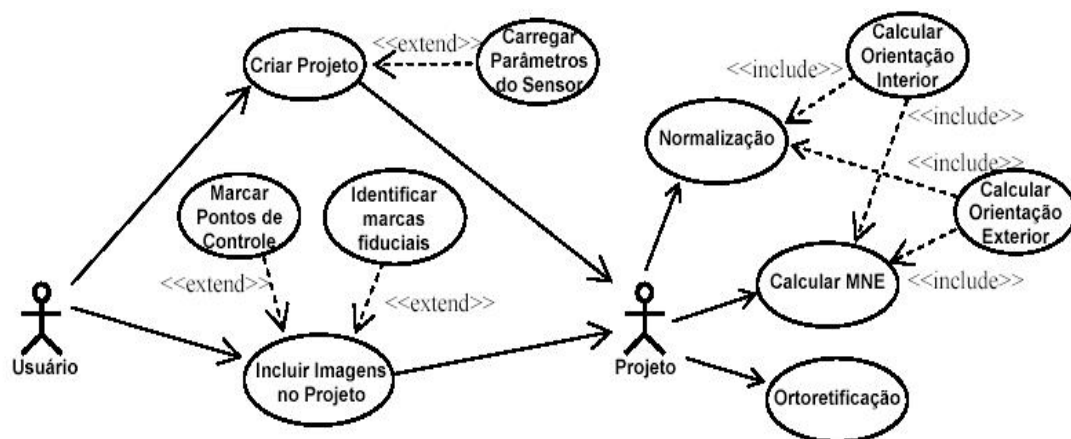


**Figura III.6 – Diagrama de módulos do E-FOTO**

Visando facilitar a integração entre os diversos módulos, assim como, adequá-los as boas práticas da Engenharia de *Software*, em (BRITO *et al.*, 2005), optou-se pelo uso da linguagem unificada de modelagem (UML – *Unified Modeling Language*) para a realização da análise do sistema. A UML foi adotada por trazer vantagens como: diminuição da complexidade, simplificação da análise realizada, visualização de possíveis problemas futuros e a possibilidade de efetuar simulações.

Dessa forma, a modelagem de *software* adotada no E-FOTO passa a seguir o padrão de metodologia estabelecido pelo OMG (*Object Management Group*), organização internacional que procura padronizar métodos voltados a orientação a objetos. Esse padrão se baseia em três notações: OOD – *Design* orientado a objetos, OMT – Técnica de Modelagem de Objetos e OOSE Engenharia de *Software* orientado a objetos (BRITO *et al.*, 2005).

Através da análise do E-FOTO foram obtidos os diagramas de caso de uso e de classes para este projeto. A figura a seguir, extraída de (BRITO *et al.*, 2005) ilustra o diagrama de casos de uso gerado.

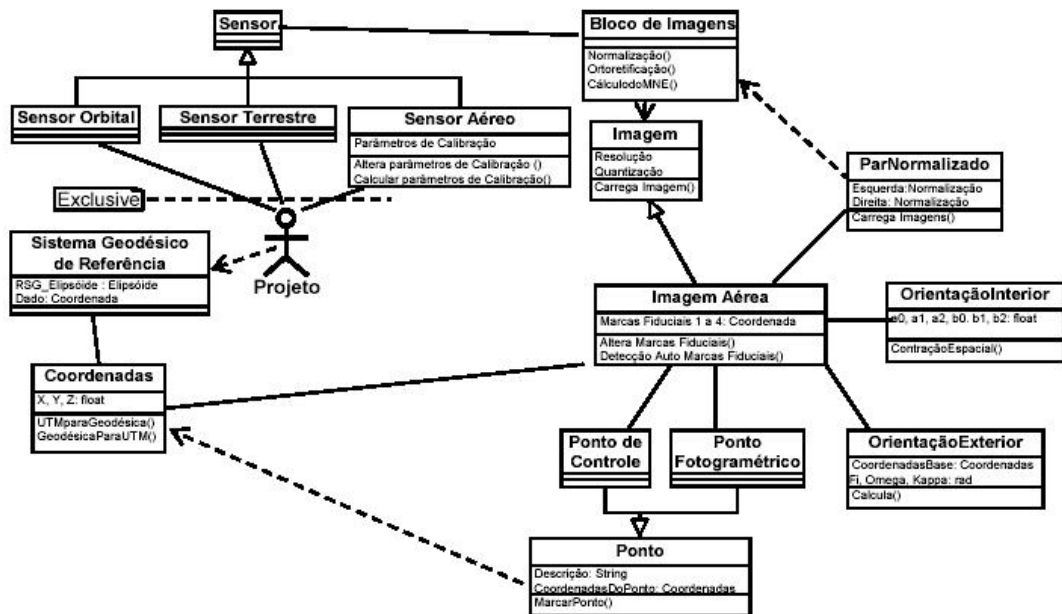


**Figura III.7 – Diagrama de casos de uso do E-FOTO**

O diagrama anterior apresenta as principais funcionalidades previstas para o E-FOTO. Com base neste contexto foram identificados dois atores (sistema ou pessoa que interaja com o sistema proposto, sendo representado por um ícone que lembra o desenho rupestre de um ser humano): *usuário* e *projeto*.

O *usuário* pode ser entendido como o operador da estação fotogramétrica, que irá executar os casos de uso (representados por elipses correspondem às funcionalidades que o sistema deve possuir) criar projeto e incluir imagens no projeto. O ator *projeto*, após o término da execução da inclusão de imagens do projeto, executa os casos de uso calcular MNE e ortorectificação.

Outro diagrama importante para o projeto, é o diagrama de classes. Através dele, são apresentadas as estruturas e algoritmos contemplados no sistema, organizados em classes de objetos. A figura a seguir, extraída de (BRITO *et al.*, 2005), apresenta o diagrama de classes utilizado no projeto E-FOTO.



**Figura III.8 – Diagrama de Classes do projeto E-FOTO.**

No diagrama de classes apresentado na figura III.8, a classe *Projeto* envolve a estrutura para a representação do ator projeto. Esta classe apresenta dependência com a classe *Sistemas Geodésicos de Referência*. Isto faz com que haja obrigatoriedade da escolha de um sistema, no momento da criação de um projeto fotogramétrico. Em adição a esta relação, a classe *Projeto* está relacionada a uma das subclasses de *Sensor* fotogramétrico: *Orbital*, *Terrestre* ou *Aéreo*.

Cabe lembrar, que a linha pontilhada que cruza os relacionamentos entre projeto e as subclasses de *Sensor* representa em UML que estes relacionamentos são mutuamente excludentes. Isto quer dizer que um *Projeto* pode estar associado a um único tipo de sensor.

A superclasse *Sensor* utilizada no projeto está associada à classe *Bloco* fotogramétrico, que é composta por uma ou mais imagens adquiridas através do sensor. A classe *Bloco* contempla ainda a implementação dos algoritmos de normalização, cálculo do MNE e ortorretificação.

A classe *Par Normalizado* implementa os parâmetros de normalização de um par de imagens, onde a área coberta pode ser observada. Esta classe depende da instanciação de um *Bloco de Imagens*, estando associada a duas das imagens contidas no bloco. Os algoritmos de normalização, cálculo do MNE e ortorretificação são implementados como métodos.

Outra classe fundamental, a classe *Imagem*, também é uma superclasse. A partir dela são derivadas as subclasses *Aérea*, *Terrestre* e *Orbital*. A classe *Aérea* é relacionada não só a um objeto das classes *Orientação Exterior* e *Orientação Interior*, mas também às subclasses da superclasse *Pontos*: *Pontos de Controle* e *Pontos Fotogramétricos*.

Um ponto importante a ser observado, diz respeito a implementação de conjuntos de métodos. O usuário poderá utilizar os métodos implementados, selecionando o mais conveniente dentre eles. Como exemplo, pode-se citar os métodos *altera parâmetros de calibração* e o método *calcula parâmetros de calibração* localizados na classe *Sensor Aéreo*. Esses métodos fornecem maneiras diferentes para a aquisição dos parâmetros de calibração.

De uma maneira geral, pode-se dizer que a utilização da UML atrelada a metodologia da OMG tem sido fundamental para a organização do projeto. Sua utilização melhora facilita a compreensão do relacionamento dos módulos já disponíveis, assim como, o planejamento das implementações futuras. Visando esclarecer o planejamento considerado, bem como, quais módulos já se encontram disponíveis, é apresentada a seguir a situação atual do E-FOTO.

#### III.4 SITUAÇÃO ATUAL DO PROJETO

Iniciado em 2002, tendo como marco a elaboração de um projeto de graduação no Instituto Militar de Engenharia, no Rio de Janeiro, o projeto E-FOTO foi crescendo pouco a pouco, através de implementações sucessivas, sempre tendo como meta a produção científica de alunos e professores. Atualmente, o E-FOTO já possui uma série de módulos implementados, assim como, uma série de publicações acadêmicas e científicas produzidas. Todo o material, inclusive o código fonte dos módulos produzidos, encontra-se disponível para *download* no sítio: <http://www.efoto.eng.uerj.br>. O quadro a seguir ilustra de maneira resumida os módulos implementados.

**Tabela III.1 – Módulos implementados do Projeto E-FOTO**

<b>Módulo</b>
Pré-processamento de imagens
Definição de parâmetros de projeto
Calibração de câmara
Orientação interior
Medição digital e Aerotriangulação analítica
Extração de um modelo digital do terreno
Ortorretificação
Restituição digital

Todas as funcionalidades fotogramétricas relativas aos módulos citados na tabela III.1 já foram implementadas. O desafio que a equipe do projeto enfrenta atualmente é integrar os módulos, o que não é trivial.

Em adição aos módulos inicialmente previstos, durante o desenvolvimento do projeto, foi constada a viabilidade de produção de módulos adicionais, que podem ser utilizados como base para a produção dos módulos previstos inicialmente. Foram produzidos os seguintes módulos:

- Módulo de medição e visualização estereoscópica;
- Módulo de retificação de imagens digitais.

Cabe lembrar que o projeto E-FOTO segue em pleno desenvolvimento. A situação do projeto apresentada neste trabalho pode e deve ser alterada com o passar do tempo com a liberação de novos módulos do projeto.

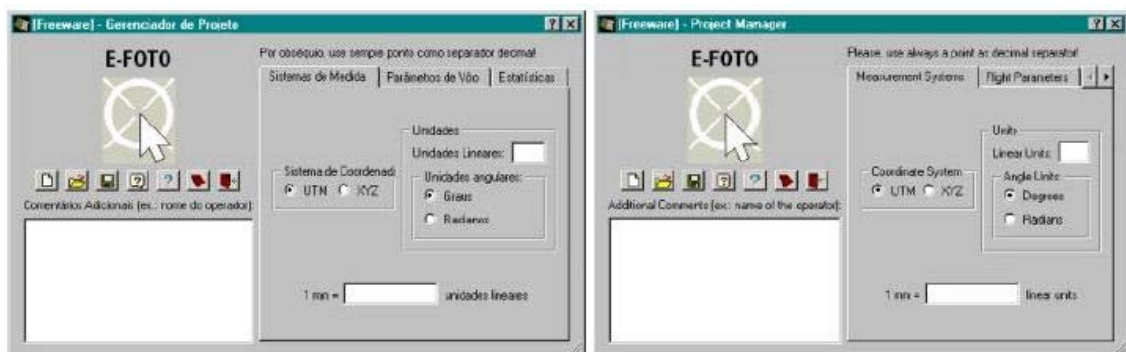
De forma a facilitar o entendimento sobre os módulos implementados, apresenta-se a seguir uma descrição sucinta de cada um.

### III.5 MÓDULOS IMPLEMENTADOS

Conforme anteriormente citado, os módulos implementados para o projeto E-FOTO são de livre distribuição, estando disponíveis no *Web site* do projeto. Apresenta-se a seguir uma breve descrição dos principais módulos produzidos e documentados.

### III.5.1 DEFINIÇÃO DE PARÂMETROS DO E-FOTO

Este módulo corresponde a configuração de parâmetros para o projeto. Através dele é possível inserir valores em campos necessários para a correta execução dos demais módulos do projeto. Os valores inseridos são armazenados em arquivo no formato texto para posterior recuperação pelo sistema. A figura a seguir, extraída de (COELHO, 2002), ilustra a interface criada para a captura dos parâmetros de configuração.

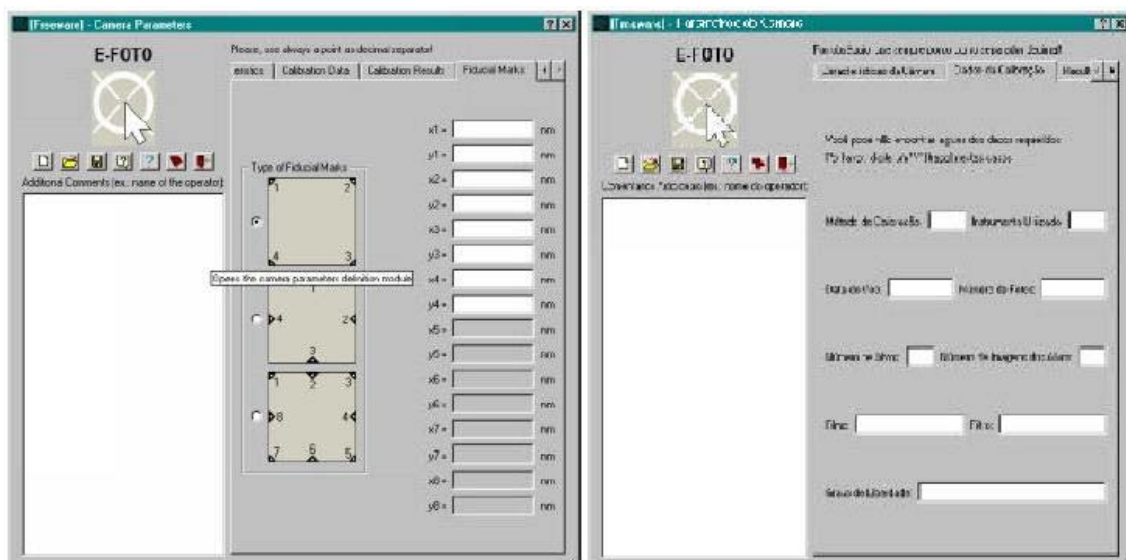


**Figura III.9 – Interfaces de configuração do projeto E-FOTO**

Em adição a estas funcionalidades, este módulo também é responsável pelo cálculo e apresentação do coeficiente de refração do ar. Este parâmetro é necessário para a utilização dos demais módulos do projeto.

### III.5.2 CALIBRAÇÃO DE CÂMARA

Este módulo é responsável por captar e armazenar os parâmetros necessários para a calibração da Câmara utilizada no processo de aquisição de imagens. Possui funcionamento similar ao módulo de definição de parâmetros do E-FOTO. Os valores inseridos nos campos disponibilizados pela interface são armazenados em arquivo no formato texto para posterior recuperação e utilização pelos demais módulos do sistema. A figura a seguir, extraída de (COELHO, 2002), ilustra a interface deste módulo.

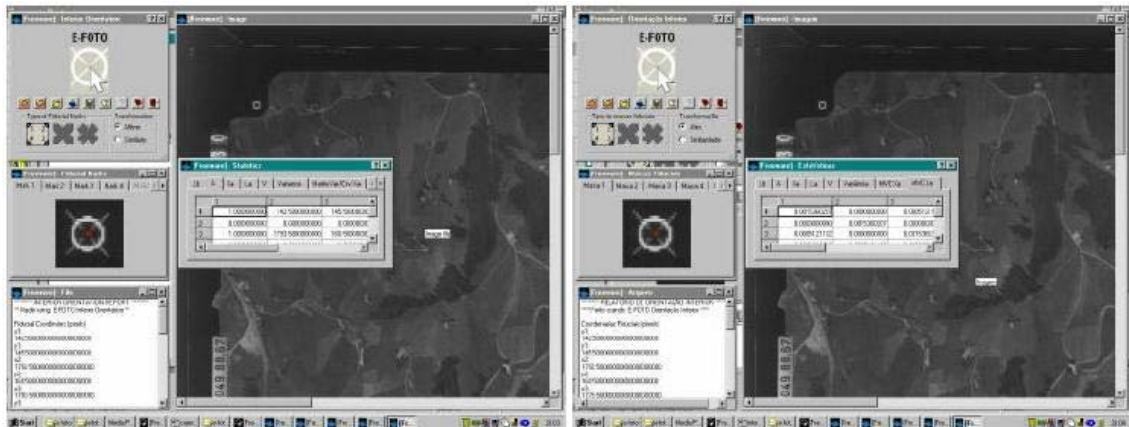


**Figura III.10 – Interface de calibração de câmara do E-FOTO**

### III.5.3 ORIENTAÇÃO INTERIOR

Responsável por referenciar a imagem em relação a câmara, o módulo de orientação interior foi implementado em uma interface que possui 4(quatro) janelas. A primeira delas contém os comandos do programa em execução. A segunda janela é responsável pela visualização das marcas fiduciais. A terceira janela, localizada abaixo da segunda, fornece ao usuário uma cópia dos parâmetros de orientação interior a serem armazenados. A Quarta janela é a janela principal que apresenta a imagem contemplada.

De uma maneira geral, o módulo de orientação interior permite que o usuário informe quais marcas fiduciais gostaria de considerar (que opcionalmente também podem ser capturadas de maneira automática). De posse destas marcas, o usuário define qual método gostaria de utilizar para gerar a orientação. Existem duas opções possíveis: Afim geral e similaridade. Uma vez escolhido o método o processo pode ser executado. Ao final de sua execução, é exibida ao usuário uma interface com estatísticas oriundas do processo, bem como, informação sobre a qualidade do ajustamento realizado. A seguir é apresentada uma figura, extraída de (COELHO, 2002), que ilustra a interface do módulo de orientação interior.



**Figura III.11 – Interface do módulo de orientação Interior do Projeto E-FOTO**

### III.5.4 MEDIÇÃO DIGITAL

Este módulo tem como objetivo realizar a medição de coordenadas no sistema imagem de pontos de campo e fotogramétricos e armazená-los em um arquivo para posterior processamento pelo módulo de aerotriangulação. A figura a seguir, obtida de (COELHO, 2002), ilustra a interface desenvolvida para este módulo.



**Figura III.12 – Interface do módulo de medição digital**



### III.5.5 MEDIÇÃO E VISUALIZAÇÃO ESTEREOSCÓPICA

Visando resolver o problema da paralaxe esteresópica, “deslocamento aparente da posição de um corpo ou objeto em relação a um ponto ou sistema de referência causado pela mudança de ponto de observação” (BRITO *et al.*, 2005), foi implementado o módulo de medição e visualização estereoscópica.

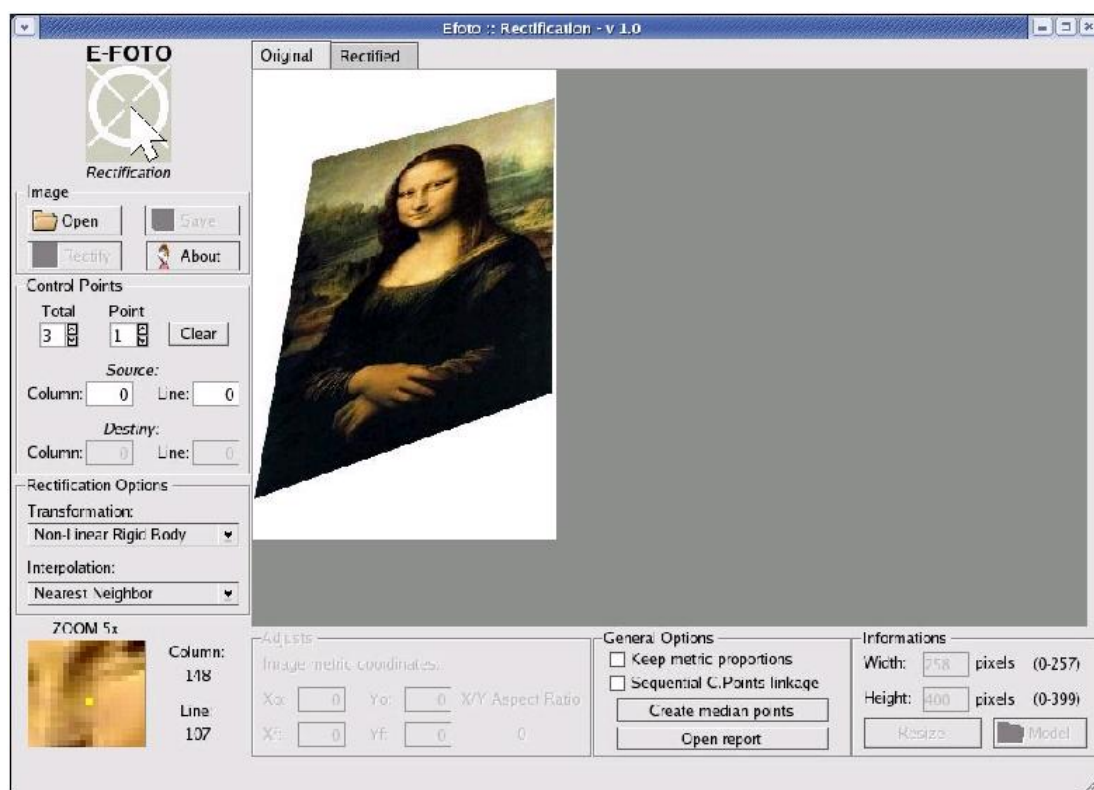
De uma forma geral, o sistema permite ao usuário realizar medições de pontos sobre determinada imagem visualizada. As medições são realizadas com base em dois modos de visão: anaglifo e estereoscópio. Também é possível realizar a medição da altura de um objeto através da marcação do topo e da base do objeto e a sua comparação em relação a mesma imagem em outro modo de visão. A figura a seguir, extraída de (BRITO *et al.*, 2005), ilustra a interface utilizada pelo módulo para a medição da altura de objetos.



**Figura III.13 Interface do módulo de medição e visualização estereoscópica**

### III.5.6 RETIFICAÇÃO DE IMAGENS DIGITAIS

Este módulo tem como objetivo realizar a retificação de uma imagem fornecida, gerando como saída a projeção da imagem já retificada. Resumidamente, o sistema permite ao usuário selecionar a imagem a ser retificada, escolher o tipo de transformação a ser utilizada, inserir como parâmetro os pontos de controle a serem utilizados e executar a operação de retificação. A versão atual do sistema permite a escolha de uma entre duas, das seguintes transformações: afim caso geral e afim caso isogonal. Uma vez escolhida a transformação e executada a operação o sistema apresenta uma interface com a imagem retificada. A figura a seguir, extraída de (COELHO, 2002), ilustra a interface de retificação de imagens desenvolvida para o projeto E-FOTO.



**Figura III.14 – Interface de retificação de imagens digitais do E-FOTO**

### III.6 COMENTÁRIOS

Conforme anteriormente apresentado, o projeto E-FOTO tem como objetivo principal a implementação de uma estação fotogramétrica digital com foco educacional. A construção de tal estação visa suprir a necessidade de acesso à tecnologia de última geração na área da fotogrametria digital por parte das instituições de ensino e pesquisa não somente do Brasil, mas dos países em desenvolvimento.

A construção da estação envolve implementação de sistemas com foco no processamento de cálculos matemáticos. Implementações desta natureza precisam ser corretamente medidas e estimadas para que o planejamento de sua construção seja o mais perfeito possível. O planejamento adequado leva um melhor aproveitamento dos recursos utilizados para a sua implementação, o que é fundamental nos dias de hoje face à carência de verbas das instituições de ensino e pesquisa. Visando suprir a necessidade latente da mensuração de esforço para desenvolvimento de *software* aplicado ao processamento matemático, propõe-se, no capítulo a seguir, uma métrica para *softwares* desta natureza.

## CAPÍTULO IV – PROPOSTA DE MÉTRICA

### IV.1 INTRODUÇÃO

Até o presente capítulo, foram apresentados os fundamentos teóricos de métricas de *software*, tendo como foco a métrica de Pontos de Função, conceitos de fotogrametria, cálculo numérico aplicado à fotogrametria e, finalmente, foi abordado o projeto E-FOTO.

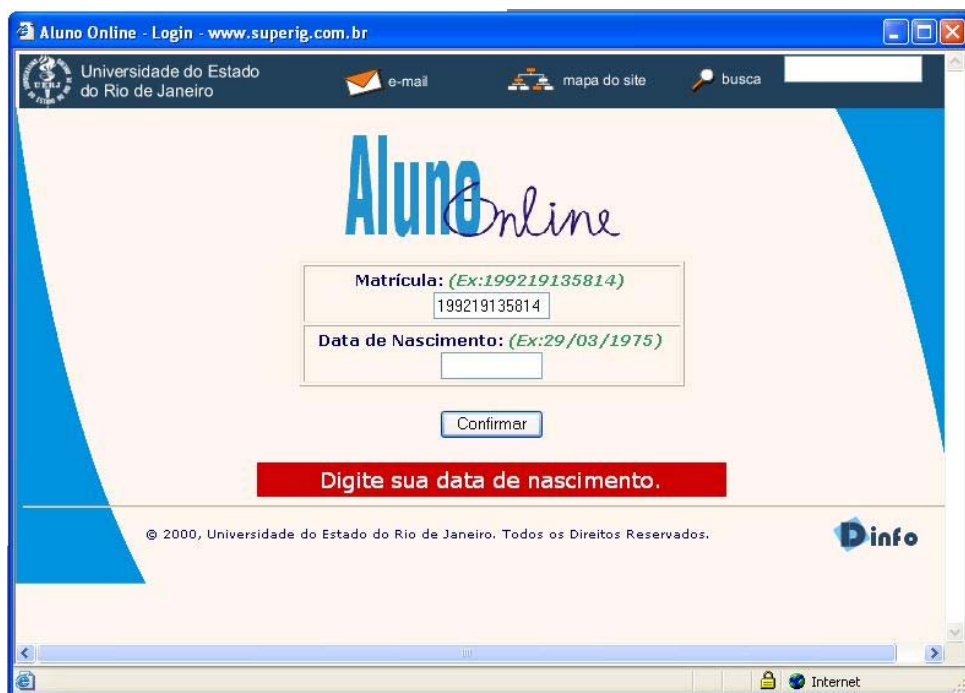
O capítulo atual utiliza-se da base conceitual anteriormente exposta para propor uma alternativa à utilização da métrica de pontos de função, tal como atualmente é utilizada, para dimensionamento de projetos de *software* aplicados à fotogrametria digital. Inicia-se com a uma análise da métrica de pontos de função, expondo possíveis problemas na sua utilização em projetos que envolvam larga utilização de cálculos matemáticos. A seguir é proposta uma modificação na métrica atualmente utilizada para enfrentar projetos de fotogrametria digital. Ao final do capítulo, exemplifica-se o uso da métrica proposta tomando como base a implementação realizada para o projeto E-FOTO.

### IV.2 A ANÁLISE DE PONTOS DE FUNÇÃO E PROJETOS DE *SOFTWARE* FOCADOS NA ELABORAÇÃO DE CÁLCULOS CIENTÍFICOS

Conforme apresentado nos fundamentos teóricos expostos no capítulo II(dois) desta dissertação, a análise de pontos de função vem sendo largamente utilizada, em projetos de desenvolvimento de *software* em empresas e instituições governamentais. Apesar de atualmente estar estabelecida como um padrão na indústria de *software*, a análise de pontos de função não pode ser declarada uma unanimidade, tendo seu vasto campo de aplicação constantemente questionado pela comunidade científica.

Criada para mensurar a quantidade de *software* a ser implementada e ajudar assim na estimativa de esforço para a realização de projetos de *software*, a análise de pontos de função tem aplicação extremamente difundida e reconhecida pelo mercado de *software* para mensuração de sistemas de informações gerenciais. Tais sistemas constituem-se na maioria das aplicações comerciais existentes, sendo concentrados na realização de transações em base de dados e apresentação de resultados em uma camada

de interface amigável ao usuário da aplicação. Normalmente, sistemas dessa natureza possuem funcionalidades ligadas à execução de atividades como: consulta, alteração, exclusão e inserção de dados em um banco de dados, além de outras características ligadas à camada de apresentação como: elaboração de gráficos, relatórios, e apresentação de resultados.



**Figura IV.1 – Sistema de verificação de dados do aluno - exemplo de sistema de informação gerencial**

Apesar de ter uso consagrado para mensuração de sistemas de informações gerenciais, a análise de pontos de função não tem tido uma performance adequada no que diz respeito à mensuração de outros tipos de sistemas. Sua aplicação tem sido debatida pela comunidade científica em sistemas como:

- Sistemas de *real time*;
- Sistemas que envolvem processamento matemático pesado;
- Sistemas operacionais;
- Sistemas de comunicação.

De acordo com (ABRAN *et al.*, 1997), a análise de pontos de função considera apenas uma parcela dos tipos de aplicação existentes, tipicamente sistemas baseados em

arquivos, geralmente produzidos por bancos e organizações de varejo. Ainda segundo (ABRAN *et al.*, 1997), a análise de pontos de função consegue capturar muito bem as características funcionais de sistemas de informações gerenciais, mas deixam a desejar quando a medição é realizada sobre sistemas *real-time*. De acordo com (REDGATE, *et al.*, 2002) desenvolvedores de *software* sofrem o mesmo problema para realizar a medição de sistemas que tenham aplicação científica e realizem cálculos de elevada complexidade.

Sistemas *real-time*, assim como sistemas científicos de processamento matemático intenso, possuem em geral características muito distintas as de sistemas de informações gerenciais. Enquanto os últimos possuem foco em funcionalidades relacionadas a transações em arquivos de dados e camada de apresentação, os sistemas *real-time* e Científicos implementam algoritmos de alta complexidade com poucas entradas e saídas de dados. Como a base para a análise de pontos de função está nos elementos de entrada e saída, a contagem subestimada destes elementos pode levar a conclusão de que determinado procedimento, por mais complexo que possa ser possua poucos pontos de função. Este fato pode levar a uma medição equivocada do esforço necessário para a implementação do procedimento, podendo acarretar prejuízo para o projeto e para a organização responsável pela sua implementação.

Situações e fatos como os anteriormente apresentados levam à necessidade da adaptação da análise de pontos de função para implementação de projetos que envolvam cálculo científico apurado. Visando atender a essa demanda, este trabalho propõe a seguir, uma adaptação da análise de pontos de função, tomando como base a implementação da estação fotogramétrica educacional desenvolvida no projeto E-FOTO.

#### IV.3 DADOS UTILIZADOS NA PROPOSTA DE MÉTRICA

Tomando-se como base os fatos citados anteriormente sobre a questão da utilização da análise de pontos de função em projetos de *software* que possuam foco em processamento matemático, este trabalho vem a propor uma adaptação da métrica de pontos de função. Para realização desta adaptação tomou-se, como base histórica o esforço de implementação do módulo de retificação do projeto E-FOTO. Foi medido o esforço, em unidade de tempo de alocação dos programadores, para desenvolver cada um dos componentes de *software* integrantes do módulo de retificação.

Cabe destacar que a equipe responsável pela implementação do módulo de retificação do projeto E-FOTO é composta por pesquisadores com dedicação integral ao projeto. Os responsáveis pela implementação possuem formação em Engenharia de Sistemas e Computação, portanto, já possuíam conhecimento da linguagem de programação utilizada (C++). No entanto, vale ressaltar que, no que diz respeito aos conceitos de fotogrametria digital e cartografia foi necessário investir tempo considerável para o aprendizado de conceitos básicos indispensáveis para o entendimento e implementação do projeto.

Apresenta-se a seguir, tabela com o resultado da medição para implementação de componentes relacionados a cálculo com matrizes, por componente de *software* e por unidade de tempo.

**Tabela IV.1 – Esforço de implementação de componentes relacionados a matrizes**

<b>Componente</b>	<b>Tempo(Horas)</b>
Inversa	17,5
Produto	8
Identidade	8
Transposta	5,16
Soma	5,16
Subtração	5,16

O esforço relacionado à implementação de operações transformação também foi aferido, sendo apresentado de acordo com a tabela a seguir:

**Tabela IV.2 – Esforço de implementação de componentes de transformações**

<b>Componente</b>	<b>Tempo(Horas)</b>
Afim Geral	22,5
Afim Isogonal	9,5
Linear Direta	9,5
Projetiva	9,5

#### IV.4 UMA PROPOSTA DE MÉTRICA PARA MEDIÇÃO DE *SOFTWARE* FOCADO NA IMPLEMENTAÇÃO DE PROJETOS DE SOFTWARE NA ÁREA DA FOTOGRAMETRIA DIGITAL

A base teórica exposta neste trabalho deixa clara a necessidade de criação de uma alternativa de métrica para mensurar sistemas de informação com foco em processamento matemático. Tomando como base este objetivo propõe-se a seguir as seguintes modificações na análise de pontos de função:

##### IV.4.1 CRIAÇÃO DE FUNÇÕES TIPO CÁLCULO MATEMÁTICO

A proposta considera a criação de um novo grupo de funções, denominado funções tipo cálculo matemático, para mensurar especificamente o esforço para implementação de funcionalidades que dependam exclusivamente de processamento matemático. Tomando-se como base o escopo de aplicação do projeto E-FOTO, propõe-se inicialmente a criação das seguintes classes de funções:

- Matriz;
- Transformações.

A partir desta classificação, define-se que:

##### a) Funções Matriz:

- Constituem-se de operações matemáticas sobre matrizes;
- São mantidas dentro da fronteira da aplicação;
- São utilizadas por meio de processos ou sub-processos.

##### b) Transformações:

- Constituem-se de operações matemáticas relativas a transformações sobre matrizes;
- São mantidas dentro da fronteira da aplicação;
- São utilizadas por meio de processos ou sub-processos.

Como exemplos de funções do tipo Matrizes podem-se considerar:



- Uma operação de soma de matrizes;
- Uma operação de produto de matrizes;
- Uma operação de transposição de matriz.

Como exemplos de funções do tipo Transformações, podem-se considerar:

- Operação de transformação Afim Geral;
- Operação de transformação Afim Isogonal;
- Operação de transformação Projetiva.

#### IV.4.2 REGRAS DE CONTAGEM PARA AS FUNÇÕES PROPOSTAS

##### a) Funções Matriz:

- Deverá ser contada a quantidade de tipos de operações realizada sobre matrizes;
- Deverão ser consideradas para efeito de contagem apenas as matrizes que estejam dentro do escopo da aplicação.

##### b) Transformações:

- Deverá ser contada a quantidade de tipos de transformações a serem realizadas sobre as matrizes que compõem as imagens;
- Deverão ser consideradas para efeito de contagem apenas as operações de transformação que estejam dentro do escopo da aplicação.

#### IV.4.3 TABELAS DE COMPLEXIDADE

Levando-se em conta o esforço mensurado no projeto E-FOTO para a implementação das funções anteriormente citadas, pode-se agrupá-las com base nos seguintes níveis de complexidade:

a) Funções Matriz

**Tabela IV.1 – Complexidade de Funções do tipo Matriz**

Complexidade	Matriz
Alta	Inversa
Média	Produto, Identidade
Baixa	Transposta, Soma e subtração

b) Transformações

**Tabela IV.2 – Complexidade de Funções do tipo Transformações**

Complexidade	Transformação
Alta	Afim Geral
Baixa	Afim, Isogonal, Linear Direta e Projetiva

As complexidades foram medidas apenas sobre as funções estudadas. As funções foram agrupadas de acordo com o esforço dispensado para sua implementação, mensurado junto à equipe do projeto E-FOTO.

#### IV.4.4 DETERMINAÇÃO DA CONTRIBUIÇÃO

Após a determinação da complexidade das funções Matrizes e Transformações, deve-se determinar a contribuição dessas funções em termos de pontos de função. Essa contribuição deve ser obtida por meio das seguintes tabelas:

a) Para Funções Matriz:

**Tabela IV.3 – Contribuição de Funções do tipo Matriz**

Complexidade	Pontos de Função
Alta	3
Média	2
Baixa	1

b) Para Transformações

**Tabela IV.4 – Contribuição de Funções do tipo Transformações**

<b>Complexidade</b>	<b>Pontos de Função</b>
Alta	4
Baixa	2

As tabelas de complexidade anteriormente descritas foram obtidas através da análise de registro de esforço para implementação do projeto E-FOTO. Tomando-se como base a mediana da taxa de produtividade para implementação de 1 (um) ponto de função em projetos de *software* desenvolvidos por instituições governamentais foi possível propor patamares de complexidade de implementação para as funções estudadas e descritas anteriormente. A taxa de produtividade utilizada neste trabalho pode ser encontrada em (HILL, 2004) como uma mediana para projetos de *software* desenvolvidos em plataforma baixa (computadores padrão IBM-PC) para instituições governamentais. A taxa aferida em (HILL, 2004) para projetos desta natureza foi de 5,3 (cinco vírgula três) horas por ponto de função entregue.

Através da utilização da taxa de produtividade obtida em (HILL, 2004), pode-se calcular a contribuição, expressa em número de pontos de função, de cada tipo de função proposta. Como exemplo, é possível determinar que transformações de complexidade Alta contribuem com quatro pontos de função. Para chegar a essa conclusão, foi utilizada a seguinte fórmula:

$$\text{CPF} = \text{EGI} / \text{PDT}$$

IV.1

onde:

CPF - Contribuição em Pontos de Função;

EGI - Esforço Gasto para Implementação, obtido através de consulta a tabela IV.1;

PDT - Produtividade obtida em (HILL, 2004).

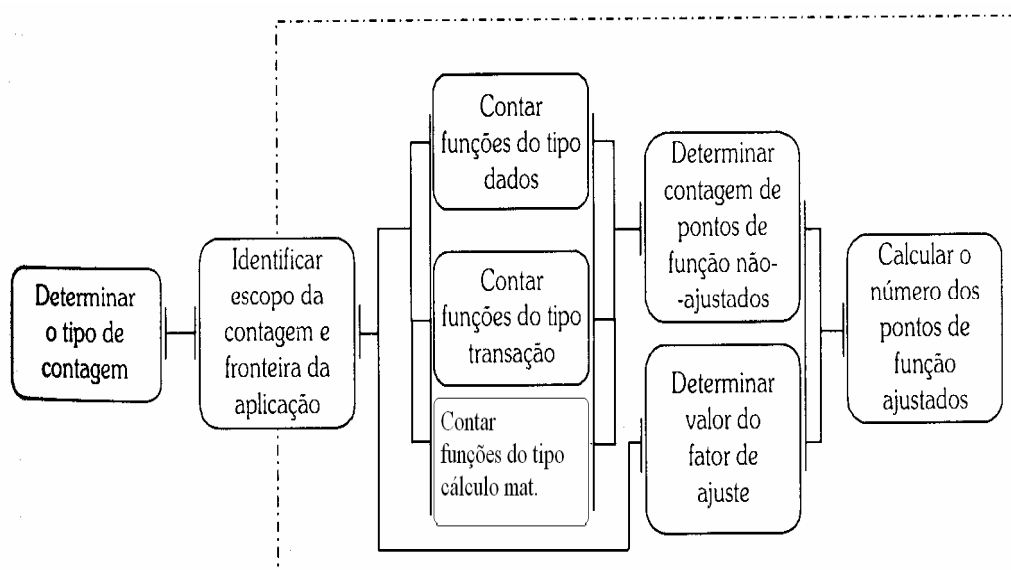
Através da utilização da fórmula anteriormente proposta tem-se para a Transformação Afim Geral:

$$\text{CPF} = 22,26 \text{ horas} / 5,3 \text{ horas} = 4,2 \text{ PF} \approx 4 \text{ PF}$$

Adotando o procedimento anteriormente citado repetidas vezes, pode-se determinar as tabelas de contribuição para os tipos de funções propostos.

#### IV.4.5 O PROCESSO DE CONTAGEM

A criação das funções Tipo cálculo matemático proposta no item IV.4.1 deste documento, não altera de forma significativa o processo de contagem de pontos de função. A figura a seguir apresenta o novo esquema considerando a inclusão da etapa de contagem das funções tipo cálculo matemático.



**Figura IV.2 – Novo Processo de Contagem**

O processo de contagem considera as mesmas etapas anteriormente existentes. Em paralelo às etapas *contar funções do tipo dados* e *contar funções do tipo transação*, propõe-se a inclusão de uma nova etapa denominada *contar funções tipo cálculo matemático*. Durante essa etapa são contadas as funções descritas no item IV.4.1 deste documento, considerando-se os procedimentos necessários para a determinação de sua complexidade, e de sua contribuição, descritos nos itens IV.4.2, IV.4.3 e IV.4.4.

O novo processo de contagem propõe ainda uma alteração com relação à utilização do elemento Saída Externa, existente no processo do IFPUG. A proposta deste trabalho é substituir o elemento SE pelas funções de cálculo matemático, sempre que ficar caracterizada a implementação de funções desta natureza. Os demais elementos constantes do processo do IFPUG permanecem inalterados, sendo contados de acordo com as normas sugeridas por esta organização.

Uma vez realizada a contagem das funções tipo cálculo matemático, deve-se manter os procedimentos citados na seção II.3.1 para cálculo do fator de ajuste e número de pontos de função ajustados. Por sua vez, a determinação da contagem dos pontos de função não ajustados requer que seja considerado em seu cômputo, o número de pontos de função obtidos na contagem das funções tipo cálculo matemático. Dessa forma, o número de pontos de função não ajustados passa a ser calculado através do somatório da quantidade de pontos de função obtida nas contagens das funções tipo dados, tipo transação e tipo cálculo matemático.

#### IV.5 COMENTÁRIOS

Um exemplo sobre a aplicação da proposta descrita nesta seção pode ser obtido na seção a seguir. No transcorrer da próxima seção, a contagem de pontos de função obtida pela proposta fornecida por este trabalho é comparada com a contagem sugerida pelo IFPUG. Os resultados de ambas as contagens são comparados com o resultado real medido ao final do projeto de desenvolvimento.

## **CAPÍTULO V – APLICAÇÃO E APRESENTAÇÃO DE RESULTADOS**

### **V.1 INTRODUÇÃO**

O capítulo anterior introduziu a questão sobre a dificuldade de se mensurar sistemas de informação que possuam foco em operações matemáticas complexas, através dos procedimentos de contagem de pontos de função sugeridos pelo IFPUG. Como resposta a esta questão, o capítulo propõe uma extensão da métrica para mensurar de forma mais precisa o esforço de implementação de sistemas de informação com foco em operações matemáticas complexas.

Visando comprovar a viabilidade da proposta fornecida no capítulo anterior, o presente capítulo apresenta dois estudos de caso baseados em módulos de *software* desenvolvidos para o projeto E-FOTO. O primeiro estudo de caso implementa a proposta de métrica sobre o módulo de retificação do E-FOTO, enquanto que o segundo, sobre o módulo de orientação interior. Em ambos os casos, o processo de contagem também é realizado de acordo com os procedimentos sugeridos pelo IFPUG. Ao final da contagem por ambas as propostas (IFPUG e deste trabalho) os resultados são comparados com o tamanho real do esforço mensurado após a conclusão do projeto.

De forma a verificar a influência do modelo de custo nos resultados obtidos, aplica-se ainda o modelo de custos COCOMO II sobre o resultado da contagem de acordo com o padrão IFPUG. Os resultados da aplicação dos modelos de custo Homem x Hora e COCOMO II são analisados na seção de apresentação de resultados.

### **V.2 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE RETIFICAÇÃO DE ACORDO COM A PROPOSTA DO IFPUG**

Analisando-se o módulo de retificação implementado no projeto E-FOTO, descrito de forma sucinta no item III.5.6 deste documento e de forma mais detalhada no manual do usuário, pode-se, tomando como base a teoria da métrica de pontos de função proposta pelo IFPUG, aferir a seguinte contagem de pontos de função:

- a) Tipo de contagem: Projeto de Desenvolvimento;
- b) Escopo da Contagem: Funcionalidade de retificação de imagens;

- c) Fronteira da Aplicação: Funções desenvolvidas para implementar o módulo de retificação, de acordo com o ponto de vista funcional (visão do usuário);
- d) Funções do Tipo Dado:

**Tabela V.1 – Contagem de funções tipo dados –Retificação - IFPUG**

Descrição	Tipo	TD	TR	Complexidade	P.F.
Arquivo de resultados	ALI	4	1	Baixa	7
<b>Total</b>	-	-	-	-	<b>7</b>

- e) Funções do tipo Transação:

**Tabela V.2 - Contagem de funções tipo transação – Retificação - IFPUG**

Descrição	Tipo	TD	AR	Complexidade	P.F.
Interface de entrada de dados	EE	8	1	Baixa	3
Resultado da Retificação da imagem	SE	5	1	Baixa	4
<b>Total</b>	-	-	-	-	<b>7</b>

Desta forma, considerando-se apenas a utilização de pontos de função não ajustados, pode-se aferir, com base no processo de contagem proposto pelo IFPUG, que o módulo de retificação possui 14 (quatorze) Pontos de Função não ajustados.

### V.3 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE RETIFICAÇÃO COM BASE NA PROPOSTA DESTA DISSERTAÇÃO

Analisando-se o módulo de retificação implementado no projeto E-FOTO com base na teoria da métrica de pontos de função proposta neste trabalho, pode-se aferir a seguinte contagem de pontos de função:

- a) Tipo de contagem: Projeto de Desenvolvimento;
- b) Escopo da Contagem: Funcionalidade de retificação de imagens;
- c) Fronteira da Aplicação: Funções desenvolvidas para implementar o módulo de retificação, de acordo com o ponto de vista funcional (visão do usuário);

d) Funções do Tipo Dado:

**Tabela V.3 – Contagem de funções tipo dados – Retificação - Proposta**

Descrição	Tipo	TD	TR	Complexidade	P.F.
Arquivo de resultados	ALI	4	1	Baixa	7
<b>Total</b>	-	-	-	-	<b>7</b>

e) Funções do tipo Transação:

**Tabela V.4 - Contagem de funções tipo transação – Retificação**

Descrição	Tipo	Operações	TD	AR	P.F.
Interface de entrada de dados	EE	-	8	1	3
<b>Total</b>	-	-	-	-	<b>3</b>

f) Funções tipo cálculo matemático:

**Tabela V.5 - Contagem de funções tipo cálculo matemático – Retificação**

Descrição	Operação	Complexidade	Contribuição (P.F.)
Retificação da imagem	Inversa	Alta	3
	Soma	Baixa	1
	Identidade	Média	2
	Multiplicação	Média	2
	Transposta	Baixa	1
	Afim geral	Alta	4
	<b>Total</b>	-	<b>13</b>

A tabela V.5 explicita na coluna operações, as funções do Tipo Cálculo matemático integrantes do módulo de retificação, contemplando: 6 (seis) funções, sendo 1 (uma) transformação e 5 (cinco) operações com matrizes. Considerando a utilização da transformação Afim Geral (Pior caso), e tomando por base a complexidade e contribuição descritas nas seções IV.3.2, IV.3.3 e IV.3.4 deste trabalho, pode-se aferir as contribuições citadas na tabela V.5.



Desta forma, considerando-se apenas a utilização de pontos de função não ajustados, pode-se aferir, com base no processo de contagem proposto por este trabalho, que o módulo de retificação possui 23 (vinte e três) Pontos de Função.

#### V.4 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE ORIENTAÇÃO INTERIOR DE ACORDO COM A PROPOSTA DO IFPUG

Analisando-se o módulo de orientação interior implementado no projeto E-FOTO, descrito de forma sucinta no item III.5.3 deste documento, pode-se, tomando como base a teoria da métrica de pontos de função proposta pelo IFPUG, aferir a seguinte contagem de pontos de função:

- a) Tipo de contagem: Projeto de Desenvolvimento;
- b) Escopo da Contagem: Funcionalidade de Orientação Interior de imagens;
- c) Fronteira da Aplicação: Funções desenvolvidas para implementar o módulo de Orientação Interior, de acordo com o ponto de vista funcional(visão do usuário);
- d) Funções do Tipo Dado:

**Tabela V.6 – Contagem de funções tipo dados – O.I. - IFPUG**

Descrição	Tipo	TD	TR	Complexidade	P.F.
Arquivo de resultados	ALI	3	1	Baixa	7
<b>Total</b>	-	-	-	-	<b>7</b>

- e) Funções do tipo Transação:

**Tabela V.7 - contagem de funções tipo transação – O.I. - IFPUG**

Descrição	Tipo	TD	AR	Complexidade	P.F.
Interface de entrada de dados	EE	3	1	Baixa	3
Resultado da Orientação Interior	SE	2	1	Baixa	4
<b>Total</b>	-	-	-	-	<b>7</b>

Desta forma, considerando-se apenas a utilização de pontos de função não ajustados, pode-se aferir, com base no processo de contagem proposto pelo IFPUG, que o módulo de Orientação interior possui 14(quatorze) Pontos de Função não ajustados.

## V.5 ESTUDO DE CASO – MEDIÇÃO DO MÓDULO DE ORIENTAÇÃO INTERIOR COM BASE NA PROPOSTA DESTA DISSERTAÇÃO

Analisando-se o módulo de orientação interior implementado no projeto E-FOTO, com base na teoria da métrica de pontos de função proposta neste trabalho pode-se aferir a seguinte contagem de pontos de função:

- a) Tipo de contagem: Projeto de Desenvolvimento;
- b) Escopo da Contagem: Funcionalidade de Orientação interior de imagens;
- c) Fronteira da Aplicação: Funções desenvolvidas para implementar o módulo de retificação, de acordo com o ponto de vista funcional (visão do usuário);
- d) Funções do Tipo Dado:

**Tabela V.8 – Contagem de funções tipo dados - O. I. – Proposta Dissertação**

Descrição	Tipo	TD	TR	Complexidade	P.F.
Arquivo de resultados	ALI	3	1	Baixa	7
<b>Total</b>	-	-	-	-	<b>7</b>

- e) Funções do tipo Transação:

**Tabela V.9 – Contagem de funções tipo transação - O. I. – Proposta Dissertação**

Descrição	Tipo	Operações	TD	AR	Contribuição (PF)
Interface de entrada de dados	EE	-	3	1	3
<b>Total</b>	-		-	-	<b>3</b>

f) Funções tipo cálculo matemático:

**Tabela V.10 - Contagem de funções tipo cálculo matemático – O. I. – proposta dissertação**

Descrição	Operação	Complexidade	Contribuição (P.F.)
Retificação da imagem	Inversa	Alta	3
	Soma	Baixa	1
	Identidade	Média	2
	Multiplificação	Média	2
	Transposta	Baixa	1
	Afim geral	Alta	4
	<b>Total</b>	-	<b>13</b>

Onde tem-se as seguintes funções do Tipo Cálculo matemático: 6 (seis) funções, sendo 1 (uma) transformação e 5 (cinco) operações com matrizes.

Desta forma, considerando-se apenas a utilização de pontos de função não ajustados, pode-se aferir, com base no processo de contagem proposto por este trabalho, que o módulo de Orientação interior possui 23(vinte e três) Pontos de Função não ajustados.

## V.6 APLICAÇÃO DO MODELO DE CUSTO COCOMO II SOBRE A PROPOSTA DO IFPUG

Visando realizar uma análise da influência do modelo de custo sobre os resultados obtidos na aplicação das propostas do IFUG, e desta dissertação, sobre os módulos de orientação interior e retificação, faz-se necessária a aplicação de um modelo de custo diferente da relação homem x hora utilizada para obter-se a métrica proposta neste trabalho. O modelo de custo escolhido para realizar a análise foi o modelo COCOMO II. Este modelo foi escolhido devido aos seguintes fatores: Larga utilização no mercado de software, facilidade de acesso a material didático sobre o modelo(uma vez que o mesmo é gratuito) e as características técnicas do modelo serem aderentes a métrica utilizada.

A aplicação do COCOMO II foi implementada sobre as medições realizadas segundo o modelo de Pontos de Função do IFPUG. Utilizando-se um *software*

disponibilizado pela *University of Southern California* (USC), denominado *USC COCOMO II.1999.0*, foram calculados os esforços de desenvolvimento dos módulos de Retificação e Orientação Interior. Esta ferramenta realiza cálculos com base em uma unidade de medida (Pontos de função ou linhas de código) e parâmetros de calibração, fornecendo como resultados, dentre outros resultados, o esforço de implementação, o tempo de desenvolvimento e a equipe envolvida.

Para o experimento em questão, a ferramenta da USC foi calibrada com dados de esforço e tempo de implementação dos módulos de Orientação exterior e Estereoplotter, que possuem valores de pontos de função não ajustados (segundo o modelo do IFPUG) similares aos dos módulos mensurados. Os valores utilizados como parâmetros de calibração encontram-se explicitados na tabela a seguir.

**Tabela V.11 – Valores de calibração da ferramenta *USC COCOMO II.1999.0***

<b>Módulo</b>	<b>Effort</b>	<b>Schedule</b>
Orientação Exterior	0,79	0,79
Estereoplotter	1	1

A tabela V.11 explicita em sua coluna *Effort*, o esforço, em meses, para a implementação do módulo. O valor explicitado para este parâmetro constitui-se no somatório dos meses onde os recursos humanos do projeto estão alocados para realizar sua implementação (LOPEZ, 2005). A coluna *Schedule*, por sua vez, informa o tempo de desenvolvimento, em meses, para a implementação do módulo. Os valores utilizados para a calibração foram encontrados através da divisão do tempo real de desenvolvimento aferido para cada módulo com a quantidade horas de trabalho mensal fornecidas pelo modelo COCOMO II, no caso estabelecidas pelo modelo em 152 (cento e cinquenta e duas) horas (USC, 1999). Para ambos os módulos considerou-se a equipe base de 1 (um) integrante alocado integralmente ao projeto durante o período de desenvolvimento do projeto. A tabela a seguir explicita o tempo real para implementação dos módulos de orientação exterior e estereoplotter.

**Tabela V.12 – Horas reais utilizadas para implementação de módulos do e-foto**

<b>Módulo</b>	<b>Tempo Real (em Horas)</b>
Orientação Exterior	120
Estereoplotter	152

O modelo COCOMO II conforme fundamentação teórica exposta no capítulo II desta dissertação possui ainda duas classes de parâmetros, denominados fatores de escala e multiplicadores de esforço. Para a implementação experimento citado nesta dissertação, considerou-se os valores nominais dos referidos parâmetros. Foram utilizados os valores padrão, previamente estabelecidos pelo modelo COCOMO II.

Uma vez calibrado o modelo, utilizando-se os parâmetros com valores nominais, informou-se a quantidade de pontos de função não ajustados para os módulos retificação e orientação interior. O resultado da aplicação de tais parâmetros pode ser observado na seção a seguir, de apresentação de resultados.

## V.7 APRESENTAÇÃO DE RESULTADOS

Considerando-se a aplicação da métrica de pontos de função nos modelos do IFPUG, da proposta de dissertação, e ainda o modelo de custo COCOMO II e a relação Homem x hora, apresenta-se nesta seção os resultados obtidos nos experimentos realizados. Os resultados são apresentados em formato de tabela contendo os tempos de implementação dos módulos estudados segundo o modelo de custo e a relação homem x hora contemplados nesta dissertação.

O quadro de resultados apresenta ainda as horas reais de implementação. O esforço real foi obtido junto a equipe de implementação do projeto E-FOTO, formada, por pesquisadores com dedicação parcial ao projeto. Os pesquisadores envolvidos possuem formação em Engenharia de sistemas e computação e já conheciam a linguagem de desenvolvimento C++ utilizada na implementação do Projeto. O maior desafio para a equipe foi aprender os conceitos e cartografia e fotogrametria digital necessários para a implementação dos módulos do E-FOTO.

Os módulos aferidos individualmente foram considerados, para efeito de medição, como novos projetos de *software*, uma vez que nada ou muito pouco foi

aproveitado de um módulo para outro. Apenas um dos membros da equipe participou da implementação de todos os módulos estudados.

A definição de quais módulos aferir foi baseada em dois fatores:

- 1) O período de implementação dos mesmos em relação ao cronograma do projeto E-FOTO;
- 2) A similaridade sobre as operações de cálculo realizadas nos módulos.

O fato dos módulos terem sido implementados no decorrer do desenvolvimento desta dissertação possibilitou que os mesmos pudessem ser mensurados e considerados para este trabalho. A similaridade sobre as operações de cálculo realizadas em ambos os projetos possibilitou que fossem comparados para verificar a acuracidade da proposta de métrica.

Sobre o escopo da contagem, cabe ressaltar que os tempos de esforço reais aferidos e descritos na tabela anterior, dizem respeito à implementação de todas as funcionalidades existentes em cada módulo estudado. A contagem detalhada dos módulos de retificação e orientação interior pode ser encontrada nas seções V.2 a V.5 desta proposta de dissertação.

Desta forma, com base nos detalhes anteriormente expostos apresenta-se a seguir o quadro de resultados de esforço para os módulos do projeto E-FOTO estudados.

**Tabela V.13 – Resultados de Esforço e tempo real de implementação.**

<b>Módulo</b>	<b>Tempo IFPUG (H)</b>	<b>Tempo COCOMO II IFPUG (H)</b>	<b>Tempo Proposta de dissertação (H)</b>	<b>Tempo COCOMO II Proposta (H)</b>	<b>Tempo Real (H)</b>
Retificação	$14 * 5,3 = 74,2$	$0,9 * 152 = 136,8$	$23 * 5,3 = 121,9$	$1,1 * 152 * 1,5 = 250,8$	105
Orientação Interior	$14 * 5,3 = 74,2$	$0,9 * 152 = 136,8$	$23 * 5,3 = 121,9$	$1,1 * 152 * 1,5 = 250,8$	108

A tabela V.13 apresenta como primeira coluna, da esquerda para direita, o nome do módulo estudado. A coluna seguinte contempla o esforço de implementação em horas aferido com base no modelo de pontos de função do IFPUG e a relação Homem x Hora. Utilizando-se a produtividade de 5,3 (cinco vírgula três) Horas por pontos de

função obtida em (HILL, 2004), multiplicando-se pela quantidade de pontos de função não ajustados obtidos para o módulo em questão pelo modelo do IFPUG, pode-se chegar ao resultado exposto.

A terceira coluna da tabela V.13 traz o tempo obtido através da aplicação do modelo de custo COCOMO II sobre a quantidade de pontos de função não ajustados do modelo IFPUG. A aplicação do COCOMO II realizada com a utilização de uma ferramenta de software específica traz como resultado o tempo de implementação de 0,9(zero vírgula nove) meses. Considerando a equipe de desenvolvimento do módulo composta por um único profissional, o esforço em horas pode ser determinado multiplicando-se 0,9 por 152(total de horas trabalhadas em um mês segundo modelo COCOMO II).

A quarta coluna corresponde ao esforço, em horas, mensurado de acordo com a proposta de métrica constante desta dissertação. A proposta está associada ao fator de produtividade de 5,3(cinco vírgula três) obtido em (HILL, 2004). Multiplicando-se o fator de produtividade pela quantidade de pontos de função não ajustados medidos, obtém-se o resultado.

A quinta coluna informa o esforço, em horas, mensurado de acordo com o modelo COCOMO II aplicado sobre a quantidade de pontos de função aferidos pela proposta de métrica descrita neste trabalho. O esforço é obtido através da multiplicação dos fatores correspondentes respectivamente a: o tempo de implementação do projeto em meses (1,1), a quantidade de horas utilizada pelo modelo COCOMO II em um mês de trabalho (152) e a quantidade de pessoas envolvidas no projeto (1,5).

A sexta e última coluna, corresponde ao tempo real de implementação aferido junto a equipe de desenvolvimento. O valor apresentado constitui-se do somatório de horas dos integrantes da equipe utilizadas para a implementação de cada um dos módulos.

Cabe ressaltar que no que diz respeito ao processo de contagem, este trabalho considerou apenas a contagem do ponto de função não ajustado. Isto se deve ao fato da proposta de métrica aqui apresentada não incluir nenhuma alteração em relação à formulação e contagem dos pontos de função ajustados.

## V.8 COMENTÁRIOS

Neste capítulo, a proposta de métrica foi aplicada e seus resultados apresentados de forma comparativa, inclusive considerando a aplicação da relação de custo Homem x Hora e COCOMO II. O capítulo a seguir fornece as conclusões obtidas com base nos resultados alcançados.



## CAPÍTULO VI – CONCLUSÕES

Conforme apresentado nas seções anteriores deste trabalho, a análise de pontos de função vem sendo largamente utilizada, em projetos de desenvolvimento de *software* em empresas e instituições governamentais. Apesar de atualmente estar estabelecida como um padrão na indústria de *software*, a análise de pontos de função não apresenta um desempenho satisfatório para mensuração de esforço de implementação de sistemas com foco em processamento matemático complexo. Para atender a sistemas dessa natureza, cada vez mais se busca especializar a análise de pontos de função, ou utilizar métodos alternativos para mensurar o esforço de projetos de desenvolvimento de *software*.

Com base nos resultados expostos neste trabalho, pode-se averiguar que a aplicação da análise de pontos de função segundo os critérios do IFPUG, com um modelo de custo baseado em homem x hora e uma produtividade pré-definida por tecnologia, obteve menor precisão em relação à proposta da dissertação. Analisando-se a tabela V.13 (comparando-se esforço real e mensurado), pode-se verificar que o esforço mensurado segundo os critérios do IFPUG para o escopo considerado ficou um pouco aquém do esforço real despendido pela equipe de desenvolvimento do E-FOTO. Uma verificação aos valores obtidos na tabela V.13 mostra que, para o módulo de retificação, o esforço mensurado segundo os critérios do IFPUG ficou aproximadamente 30% abaixo esforço real para implementação do módulo. Considerando-se o módulo de orientação interior, a métrica mantida pelo IFPUG ficou 32% abaixo do esforço real do projeto.

A proposta de especialização da análise de pontos de função descrita por este trabalho, por sua vez, ao ser aplicada ao projeto, obteve resultados mais próximos ao esforço real aferido. A aplicação da proposta de métrica sobre o módulo de retificação obteve um valor acima do esforço real na ordem de 14%. Um valor mais próximo ao esforço real era esperado para este módulo, uma vez que o mesmo serviu de base para a formulação da proposta de especialização da métrica. No entanto, a aplicação da proposta de métrica descrita neste trabalho sobre o módulo de orientação interior, obteve um desvio de 12 % em relação ao valor real aferido.

Embora aparentemente a proposta de métrica apresentada neste trabalho tenha tido um resultado mais preciso em comparação com o modelo do IFPUG, faz-se necessário explicitar que sua aplicação foi realizada sobre uma base de projetos

extremamente pequena. Este fato impede se conclua de maneira incisiva que a proposta apresente maior eficácia em relação ao modelo do IFPUG. Deve-se considerar também a possibilidade de sua performance estar diretamente relacionada ao modelo de custo utilizado. Este ponto pode ser aferido a partir da comparação entre os resultados da proposta de métrica apresentada e o modelo de custo COCOMO II. Este último, utilizando-se dos pontos de função não ajustados pelo modelo IFPUG, obteve um desempenho menos preciso, porém muito próximo ao da métrica proposta. Quando aplicado sobre os módulos de retificação e orientação interior, o COCOMO II apresentou um desvio de aproximadamente 30% em relação ao tempo real de implementação do primeiro módulo e de 26 % em relação ao segundo. Se comparado ao resultado da proposta de métrica, este desvio é de apenas 12%. A proximidade entre os resultados da proposta de métrica e do modelo COCOMO II pode ser explicado, pelo menos em parte, pela utilização de dois outros módulos do E-FOTO muito parecidos com os de retificação e orientação interior em termos de funcionalidades e produtividade de desenvolvimento, para calibração do modelo.

Outro dado aferido foi oriundo da utilização do COCOMO II sobre os módulos de retificação e orientação interior, segundo a quantidade de pontos de função mensurada com base na proposta desta dissertação. O resultado aferido ficou distante do valor real do software. Este ponto pode ser explicado pelo fato da calibração do COCOMO II ter sido realizada com base na contagem de pontos de função segundo o modelo IFPUG, o que impossibilita classificar o resultado como conclusivo. A adoção de uma produtividade baseada no modelo do IFPUG também contribuiu para o resultado aferido, uma vez que a ferramenta de software da USC converte a quantidade de pontos de função em linhas de código através de uma taxa de produtividade aferida para os critérios do IFPUG. O cálculo da fórmula proposta pelo COCOMO II para determinação de esforço somente é calculada após a determinação da quantidade de linhas de código produzidas.

Através dos resultados apresentados neste trabalho pode-se concluir de maneira sucinta que:

- a) A amostra do experimento não é representativa para qualquer conclusão definitiva. Apesar dos dados coletados serem representativos para o teste do projeto estudado, se faz necessário verificar o comportamento da proposta de métrica em uma amostra maior e mais significativa de projetos.

- b) Os resultados obtidos obtiveram uma amplitude muito elevada. Os tempos de implementação em horas (Tabela V.13) variaram de 74,2 (setenta vírgula dois) a 250,8 (duzentas e cinquenta vírgula oito) horas versus um "*tempo real*" da amostra de 105(cento e cinco) e 108(cento e oito) horas.
- c) Os resultados obtidos permitem algumas interpretações distintas. Pode-se comparar cada um dos resultados das métricas com o tempo real ou pode-se compará-los dois a dois dificultando a determinação de qual seria o resultado ideal.

Segundo (BOEHM, 1981), a Engenharia de Software ainda é imatura no que tange a modelos de custo, se comparada a outras áreas mais tradicionais da Engenharia. Considerando-se esta afirmação, apesar dos resultados obtidos não serem conclusivos, a proposta apresentada permanece como contribuição significativa para a área de estudos das métricas de *software*. Situações onde se faz necessário o dimensionamento do custo de um projeto com base em uma taxa de produtividade pré-definida para determinada tecnologia ou plataforma são cada vez mais comuns no mercado. Para cenários como esse, a adaptação da métrica de pontos de função para mensurar funcionalidades que envolvam implementação de funções de cálculo matemático complexo pode vir a alcançar resultados mais próximos ao esforço real, e fazer com que determinada instituição ou empresa economize uma quantia significativa de recursos.

No caso específico do projeto E-FOTO, considerando-se a utilização do modelo homem x hora com taxa de produtividade pré-definida, a diferença obtida entre o dimensionamento de esforço com base no modelo do IFPUG e o da proposta desta dissertação poderia representar um atraso significativo no cronograma do projeto, assim como, um incremento considerável no custo do projeto, pois poderia acarretar a necessidade de alocar mais pesquisadores, ou prolongar a bolsa paga aos atuais integrantes da equipe por um período superior ao inicialmente estipulado. Este prejuízo seria tanto maior quanto fosse o tamanho do escopo das aplicações com foco em cálculo matemático complexo.

Desta forma, com base nos resultados obtidos e apresentados na seção V.7 deste trabalho sugere-se, para mensurar de forma mais precisa o esforço de implementação da estação fotogramétrica digital (objetivo do Projeto E-FOTO) com base na análise de pontos de função e um modelo de custo homem x hora com taxa de produtividade pré-

definida pela tecnologia, a implementação de uma especialização da métrica de pontos de função. Tal especialização, deve considerar a complexidade da implementação das operações matemáticas inerentes a projetos da área da fotogrametria, assim como, propor uma solução para mensurar de forma correta o esforço de implementação de tais operações.

Por fim, como sugestão para futuros trabalhos, propõe-se a extensão da especialização da métrica apresentada para contemplar mais operações matemáticas que porventura venham a ser utilizadas em futuros módulos do projeto E-FOTO ou em outro *software* científico. Essa extensão deve ser testada e ajustada em uma base extensa de projetos de *software*, de forma a comprovar sua eficácia.

## REFERÊNCIAS

**ABRAN**, A.; **DESHARNAIS**, J. L.; **MAYA**, M.; **ST-PIERRE**, D., 1997, Adapting Function Points to Real Time Software, 1997 IFPUG Fall Conference, Montréal, Canada.

**ABRAN**, A.; **ROBBILARD**, P. N., 1991, Identification Of the Structural Weaknesses of Function Point Metrics, 3<sup>rd</sup> Anual Oregon Workshop on Software Metrics, Portland – Oregon, EUA.

**ASMA**, 1999, Padrão ISO para medição funcional de tamanho, [http://www.bfpug.com.br/Artigos/iso\\_fsm.htm](http://www.bfpug.com.br/Artigos/iso_fsm.htm), acesso em: 15.01.2007.

**ANDRADE**, J. B., 1998, Fotogrametria, Editora SBEE, Curitiba – PR, Brasil.

**BASTOS**, D. L.; **BRITO**, J. L. N. e S.; **BERNARDO FILHO**, O. ; **RIBEIRO**, J. A.; **SILVA**, R. P., 2005, E-Foto: Módulo de Retificação de Imagens em uma Estação Fotogramétrica Digital. In: XI Simpósio Brasileiro de Sensoriamento Remoto, 2005, Goiânia.

**BFPUG**, Perguntas e dúvidas freqüentes, <http://www.bfpug.com.br/FAQ.htm>, acesso em: 05.02.2007.

**BOEHM**, B., 1981, Software Engineering Economics, Prentice Hall, 1<sup>a</sup> Edição, Upper Saddle River, NJ, EUA.

**BRITO**, J. L. N. e S.; **COELHO F.**, L. C. T., 2002, Fotogrametria Digital 1<sup>a</sup> Edição Instituto Militar de Engenharia Rio de Janeiro, Brasil.

**BRITO**, J. L. N. e S.; **COELHO F.**, L. C. T., 2002, Projeto E-Foto: Uma Estação Fotogramétrica Digital Educacional. In: III Encontro de Iniciação Científica do Instituto Militar de Engenharia, 2002, Rio de Janeiro.

**BRITO, J. L. N. e S.; COELHO F., L. C. T., 2002, The E-Foto Project: An Educational Digital Photogrammetric Workstation. In: ISPRS Commission VI Mid Term Symposium on New approaches for Education and Communication, 2002, São José dos Campos.**

**BRITO, J. L. N. e S.; COELHO F., L. C. T., 2003, Projeto E-Foto: Uma Estação Fotogramétrica Digital Educacional . In: XI Simpósio Brasileiro de Sensoriamento Remoto, 2003, Belo Horizonte.**

**BRITO, J. L. N. e S.; COELHO F., L. C. T., 2003, Projeto E-Foto: Uma Estação Fotogramétrica Digital Educacional. Portal de Educação do Exército.**

**BRITO, J. L. N. e S.; COELHO F., L. C. T.; BERNARDO FILHO, O. , 2004, Projeto e-foto: uma estação fotogramétrica digital educacional. In: Geolive Rio 2004, 2004, Rio de Janeiro.**

**BRITO, J. L. N. e S.; COELHO F., L. C. T., 2005, The E-FOTO Project - A digital photogrammetric softcopy kit for educational purposes. In: VI Barcelona Geomatic Week, 2005, Barcelona. 6th Geomatic Week Proceedings.**

**BRITO, J. L. N. e S.; SOUZA FILHO, L. A., 2005, Mapeamento Fotogramétrico Digital. In: XXII Congresso Brasileiro de Cartografia, 2005, Macaé.**

**BRITO, J. L. N. e S.; BERNARDO FILHO, O. ; SILVEIRA, M. T., 2005, Visualização e Medição Estereoscópicas de Imagens Fotogramétricas Digitais. In: XXII Congresso Brasileiro de Cartografia, 2005, Macaé.**

**BRITO, J. L. N. e S.; SILVEIRA, F. J. C., 2005, Fototriangulação pelo Método dos Feixes Perspectivos. In: XXII Congresso Brasileiro de Cartografia, 2005, Macaé.**

**BRITO, J. L. N. e S.; BERNARDO FILHO, O. ; COELHO F., L. C. T.; MOTA, G. L. A.; RIBEIRO, J. A.; SILVEIRA, M. T., 2005, Projeto e-foto: Uma Estação Fotogramétrica Digital Livre. In: XXII Congresso Brasileiro de Cartografia, 2005, Macaé.**

**COELHO F., L. C. T.,** 2002, Estação Fotogramétrica Digital Instituto Militar de Engenharia Rio de Janeiro, Brasil.

**COELHO F., L. C. T. .,** 2004, The Development of a Digital Photogrammetric Softcopy Kit for Educational Purposes. In: XXth ISPRS Congress, 2004, Istambul.

**COSMIC,** 2003, COSMIC FFP Measurement Manual Version 2.2, COSMIC.

**DEKKERS, C. A.,** 1998, Function Points and Measurament – What’s a Function Point, QAI Journal, EUA.

**DEKKERS, C. A.,** 1998, Demystifying Function Points: Let’s Understand Some Terminology, IT Metrics Strategies, EUA.

**DEKKERS, C. A.,** 1999, Use Cases and Functions Points – Where’s to Fit?, IT Metrics Strategies, EUA.

**GALVÃO, A.,** 1999, Pontos de Função como Ferramenta para o Gerenciamento de Projetos de Sistemas, <http://www.bfpug.com.br/Artigos/PalestraAnaGalvaoFPA.htm>, acesso em 10.08.2006.

**HAZAN, C.,** Implantação de um processo de medições de software, [http://www.bfpug.com.br/Artigos/Palestra%20 Medicoes%20Claudia%20Hazan.pdf](http://www.bfpug.com.br/Artigos/Palestra%20Medicoes%20Claudia%20Hazan.pdf), acesso em:26.08.2006.

**HAZAN, C.,** Análise de pontos por função - Uma aplicação da gerência de sub-contratação de software, [http://www.bfpug.com.br/Artigos/Palestra\\_Claudia\\_Subcontratacao.pdf](http://www.bfpug.com.br/Artigos/Palestra_Claudia_Subcontratacao.pdf), acesso em: 22.08.2006.

**HAZZAN, S.; IEZZI, G.,** 1985, Fundamentos de Matemática Elementar, Volume 4, Atual Editora, São Paulo - SP, Brasil.

**HILL**, P. R., 2004, Software Development Projects in Government – performance practices and predictions, International Software Benchmarking Standards Group, [www.isbsg.org](http://www.isbsg.org), acesso em: 15.08.2006

**JÚNIOR**, O. S. L., 2002, Análise de Pontos por Função Fuzzy, Dissertação de mestrado, Universidade de Fortaleza, Fortaleza – CE, Brasil.

**LEFFINGWELL**, D.; **WIDRIG**, D., 2000, Managing Software Requirements - A Unified Approach, 1ª Edição, Addison Wesley Logman, EUA.

**LING**, M.; **Kececi**, N.; **Smidts**, C., Function Point Analysis: An Application to a Nuclear Reactor Protection System,  
<http://scholar.google.com/url?sa=U&q=http://www.lrgl.uqam.ca/publications/pdf/718.pdf>, acesso em: 17.01.2007

**LONGSTREET**, D., 2000, A Utilidade dos Pontos de Função,  
<http://www.bfpug.com.br/Artigos/MuitosUsos.htm>, acesso em: 22.08.2006.

**LÓPEZ**, P.A.P, 2005, COCOMO II – Um Modelo para Estimativa de Custos de Gerência de Projetos, <http://www.ulbra-to.br/ensino/43020/artigos/anais2005/anais/COCOMO.pdf>, acesso em 10.01.2007.

**NOVO**, E., 1992, Sensoriamento Remoto Princípios e Aplicações, Editora Edgard Blucher, 2ª Edição, São Paulo-SP, Brasil.

**PARO**, C. J. Medidas de Tamanho de Desenvolvimento e de Melhorias de Software,  
<http://www.bfpug.com.br/Artigos/Medidas%20de%20tamanho%20de%20desenvolvimento%20e%20de%20melhorias%20de%20software.doc>, acesso em: 22.08.2006.

**PECH**, G., 2002, Como gerenciar na prática projetos de desenvolvimento de software,  
<http://www.bfpug.com.br/Artigos/PalestraPech.htm>, acesso em: 23.08.2006.



**RAMOS, C. S., ANQUETIL, K., M., O., 2004,** Conhecendo sistemas legados através de métricas de software, Sociedade Brasileira de Computação, Porto Alegre–RS, Brasil.

**REDGATE, N.;** Tichenor, C., 2001, Measure Size Complexity of Algorithms Using Function Points, CrossTalk, <http://www.stsc.hill.af.mil/crossTalk/2001/02/redgate.html>  
acesso em: 24.08.2006

**REDGATE, N.;** Tichenor, C., 2002, Measuring Calculus Integration Formulas Using Function Point Analysis, CrossTalk,  
<http://www.stsc.hill.af.mil/CrossTalk/2002/06/redgatetichenor.pdf>., acesso em: 24.08.2006.

**RIBEIRO, A. L., 2001,** Análise de Pontos de Função,  
<http://www.bfpug.com.br/Artigos/PalestraAltair1.htm>, acesso em: 22.08.2006.

**SILVEIRA, M. T., 2005,** Visualização e Medição Estereoscópicas de Imagens Fotogramétricas Digitais, Dissertação de Mestrado, Universidade do Estado do Rio de Janeiro, Rio de Janeiro – RJ, Brasil.

**SOFTWAREMETRICS, 2004,** Function Points Analysis Training Course,  
<http://www.softwaremetrics.com/Function%20Point%20Training%20Booklet%20New.pdf>, acesso em: 15.08.2006.

**TORRES, M. B., 2004,** Análise de Pontos de Função Estimativa Qualitativa x Estimativa Quantitativa, <http://www.bfpug.com.br/Artigos/APF%20-%20Qualitativo%20x%20Quantitativo.pdf>, acesso em: 17.01.2007.

**UKSMA, 1998,** MK Function Point Analysis Counting Practices Manual,  
<http://www.uksma.co.uk/public/mkIIr131.pdf>, acesso em: 10.12.2006

**USC, 1999,** COCOMO II Model Definition Manual,  
[ftp://ftp.usc.edu/pub/soft\\_engineering/COCOMOII/cocomo99.0/modelman.pdf](ftp://ftp.usc.edu/pub/soft_engineering/COCOMOII/cocomo99.0/modelman.pdf)  
acesso em: 20.04.2007

**USC**, 1999, USC COCOMO II Reference Manual,

[ftp://ftp.usc.edu/pub/soft\\_engineering/COCOMOII/cocomo99.0/userman.pdf](ftp://ftp.usc.edu/pub/soft_engineering/COCOMOII/cocomo99.0/userman.pdf)

acesso em: 20.04.2007

**VAZQUEZ**, C. E.; **SIMÕES**, G. S.; **ALBERT**, R. M., 2003, Análise de Pontos de Função, 3a Edição, Editora Érica, São Paulo - SP, Brasil.

**VOGELEZANG**, F., 2005, COSMIC Full Function Points – The next Generation of functional sizing, Software Measurement European Forum - SMEF 2005, Roma, Itália, pp.281-289.