

**DESENVOLVIMENTO DE UM SISTEMA DE
DIGITALIZAÇÃO TRIDIMENSIONAL DE
SUPERFÍCIES BASEADO EM VISÃO
COMPUTACIONAL COM VARREDURA A
LASER PARA USO EM METROLOGIA
DIMENSIONAL**

Luciano Selva Ginani

**DISSERTAÇÃO DE MESTRADO
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**FACULDADE DE TECNOLOGIA
UNIVERSIDADE DE BRASÍLIA**

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**DESENVOLVIMENTO DE UM SISTEMA DE DIGITALIZAÇÃO
TRIDIMENSIONAL DE SUPERFÍCIES BASEADO EM VISÃO
COMPUTACIONAL COM VARREDURA A LASER PARA
USO EM METROLOGIA DIMENSIONAL**

LUCIANO SELVA GINANI

ORIENTADOR: JOSÉ MAURÍCIO SANTOS TORRES DA MOTTA

DISSERTAÇÃO DE MESTRADO EM SISTEMAS MECATRÔNICOS

**PUBLICAÇÃO: ENM.DM – 18A/2008
BRASÍLIA/DF: MAIO – 2008**

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**DESENVOLVIMENTO DE UM SISTEMA DE DIGITALIZAÇÃO
TRIDIMENSIONAL DE SUPERFÍCIES BASEADO EM VISÃO
COMPUTACIONAL COM VARREDURA A LASER PARA USO EM
METROLOGIA DIMENSIONAL**

LUCIANO SELVA GINANI

**DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE
ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA DA
UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM
SISTEMAS MECATRÔNICOS**

APROVADA POR:

**PROF. DR. JOSÉ MAURÍCIO S. T. DA MOTTA (ENM-UnB)
(Orientador)**

**PROF. DR. DÍBIO LEANDRO BORGES (CIC-UnB)
(Examinador Externo)**

**PROF. DRA. RAQUEL FRIZERA VASSALLO (DEL-UFES)
(Examinador Externo)**

BRASÍLIA/DF, 28 DE MAIO DE 2008

FICHA CATALOGRÁFICA

GINANI, LUCIANO SELVA

Desenvolvimento de um Sistema de Digitalização Tridimensional de Superfícies Baseado em Visão Computacional com Varredura a Laser para Uso em Metrologia Dimensional

[Distrito Federal] 2008.

xviii, 187p., 297 mm (ENM/FT/UnB, Mestre, Sistemas Mecatrônicos, 2008).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Mecânica.

1. Digitalização de Superfícies

2. Varredura Laser

3. Calibração de Câmeras

4. Visão Computacional

I. ENM/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

Ginani, L.S. (2008). Desenvolvimento de um Sistema de Digitalização Tridimensional de Superfícies Baseado em Visão Computacional com Varredura a Laser para Uso em Metrologia Dimensional. Dissertação de Mestrado em Sistemas Mecatrônicos, Publicação ENM.DM-18A/2008, Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 187p.

CESSÃO DE DIREITOS

AUTOR: Luciano Selva Ginani.

TÍTULO: Desenvolvimento de um Sistema de Digitalização Tridimensional de Superfícies Baseado em Visão Computacional com Varredura a Laser para Uso em Metrologia Dimensional.

GRAU: Mestre ANO: 2008

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

Luciano Selva Ginani
SQSW 103 Bl. A Apt. 603.
70.670.-301 Brasília – DF – Brasil.

AGRADECIMENTOS

Agradeço a todos aqueles que de alguma forma contribuíram no desenvolvimento desse trabalho, tornando possível sua conclusão.

Luciano Selva Ginani

DEDICATÓRIA

Dedico esse trabalho a todos aqueles que sempre acreditaram em meu potencial e me incentivaram e continuam a incentivar em minha busca por excelência.

Luciano Selva Ginani

DESENVOLVIMENTO DE UM SISTEMA DE DIGITALIZAÇÃO TRIDIMENSIONAL DE SUPERFÍCIES BASEADO EM VISÃO COMPUTACIONAL COM VARREDURA A LASER PARA USO EM METROLOGIA DIMENSIONAL**Autor: Luciano Selva Ginani****Orientador: José Maurício Santos Torres da Motta****Programa de Pós-graduação em Sistemas Mecatrônicos****Brasília, maio de 2008**

A grande maioria dos sistemas atuais de digitalização tridimensional de superfícies baseia-se no uso de sensores de posicionamento para a medição da rotação de uma fonte emissora de luz. Esses sensores impõem restrições sobre o volume e a distância de trabalho do digitalizador e, como a reconstrução é normalmente baseada em equações de triangulação, eles também geram incertezas de medição devido à relação não-linear entre a posição angular e a distância. Para superar essas dificuldades, removendo a necessidade do uso de sensores angulares de posição de alto custo e aumentando a distância de medição, apresenta-se nesse trabalho o desenvolvimento de um sistema de digitalização tridimensional de superfícies compacto, de fácil transporte e auto-calibrável. Com o uso de técnicas de visão computacional elimina-se a necessidade de qualquer tipo de sensor de posicionamento e com o sistema de auto-calibração desenvolvido dispensa-se a necessidade de outros sistemas de medição para a calibração. Todas as técnicas de calibração de câmeras, processamento de imagens e a arquitetura completa do sistema são discutidas e apresentadas junto com um comparativo do sistema desenvolvido e sistemas comerciais e em desenvolvimento em diferentes instituições de pesquisa e desenvolvimento.

DEVELOPMENT OF A TRIDIMENSIONAL SURFACE DIGITALIZATION SYSTEM BASED ON COMPUTER VISION AND LASER SCANNING WITH METROLOGICAL PURPOSES**Author: Luciano Selva Ginani****Supervisor: José Maurício Santos Torres da Motta****Programa de Pós-graduação em Sistemas Mecatrônicos****Brasília, May of 2008**

Most of today's 3D laser scanning systems for object model digitalization rely on angular position sensors to measure rotation of a laser diode and to reconstruct the object. However, those types of sensors restrict the object size in target and its distance from the image sensor, since reconstruction is usually based on equations for geometric triangulation, leading to high measurement errors due to the non-linear relation between angular positions and distances. In order to overcome the need of high cost angular position sensors and to increase measurement distances it is proposed here a 3D surface scanner based on computer vision that is compact, easy to transport and self-calibrated. The system calibration includes camera calibration using radial alignment constraints, image processing routines and determination of system parameters. The complete system architecture with two laser diodes and a single camera is shown, the theoretical basis, hardware implementation and practical results obtained are presented, discussed and compared with different systems available.

ENTWICKLUNG EINES DREIDIMENSIONALEN MESS-SYSTEMS GESTÜTZT AUF MASCHINEN VISION FÜR ANWENDUNGEN IN FREIFORMFLÄSCHEDIGITALISIERUNG

Autor: Luciano Selva Ginani

Betreuer: José Maurício Santos Torres da Motta

Magisterprogramm von Mechatronischen Systemen

Brasília, Mai 2008

Die Meiste der heutigen 3D Laserscansystemen für Objektmodelldigitalisierung sind auf angulare Positionensensoren angewiesen, um die Drehung einer Laserdiode zu messen und die Objektstruktur zu rekonstruieren. Jedoch, diese Arten von Sensoren beschränken die Zielobjektgröße und ihre Entfernung vom Bildsensor, da der Wiederaufbau sich normalerweise auf Gleichungen für geometrische Triangulation gegründet ist, was zu hohen Messungsfehlern führt, aufgrund von der Nichtlinearbeziehung zwischen angularen Positionen und Entfernungen. Um die Notwendigkeit der teureren angularen Positionensensoren zu überwinden und Messungsentfernungen zu erhöhen, wird hier ein auf Computervision gegründeter 3D Flächescanner, der kompakt, leicht zu transportieren und selbstkalibriert ist, vorgeschlagen. Die Systemkalibrierung schließt Kamerakalibrierung ein, wo Strahlenförmigausrichtungseinschränkungen, Bildentwicklungsroutinen und volle Bestimmung der Systemparameter benutzt werden. Die komplette Systemarchitektur mit zwei Laserdioden und eine einzelne Kamera wird gezeigt, die theoretische Basis, die Hardware-Einsetzung und die erzielten praktischen Ergebnisse werden präsentiert, diskutiert und mit verschiedenen verfügbaren Systemen verglichen.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	SISTEMAS COMERCIAIS ATUAIS	2
1.1.1	ROMER INC.	2
1.1.2	LASER DESIGN INC.	3
1.1.3	LEICA	4
1.2	O SISTEMA PROPOSTO	5
2	COMPONENTES DE UM SISTEMA DE VISÃO	7
2.1	CÂMERAS	7
2.1.1	Projeção Central	7
2.1.2	Modelos de Câmeras	9
2.1.3	O Modelo RAC	9
2.1.4	O Modelo RAC Modificado	14
2.2	LENTE	16
2.3	FILTROS	17
2.4	LASERS	18
3	O SISTEMA VISSCAN-3D	20
3.1	O SISTEMA ÓPTICO	20
3.1.1	As Restrições de Montagem do Sistema	27
3.2	O SISTEMA ELETRO-MECÂNICO	28
3.2.1	Versão de Laboratório	29
3.2.2	Versão de Campo	29
3.3	O SISTEMA DE CALIBRAÇÃO	31
3.4	O SISTEMA COMPUTACIONAL	32
4	SOFTWARE E PROGRAMAÇÃO	34
4.1	PROCESSAMENTO DE IMAGENS	34
4.1.1	Principais Tarefas de Processamento do Sistema VISSCAN-3D	39
4.2	PROCESSAMENTO DA NUVEM DE PONTOS	46
4.3	O SISTEMA GRÁFICO DE VISUALIZAÇÃO	49
4.3.1	OpenGL	49
4.3.2	A Máquina Gráfica Implementada	49
5	CALIBRAÇÃO AUTOMÁTICA	53
5.1	CALIBRAÇÃO DA CÂMERA	53
5.1.1	Obtenção dos Pontos de Calibração	53
5.1.2	Calibração do Modelo RAC	55
5.1.3	Calibração da Razão μ	58
5.1.4	Obtenção do Centro da Imagem	60
5.2	CALIBRAÇÃO DO SISTEMA ÓPTICO	61
6	SIMULAÇÃO	68
6.1	A CÂMERA RAC VIRTUAL	68
6.2	O DIODO LASER VIRTUAL	70
6.3	SUPERFÍCIES VIRTUAIS	70
6.3.1	Plano Virtual	70
6.4	SIMULAÇÃO DO PROCESSO DE DIGITALIZAÇÃO	72
6.4.1	Digitalização do Plano Virtual	72
6.5	VALIDAÇÃO DO PROCESSO DE DIGITALIZAÇÃO	73
7	ESTUDOS DE PRECISÃO	77
7.1	PRECISÃO TEÓRICA DO SISTEMA	77
7.1.1	Resolução do Sistema Óptico	77
7.1.2	Precisão do Sistema de Digitalização	83
7.2	PRECISÃO EXPERIMENTAL DO SISTEMA	91
7.3	SENSOREAMENTO DA ANGULAÇÃO DO DIODO LASER	103
7.3.1	Avaliação Teórica	103
7.3.2	Avaliação Experimental	106
8	RESULTADOS OBTIDOS	110
9	CONCLUSÕES	124
10	TRABALHOS FUTUROS	125
10.1	DESENVOLVIMENTOS FUTUROS DO SISTEMA VISSCAN-3D	125
10.2	NOVAS LINHAS DE PESQUISA	125
	REFERÊNCIAS BIBLIOGRÁFICAS	127
	APÊNDICES	130

APÊNDICE A: Diferenciação Numérica	131
APÊNDICE B: Solução de Sistemas de Equações	135
APÊNDICE C: Formatos de Arquivos 3D	139
APÊNDICE D: Funções para Processamento de Imagens	149
APÊNDICE E: Funções para Processamento de Nuvens de Pontos	170
APÊNDICE F: O Processo de Calibração do Sistema Passo a Passo	181

LISTA DE TABELAS

Tabela 1.1. Especificações do Scanner Laser LSP.....	4
Tabela 3.1. Parâmetros Físicos do Sistema VISSCAN-3D Versão de Laboratório.....	29
Tabela 3.2. Parâmetros Físicos do Sistema VISSCAN-3D Versão de Campo.....	31
Tabela 4.1. Lista de Funções de Processamento de Imagens.....	36
Tabela 4.2. Nomes Padrões das Imagens Capturadas.....	41
Tabela 4.3. Lista de Funções de Processamento da Nuvem de Pontos.....	46
Tabela 4.4. Lista de Funções de Detecção de Características da Nuvem de Pontos.....	47
Tabela 4.5. Opções de Renderização da Superfície Digitalizada.....	50
Tabela 6.1. Parâmetros para Geração do Plano Virtual.....	71
Tabela 7.1. Parâmetros da Câmera Utilizada no Sistema de Campo.....	80
Tabela 7.2. Parâmetros da Câmera Utilizada no Sistema de Laboratório.....	81
Tabela 7.3. Erro Máximo Teórico da Arquitetura de Digitalização.....	89
Tabela 7.4. Erro Teórico do Sensoreamento da Angulação do Laser.....	106
Tabela 8.1. Orientações Calculadas.....	111
Tabela D.1. Sintaxes Possíveis para uso da Função de Seleção de Região.....	158
Tabela D.2. Sintaxes Possíveis para uso da Função de Cálculo do Centro Geométrico.....	160
Tabela D.3. Parâmetros da Função de Detecção de Retas com Transformada de Hough.....	164
Tabela E.1. Diferentes Sintaxes da Função de Curvatura.....	175
Tabela E.2. Código de Cores para a Classificação HK.....	176
Tabela F.1. Parâmetros para Cálculo das Coordenadas da Placa de Calibração.....	183

LISTA DE FIGURAS

Figura 1.1. Scanner Laser G-Scan RX2.	3
Figura 1.2. Scanner Laser LSP.	4
Figura 1.3. Scanner Laser T-Scan.	5
Figura 1.4. Scanner Laser Proposto.	6
Figura 2.1. Elementos Básicos de uma Câmera.	7
Figura 2.2. Modelo Idealizado de Projeção Central.	8
Figura 2.3. Geometria Básica do Modelo RAC (Tsai, 1987).	10
Figura 2.4. Transformações de Coordenadas do Modelo RAC.	11
Figura 2.5. Sistema De coordenadas da Imagem e de Armazenamento no Computador.	14
Figura 2.6. Alguns Tipos de Lentes Simples.	16
Figura 2.7. Espectro Visível e Comprimento de Onda.	17
Figura 2.8. Tipos Básicos de Filtros Ópticos Ideais e seu Efeito.	18
Figura 2.9. Princípio de Funcionamento de um Diodo Laser.	19
Figura 2.10. Alguns Padrões de Projeção de Lasers.	19
Figura 3.1. Esquemático do Sistema VISSCAN-3D.	20
Figura 3.2. Projeção de um Plano de Luz Laser sobre uma Superfície.	21
Figura 3.3. O Primeiro Plano de Luz e a Geração da Imagem de um Ponto da Linha Laser... 21	21
Figura 3.4. Triangulação através da Projeção do Laser na Superfície.	21
Figura 3.5. O Segundo Plano de Luz e a Geração da Imagem de um Ponto da Linha Laser... 23	23
Figura 3.6. Validade das Equações das Linhas Lasers.	24
Figura 3.7. Efeito do Desalinhamento do Laser Móvel.	25
Figura 3.8. Parâmetros de Desalinhamento.	25
Figura 3.9. Algoritmo de Digitalização de Superfícies.	26
Figura 3.10. Ponto de Cruzamento dos Feixes Lasers.	27
Figura 3.11. Efeito da Não-Perpendicularidade dos Feixes Lasers.	28
Figura 3.12. Versão de Laboratório do Hardware do Sistema VISSCAN-3D.	29
Figura 3.13. Versão de Campo do Hardware do Sistema VISSCAN-3D.	30
Figura 3.14. Distribuição do Custo do Sistema.	30
Figura 3.15. Distribuição do Peso do Sistema.	30
Figura 3.16. Placa de Calibração.	31
Figura 3.17. Tela Inicial do Software de Calibração.	33
Figura 3.18. Tela Inicial do Software de Digitalização.	33
Figura 4.1. Processamento de Imagens na Etapa de Calibração.	35
Figura 4.2. Processamento de Imagens na Etapa de Digitalização.	35
Figura 4.3. Estrutura Básica do Script de Processamento de Imagens.	37
Figura 4.4. Exemplo de Script de Processamento de Imagens.	37
Figura 4.5. Efeito do Script sobre uma Imagem de Teste.	38
Figura 4.6. Software de Avaliação de Scripts de Processamento de Imagens.	39
Figura 4.7. Scripts de Pré e Pós Processamento para Extração do Ponto de Interseção.	40
Figura 4.8. Fluxograma dos Procedimentos de Captura de Imagens.	40
Figura 4.9. Tipos de Imagens Capturadas pelo Sistema VISSCAN-3D.	41
Figura 4.10. Etapas de Pré e Pós-Processamento para Extração do Ponto de Interseção.	42
Figura 4.11. Scripts de Pré e Pós Processamento para Extração da Linha Laser.	42
Figura 4.12. Exemplo das Etapas de Processamento para Extração da Linha Laser.	43
Figura 4.13. Direções de Varredura da Imagem.	44
Figura 4.14. Estrutura Básica do Script de Processamento da Nuvem de Pontos.	47
Figura 4.15. Exemplo de Script de Processamento da Nuvem de Pontos.	48
Figura 4.16. Efeito do Script sobre uma Superfície de Teste.	48

Figura 4.17. Diferentes Formas de Renderização Disponíveis.	50
Figura 4.18. Seqüência da Varredura Laser.	51
Figura 4.19. Renderização baseada em Triângulos e Quadrados.	51
Figura 5.1. Distribuição dos Pontos de Calibração.	54
Figura 5.2. Calibração Iterativa da Razão dos Fatores de Escala.	60
Figura 5.3. Parâmetros do Sistema de Digitalização.	61
Figura 5.4. Planos Lasers Gerados e Parâmetros do Sistema.	63
Figura 5.5. Algoritmo de Calibração do Laser Móvel.	67
Figura 5.6. Algoritmo de Calibração do Laser Fixo.	67
Figura 6.1. Plataforma de Simulação do Sistema VISSCAN-3D.	68
Figura 6.2. Geração de um Plano Virtual para Simulação.	71
Figura 6.3. Simulação e Validação do Sistema VISSCAN-3D.	74
Figura 6.4. Diferentes Orientações do Plano Virtual.	74
Figura 6.5. Erro Total de Mapeamento na Validação do Sistema.	75
Figura 6.6. Erro Médio de Mapeamento na Validação do Sistema.	75
Figura 7.1. Resolução de um Sistema Óptico.	78
Figura 7.2. Fator de Amplificação de uma Lente.	79
Figura 7.3. Tamanhos Padrões de Sensores CCD (Unidades em mm).	79
Figura 7.4. Determinação do Campo de Visão do Sistema Óptico.	79
Figura 7.5. Resolução Teórica do Conjunto Óptico da Versão de Laboratório.	82
Figura 7.6. Resolução Teórica do Conjunto Óptico da Versão de Campo.	82
Figura 7.7. Área de Trabalho Teórica do Sistema VISSCAN-3D.	83
Figura 7.8. Definição da Amplitude de Varredura para Análise de Precisão.	86
Figura 7.9. Definição das Angulações Máximas para Análise de Precisão.	86
Figura 7.10. Termo da Precisão do Sistema Dependente da Angulação do Laser.	87
Figura 7.11. Erros Teóricos dos Sistemas de Digitalização Implementados.	89
Figura 7.12. Precisão Teórica dos Sistemas de Digitalização Implementados.	90
Figura 7.13. Singularidade do Sistema para $d_y \cot(\theta_y) = -b_y$	91
Figura 7.14. Bloco Metrológico.	92
Figura 7.15. Diferentes Orientações para Digitalização do Bloco Metrológico.	92
Figura 7.16. Digitalização do Bloco Metrológico.	92
Figura 7.17. Esquemático de Digitalização do Bloco Metrológico e suas Bordas.	93
Figura 7.18. Erros de Medição do Bloco Padrão com Z=150 para a Versão de Laboratório. .	94
Figura 7.19. Erros de Medição do Bloco Padrão com Z=200 para a Versão de Laboratório. .	94
Figura 7.20. Erros de Medição do Bloco Padrão com Z=225 para a Versão de Laboratório. .	95
Figura 7.21. Erros de Medição do Bloco Padrão com Z=250 para a Versão de Laboratório. .	95
Figura 7.22. Erros de Medição do Bloco Padrão com Z=350 para a Versão de Laboratório. .	96
Figura 7.23. Erros de Medição do Bloco Padrão com Z=150 para a Versão de Campo.	96
Figura 7.24. Erros de Medição do Bloco Padrão com Z=200 para a Versão de Campo.	97
Figura 7.25. Erros de Medição do Bloco Padrão com Z=225 para a Versão de Campo.	97
Figura 7.26. Erros de Medição do Bloco Padrão com Z=250 para a Versão de Campo.	98
Figura 7.27. Erros de Medição do Bloco Padrão com Z=350 para a Versão de Campo.	98
Figura 7.28. Erros Obtidos com a Distância de Bordas para a Versão de Laboratório.	99
Figura 7.29. Erros Obtidos com a Distância de Bordas para a Versão de Campo.	99
Figura 7.30. Erros Obtidos com a Planicidade do Bloco para a Versão de Laboratório.	100
Figura 7.31. Erros Obtidos com a Planicidade do Bloco para a Versão de Campo.	100
Figura 7.32. Precisão da Versão de Laboratório do Sistema VISSCAN-3D.	101
Figura 7.33. Precisão da Versão de Campo do Sistema VISSCAN-3D.	101
Figura 7.34. Arestas Espelhadas do Bloco Metrológico.	102
Figura 7.35. Efeito de Reflexão Observado nos Ensaios Experimentais.	102
Figura 7.36. Precisão Teórica dos Sistemas de Digitalização Implementados.	105

Figura 7.37. Estágio Rotatório para Avaliação do Cálculo da Angulação do Laser.....	106
Figura 7.38. Posições do Plano para Cálculo da Angulação.....	106
Figura 7.39. Diferenças entre as Medições de Angulação do Laser.....	107
Figura 7.40. Barras de Precisão Experimental do Cálculo da Angulação do Laser.....	108
Figura 7.41. Precisão Experimental e Teórica do Cálculo da Angulação do Laser.....	108
Figura 8.1. Esquemático da Digitalização.....	110
Figura 8.2. Estágio Rotacional Graduado Manual.....	110
Figura 8.3. Digitalização de uma Superfície Plana.....	111
Figura 8.4. Orientações Calculadas.....	111
Figura 8.5. Digitalização de uma Esfera.....	112
Figura 8.6. Peça Mecânica e sua Digitalização.....	113
Figura 8.7. Cordão de Solda e sua Digitalização.....	114
Figura 8.8. Processamento do Cordão de Solda.....	114
Figura 8.9. Digitalização de uma Calculadora.....	115
Figura 8.10. Digitalização de um Aparelho Celular.....	116
Figura 8.11. Digitalização de um Molde.....	117
Figura 8.12. Digitalização de um Mouse de Computador.....	118
Figura 8.13. Digitalização de um Vaso.....	119
Figura 8.14. Fonte de Geração CC.....	119
Figura 8.15. Digitalização e Aplicação de Textura.....	120
Figura 8.16. As Duas Faces de uma Chave.....	120
Figura 8.17. Digitalização 3D de uma Chave.....	121
Figura 8.18. Digitalização da Face de uma Boneca.....	123
Figura A.1. Máscara para Cálculo da Primeira Derivada de uma Imagem.....	134
Figura A.2. Máscara para Cálculo da Segunda Derivada de uma Imagem.....	134
Figura C.1. Organização Hierárquica Simplificada de um Arquivo 3DS.....	140
Figura C.2. Formato de Arquivo PTO.....	142
Figura C.3. Exemplo de Arquivo PTO.....	143
Figura C.4. Formato de Arquivo IMG.....	144
Figura C.5. Exemplo de Arquivo IMG.....	145
Figura C.6. Formato de Arquivo DEF.....	147
Figura C.7. Exemplo de Arquivo DEF.....	148
Figura D.1. Exemplo de Adição de Duas Imagens.....	149
Figura D.2. Exemplo de Subtração de Duas Imagens.....	150
Figura D.3. Exemplo de Cálculo da Mediana de um Pixel de uma Imagem.....	150
Figura D.4. Função de Mediana sobre uma Imagem de Teste.....	151
Figura D.5. Expansão das Bordas da Imagem para Aplicação de Filtros.....	151
Figura D.6. Elemento Estrutural e Dilatação de uma Imagem.....	152
Figura D.7. Função de Dilatação sobre uma Imagem de Teste.....	153
Figura D.8. Elemento Estrutural e Erosão de uma Imagem.....	154
Figura D.9. Função de Erosão sobre uma Imagem de Teste.....	154
Figura D.10. Seleção e Avaliação de Área em Blobs.....	155
Figura D.11. Função de Blobs sobre uma Imagem de Teste.....	155
Figura D.12. Raios de um Pixel nas Direções X e Y da Imagem.....	156
Figura D.13. Função de Raio Máximo e Raio Mínimo sobre uma Imagem de Teste.....	156
Figura D.14. Função de Excentricidade sobre uma Imagem de Teste.....	157
Figura D.15. Operador de Seleção de Região Válida Sobre uma Imagem de Teste.....	157
Figura D.16. Formato do Arquivo para Leitura do Centro da Região de Interesse.....	158
Figura D.17. Varredura da Imagem para Determinação da Região de Interesse.....	158
Figura D.18. Janela de Cálculo do Centro Geométrico.....	159
Figura D.19. Cálculo do Centro Geométrico de uma Imagem de Teste.....	160

Figura D.20. Determinação da Janela de Cálculo do Centro Geométrico.	160
Figura D.21. Transformação de Coordenadas sobre uma Reta.	161
Figura D.22. Transformação de Coordenadas sobre um Ponto.	162
Figura D.23. Detecção de Retas Transformada em Detecção de Cruzamento.	162
Figura D.24. Reta no Espaço (X,Y) e Acumuladores Resultantes no Espaço (M,N)	163
Figura D.25. Vizinhança Nula e Efeito sobre a Detecção de Retas.	164
Figura D.26. Vizinhança Nula sobre os Acumuladores da Transformada de Hough.	165
Figura D.27. Arquivo de Saída da Detecção de Retas com a Transformada de Hough.	165
Figura D.28. Arquivo de Saída da Detecção de Interseção de Retas.	166
Figura D.29. Exemplo de Detecção de Interseção de Retas.	167
Figura D.30. Vizinhança de um Ponto em uma Imagem.	167
Figura D.31. Remoção de Vértices Utilizando o Critério de Distância Mínima.	169
Figura D.32. Formato do Arquivo de Saída da Detecção de Vértices.	169
Figura D.33. Detecção de Vértices de uma Imagem de Teste.	169
Figura E.1. Função de Saturação sobre o Perfil de uma Superfície.	170
Figura E.2. Função de Saturação sobre uma Superfície de Teste.	170
Figura E.3. Função de Média sobre uma Superfície de Teste.	171
Figura E.4. Função de Mediana sobre uma Superfície de Teste.	172
Figura E.5. Função de Limiar sobre uma Superfície de Teste.	172
Figura E.6. Função de Gradiente sobre uma Superfície de Teste.	173
Figura E.7. Função de Laplaciano sobre uma Superfície de Teste.	174
Figura E.8. Classificação de Superfícies HK (Trucco e Verri, 1998).	174
Figura E.9. Função de Segmentação HK sobre uma Superfície de Teste.	176
Figura E.10. Função de Curvatura Gaussiana sobre uma Superfície de Teste.	176
Figura E.11. Função de Curvatura Média sobre uma Superfície de Teste.	176
Figura E.12. Plano Médio Calculado para uma Nuvem de Pontos.	179
Figura E.13. Função de Plano Médio sobre uma Superfície de Teste.	179
Figura F.1. Tela Principal do Software de Calibração e Grupos de Funcionalidades.	181
Figura F.2. Barra Deslizante para Aplicação de Binarização por Limiar Fixo em Imagens.	182
Figura F.3. Imagem Capturada e Binarizada da Placa de Calibração.	182
Figura F.4. Parâmetros para Cálculo das Coordenadas da Placa de Calibração.	183
Figura F.5. Seqüência de Entrada dos Pontos da Placa de Calibração.	183
Figura F.6. Interface para Adicionar Coordenadas para a Calibração de Câmeras.	184
Figura F.7. Adição de Coordenadas para a Calibração de Câmeras.	184
Figura F.8. Calibração de Câmeras.	185
Figura F.9. Calibração dos Lasers.	186
Figura F.10. Grupo de Seleção de Eixos.	186
Figura F.11. Adição de Coordenadas da Linha Laser no Eixo X.	186
Figura F.12. Adição de Coordenadas da Linha Laser no Eixo Y.	187

LISTA DE SÍMBOLOS

f	Distância focal da câmera;
k	Coefficiente de distorção de lente da câmera;
s_x	Fator de Erro de Aquisição da Câmera;
dd_y	Inverso da distância entre dois elementos sensores de uma câmera CCD no eixo-Y;
dd_x	Inverso da distância entre dois elementos sensores de uma câmera CCD no eixo-X;
(f_x, f_y)	Fatores de escala da câmera;
μ	Razão entre os fatores de escala da câmera;
(D_x, D_y)	Distorção radial da lente;
(C_x, C_y)	Centro da imagem;
b_y	Distância base no eixo-Y entre a câmera e o plano laser paralelo ao eixo-X;
b_y'	Distância base aparente no eixo-Y entre a câmera e o plano laser paralelo ao eixo-X;
d_y	Distância no eixo-Z entre o eixo de rotação do plano laser e o eixo-Y;
b_x	Distância base no eixo-X entre a câmera e o plano laser paralelo ao eixo-Y;
b_x'	Distância base aparente no eixo-X entre a câmera e o plano laser paralelo ao eixo-Y;
d_x	Distância no eixo-Z entre o eixo de rotação do plano laser e o eixo-X;
θ_y	Ângulo entre o plano laser paralelo ao eixo-X e o eixo-Y;
θ_x	Ângulo entre o plano laser paralelo ao eixo-Y e o eixo-X;
$\cot(\bullet)$	Co-tangente de um ângulo;
θ_{Laser}	Ângulo de abertura da projeção do plano de luz laser;
tg_{Laser}	Tangente do ângulo de abertura da projeção do plano de luz laser;
(x_c, y_c, z_c)	Coordenadas tridimensionais de um ponto qualquer em relação ao sistema de coordenadas da câmera em unidades S.I;

- (x_w, y_w, z_w) Coordenadas tridimensionais de um ponto qualquer em relação a um sistema de coordenadas global em unidades S.I;
- (x_u, y_u) Coordenadas sem distorção da projeção na câmera de um ponto tridimensional em unidades S.I;
- (X_u, Y_u) Coordenadas sem distorção da projeção na câmera de um ponto tridimensional em pixels;
- (x_d, y_d) Coordenadas distorcidas da projeção na câmera de um ponto tridimensional em unidades S.I;
- (X_d, Y_d) Coordenadas distorcidas da projeção na câmera de um ponto tridimensional em pixels;
- (x_{int}, y_{int}) Coordenadas sem distorção da projeção na câmera do ponto de interseção entre os dois planos lasers projetados em uma superfície em unidades S.I;
- (X_{int}, Y_{int}) Coordenadas sem distorção da projeção na câmera do ponto de interseção entre os dois planos lasers projetados em uma superfície em pixels;
- (X_i, Y_i) Posição de memória de um pixel da imagem;
- PMAG* Fator de amplificação primário de uma lente;
- FOV* Campo de visão de um conjunto óptico;
- Res_c Resolução de uma câmera;
- Res_o Resolução de um conjunto óptico;
- d_t Distância de trabalho da câmera;
- \otimes Produto vetorial entre dois vetores.

1 INTRODUÇÃO

Tradicionalmente a realização de medições tridimensionais é, ainda hoje, realizada utilizando-se sistemas de contato como braços de coordenadas (Sansoni e Docchio, 2004, Alves, 1996). Entretanto, com os grandes avanços na tecnologia de escaneamento tridimensional, o uso de sistemas baseados em lasers está se tornando cada vez mais comum em diversas áreas como engenharia reversa, prototipagem, impressão 3D, manufatura, ciências médicas, mapeamento de terras e preservação do patrimônio cultural (Bösemann, 2005, Tognola et al, 2003, Lee *et.al.*, 2001, Barhak e Fischer, 2001, Baltsavias, 1999, Petrov *et.al.*, 1998).

A grande funcionalidade de um digitalizador tridimensional laser é a obtenção rápida e precisa de uma grande massa de dados de posição tridimensional que representa um objeto. Essa massa de dados, ou nuvem de pontos, pode então ser processada para a obtenção de um modelo tridimensional desse objeto. Esses modelos tridimensionais podem então ser utilizados em uma grande variedade de aplicações tanto na engenharia quanto em outras áreas do conhecimento.

Algumas das aplicações onde a digitalização tridimensional vêm sendo utilizada com grande sucesso que podem ser destacadas são: a animação gráfica, a arqueologia, a arquitetura, a odontologia, a medicina, a indústria da moda, a indústria têxtil e de calçados, a criminalística, o desenvolvimento de jogos, a realidade virtual, o design industrial, as indústrias de manufatura, a indústria cinematográfica, a prototipagem rápida, a criação de brinquedos, a engenharia reversa, a construção de moldes, a geração de acervos digitais de museus, e muitas outras. Essa vasta gama de aplicações mostra a importância do desenvolvimento tecnológico na área da digitalização tridimensional, algo que vem sendo realizado em vários laboratórios e empresas que se dedicam não só ao desenvolvimento de técnicas de digitalização mas também ao desenvolvimento de aplicações para essas técnicas.

Atualmente existe uma grande variedade de sistemas comerciais de digitalização 3D baseados em lasers, como os vendidos por empresas como a Leica Inc. (www.leica-geosystems.com), a Romer Inc. (www.romer.com), a Laser Design Inc. (www.laserdesign.com), a Cyberware (www.cyberware.com) e muitas outras.

Entretanto, a grande maioria dos sistemas de digitalização disponíveis atualmente baseia-se no uso de sensores de posicionamento sobre as fontes emissoras de luz ou no uso de mais de uma câmera para realizar a reconstrução da superfície (Hsuehl e Antonsson, 1997, Tognola et al, 2003). O uso desses sensores é um fator limitante desses sistemas, limitando o tamanho dos objetos a serem digitalizados e sua distância até o sistema uma vez que a reconstrução é normalmente baseada no uso de equações de triangulação geométrica, o que gera elevados erros de medição devido à relação não-linear entre a posição angular dos lasers e a distância. Esses sensores e câmeras adicionais não apenas constituem uma fonte de incerteza e imprecisão ao sistema, mas também são uns dos maiores responsáveis pelo seu elevado custo.

Entretanto, com o advento da visão computacional e os recursos disponibilizados por essa nova área da ciência sendo cada vez mais utilizados, é possível utilizar esses recursos no desenvolvimento de um sistema de digitalização tridimensional de superfícies que não necessite do uso de sensores de posicionamento. Dessa forma é possível obter uma boa precisão no sistema juntamente com uma redução no seu custo.

Este trabalho apresenta o desenvolvimento de um sistema de digitalização tridimensional de superfícies com essas características. O sistema proposto é apresentado detalhadamente juntamente com estudos teóricos e experimentais de precisão, além de uma análise de seu custo e viabilidade.

Inicialmente, na Seção 1.1, é realizada uma breve apresentação de alguns sistemas comerciais disponíveis atualmente e do sistema aqui proposto. Em seguida, no Capítulo 2 é introduzida uma revisão bibliográfica sobre os principais componentes dos sistemas óptico de visão, dando ênfase as câmeras e sua modelagem, para então dar-se início no Capítulo 3 à apresentação detalhada do sistema de digitalização desenvolvido.

No Capítulo 4 são apresentadas informações voltadas ao desenvolvimento do *software* do sistema, elucidando sobre as várias estratégias de programação adotadas e sobre as características técnicas e de uso desse *software*.

Já na Capítulo 5 é dada uma maior ênfase à calibração do sistema, sendo apresentado não apenas o desenvolvimento e os procedimentos de calibração, mas também uma análise dos resultados obtidos com o auxílio de um simulador desenvolvido com essa finalidade.

O Capítulo 6 traz a descrição do simulador implementado, apresentando seu desenvolvimento, suas peculiaridades e formas de utilização.

Concluindo a dissertação, os Capítulos 7 e 8 trazem uma avaliação do sistema em termos de precisão e apresenta uma série de digitalizações realizadas utilizando-se o sistema desenvolvido e possíveis aplicações.

1.1 SISTEMAS COMERCIAIS ATUAIS

Nessa seção são apresentados alguns dos sistemas de digitalização tridimensional disponíveis comercialmente. Esses equipamentos e as empresas responsáveis por seu desenvolvimento são brevemente apresentados para ter-se padrões de comparação com o sistema de digitalização laser proposto.

1.1.1 ROMER INC.

A Romer Inc. é uma empresa consagrada na área de equipamentos portáteis de metrologia para medição e inspeção. Fundada em 1973, com o desenvolvimento do primeiro braço articulado de vários

eixos para inspeção de tubos, a Romer é uma empresa que desde então continua desenvolvendo produtos e soluções de metrologia. Hoje, seus equipamentos são utilizados nos mais diversos setores da indústria.

O sistema G-Scan RX2 desenvolvido pela Romer consiste em um digitalizador laser acoplado a um braço articulado de medição conforme pode ser visualizado na Figura 1.1. O sistema digitaliza uma única linha laser com comprimento máximo de 110 mm a uma distância de trabalho entre 124mm e 222 mm, particionando-a em 640 pontos com uma precisão de 0.044 mm.



Figura 1.1. Scanner Laser G-Scan RX2.

1.1.2 LASER DESIGN INC.

A Laser Design Inc. é o líder mundial em desenvolvimento e manufatura de sistemas de digitalização tridimensional desde 1987. Seus sistemas de digitalização são desenvolvidos para acoplamento com máquinas de medição de coordenadas, de forma a oferecer soluções completas para a digitalização precisa.

O sistema Surveyor Laser Probe desenvolvido pela empresa traz diferentes versões para diferentes distâncias de trabalho. Na Tabela 1.1 é possível observar as especificações de duas dessas versões.



Figura 1.2. Scanner Laser LSP.

Tabela 1.1. Especificações do Scanner Laser LSP.

	SLP-250	SLP-2000
Comprimento da Linha Laser [mm]	20-25	180-320
Número de Pontos por Linha	752	752
Distância de Trabalho [mm]	72 - 110	200-500
Precisão [mm]	0.01	0.127

1.1.3 LEICA

Com quase 200 anos de história no desenvolvimento de sistema de medição, a Leica é uma empresa que oferece uma grande variedade de produtos para a digitalização tridimensional de objetos, modelagem, análise e visualização de informações espaciais.

O sistema de digitalização T-Scan desenvolvido pela Leica é um dos poucos digitalizadores laser do mercado que não necessita de uma preparação da superfície a ser digitalizada e é um digitalizador manual que não necessita de alvos de calibração.



Figura 1.3. Scanner Laser T-Scan.

O digitalizador T-Scan não projeta uma linha laser, mas sim uma série de pontos que apresentam um controle automático de intensidade para a compensação da refletividade dos diferentes materiais que podem compor um objeto. Com uma distância de trabalho média de 83 mm e uma largura da linha de digitalização de 90 mm o sistema T-Scan apresenta uma precisão de 0.03 mm e uma resolução da linha laser de 0.1 mm.

1.2 O SISTEMA PROPOSTO

O sistema proposto tem como objetivo o uso de técnicas de visão computacional para a extração de dados diretamente da imagem, tornando desnecessário o uso de sensores de posição sobre os lasers ou o uso de câmeras adicionais. Com a remoção desses componentes do sistema e o uso dessas técnicas deseja-se reduzir o custo do sistema e ao mesmo tempo obter-se uma melhora de precisão.

Para tanto foram desenvolvidas duas versões do sistema proposto. Uma versão de laboratório, onde todos os equipamentos podem ser facilmente modificados e posicionados, tornando possível um estudo experimental da influência de cada um dos parâmetros do sistema, e uma versão de campo.

A versão de campo do sistema foi construída utilizando-se componentes facilmente encontrados comercialmente e dando ênfase à miniaturização de forma a obter-se um sistema de fácil transporte e integração.

Outra característica fundamental do sistema desenvolvido é sua capacidade de auto-calibração. O sistema, com o auxílio de uma placa de calibração, pode ser completamente calibrado, tornando desnecessário o uso de outros sistemas de medição para a realização desse procedimento.

O sistema desenvolvido recebeu o nome VISSCAN-3D e foi projetado e construído no Grupo de Automação e Controle da Universidade de Brasília (GRACO-UnB).

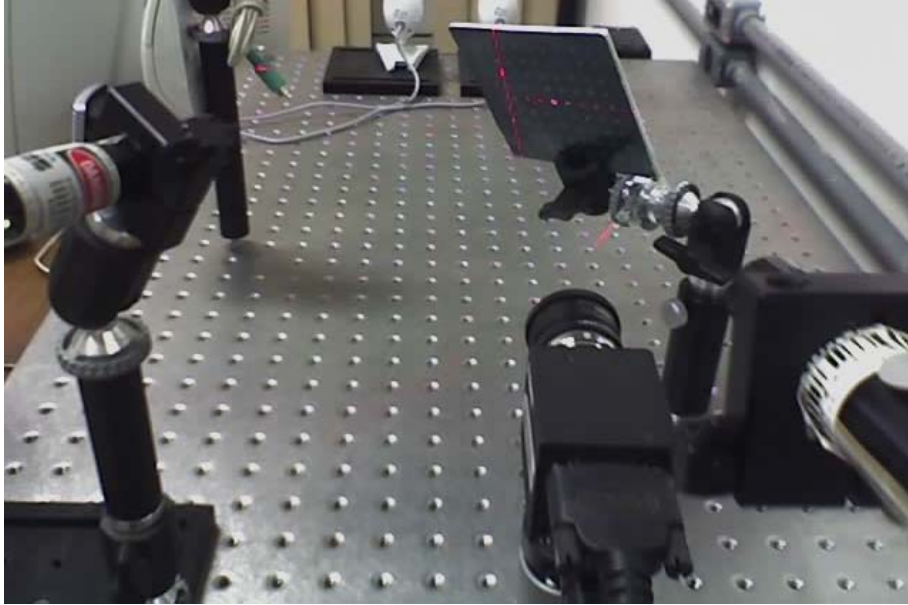


Figura 1.4. Scanner Laser Proposto.

2 COMPONENTES DE UM SISTEMA DE VISÃO

2.1 CÂMERAS

Câmeras são elementos críticos de um sistema de visão. Elas são responsáveis pela captura das imagens que serão utilizadas pelo sistema de processamento para a aquisição dos dados e informações desejados. Uma câmera pode, de maneira geral, ser considerada um sistema óptico conforme o ilustrado na Figura 2.1.

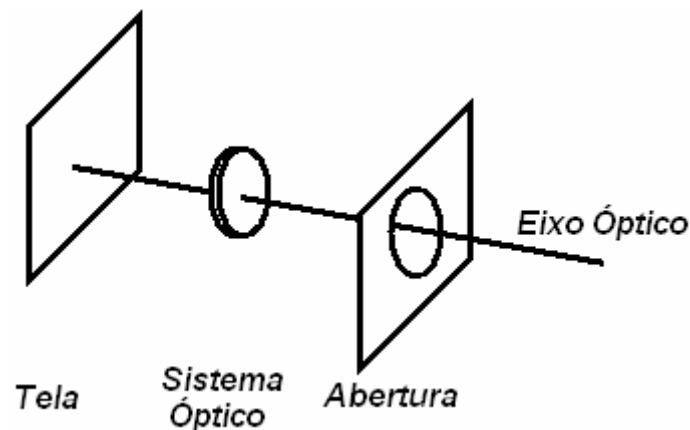


Figura 2.1. Elementos Básicos de uma Câmera.

A formação de uma imagem depende diretamente das características da câmera utilizada para sua aquisição, das características de seus elementos sensores, das características ópticas de suas lentes, da varredura digital do sinal analógico da imagem e das leis geométricas que regem todo o processo. (Klette et. al., 1998, Optical Society of America, 1978)

Esse processo de formação de imagens bidimensionais a partir dos objetos tridimensionais de uma cena é realizado utilizando-se um modelo matemático que tenta descrever cada uma dessas características da câmera.

2.1.1 Projeção Central

A projeção de um objeto tridimensional sobre um plano bidimensional para a formação de uma imagem pode ser descrito essencialmente como central ou paralela (Klette et. al., 1998). Aqui apresenta-se inicialmente o modelo idealizado de projeção central, também conhecido como projeção de perspectiva. Para maiores detalhes sobre a formulação e o uso do modelo de projeção paralela é recomendada a leitura das referências Klette et. Al. (1998) e Hallert (1960).

Sistemas ópticos são, normalmente, sistemas sofisticados, entretanto alguns conceitos básicos que regem esses sistemas podem ser compreendidos através do estudo de um dos sistemas ópticos mais simples, as câmeras *pinhole*.

Um ponto qualquer no espaço tridimensional reflete luz em diferentes direções. Para a obtenção de uma imagem nítida, entretanto, é necessário que todos os raios de luz provenientes de um ponto converjam para um mesmo ponto na imagem. Isso pode ser realizado de duas formas (Trucco e Verri, 1998):

- Através do uso de um sistema óptico composto de lentes, diafragmas e uma variedade de componentes ópticos;
- Reduzindo a abertura da câmera de forma que ela seja somente um ponto, de forma que um único raio de luz de cada ponto ira convergir para a câmera.

A redução da abertura da câmera até um único ponto conforme ilustrado na Figura 2.2 resulta no modelo idealizado de projeção central ou modelo *pinhole*.

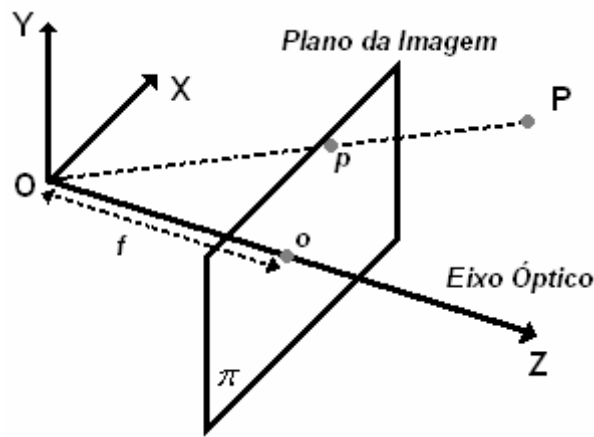


Figura 2.2. Modelo Idealizado de Projeção Central.

O modelo consiste em um plano de imagem π a uma distância f do centro óptico O do sistema, que é o ponto único de abertura para o qual todos os raios de luz convergem. Já o eixo que passa pelo ponto O perpendicularmente ao plano da imagem π é o eixo óptico do sistema. Sendo $P = (X, Y, Z)$ um ponto qualquer e $p = (x, y)$ sua projeção no plano π , as equações de projeção serão dadas por (Klette et. al., 1998, Trucco e Verri, 1998, Zhuang e Roth, 1996):

$$\begin{cases} x = f \frac{X}{Z} \\ y = f \frac{Y}{Z} \end{cases} \quad (2.1)$$

Uma consideração importante a respeito do modelo de projeção central ou perspectiva é o fato de que, devido ao fator $1/Z$, ele é um modelo não-linear.

2.1.2 Modelos de Câmeras

O modelo perspectiva considera a projeção de pontos tridimensionais em um plano bidimensional para a formação de uma imagem, entretanto esse modelo não trata de todas as transformações envolvidas no processo de captura e digitalização de uma imagem. É nesse contexto de englobar todas essas transformações envolvidas que são utilizados outros modelos de câmeras.

A função de um modelo de câmera é relacionar as coordenadas da imagem de um objeto visível pela câmera com as coordenadas desse ponto em relação a um sistema de coordenadas qualquer (Zhuang e Roth, 1996).

Essa relação de coordenadas é realizada utilizando-se dois conjuntos de parâmetros: os parâmetros extrínsecos e os parâmetros intrínsecos (Trucco e Verri, 1998). Os parâmetros extrínsecos são os parâmetros que definem a posição e orientação da câmera com referência a um sistema global de referência e os parâmetros intrínsecos são os parâmetros responsáveis por relacionar as coordenadas de um ponto da imagem em pixels com um ponto de coordenadas no sistema de referência da câmera.

2.1.3 O Modelo RAC

O modelo de Restrição de Alinhamento Radial (*Radial Alignment Constraint* - RAC) foi proposto por Tsai (1987) e, desde então, vem sendo amplamente utilizado nas mais diversas aplicações de visão computacional e provou-se um modelo eficiente tanto no ponto de vista computacional quanto no ponto de vista de precisão (Klette et al, 1996, Zhuang e Roth, 1996).

O modelo RAC apresenta quatro etapas para a transformação das coordenadas globais 3D para as coordenadas de imagem bidimensionais. Na Figura 2.3 é possível observar a geometria básica do modelo RAC.

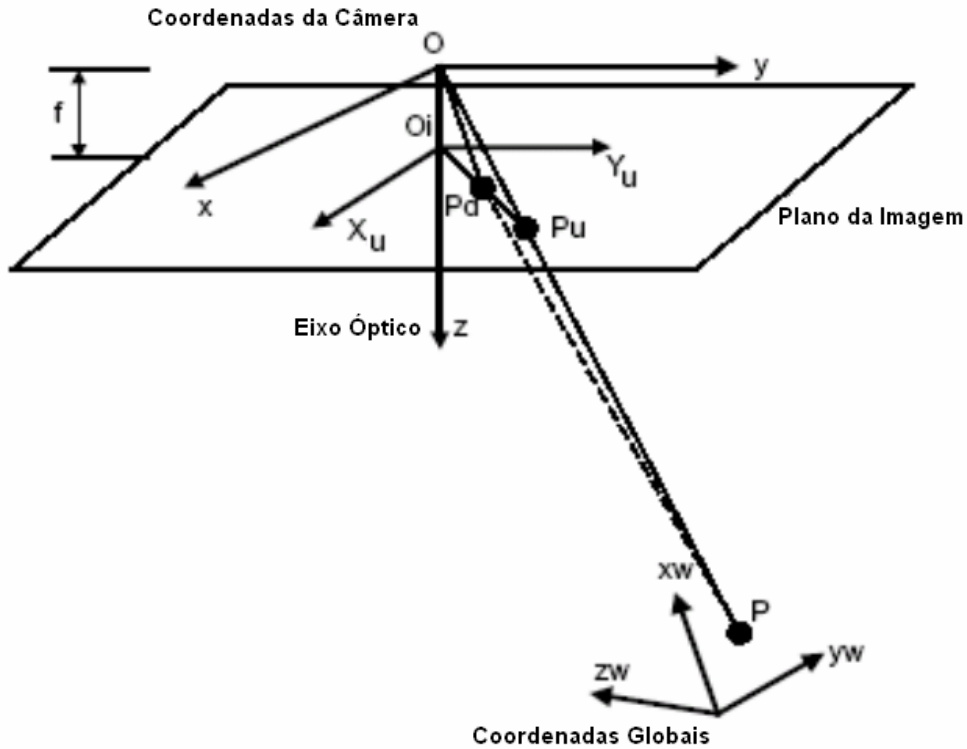


Figura 2.3. Geometria Básica do Modelo RAC (Tsai, 1987).

Na Figura 2.3 é possível observar as coordenadas (x_w, y_w, z_w) de um ponto P em relação a um referencial global qualquer e as coordenadas (x, y, z) desse mesmo ponto P em relação ao sistema de coordenadas da câmera, com origem no centro óptico O e eixo- Z coincidente com o eixo óptico da câmera.

O ponto P_u com coordenadas (x_u, y_u) ilustra a projeção do ponto P no plano da imagem utilizando-se uma projeção de perspectiva que define que um ponto qualquer de um objeto real está conectado com seu ponto de imagem correspondente através de uma linha reta que passa pelo centro óptico O (Zhuang e Roth, 1996).

Já o ponto P_d com coordenadas (x_d, y_d) ilustra o efeito de distorção radial sobre o ponto P_u , visando compensar efeitos de distorção observados nas lentes de câmeras.

Em alguns tipos de câmeras, como as câmeras do tipo CCD, ainda é necessário considerar-se mais uma transformação para o ponto P_i com coordenadas (x_i, y_i) , devido às imagens serem armazenadas na memória do computador em valores discretos de pixels.

Todas essas transformações podem ser observadas de uma forma mais clara através da Figura 2.4.

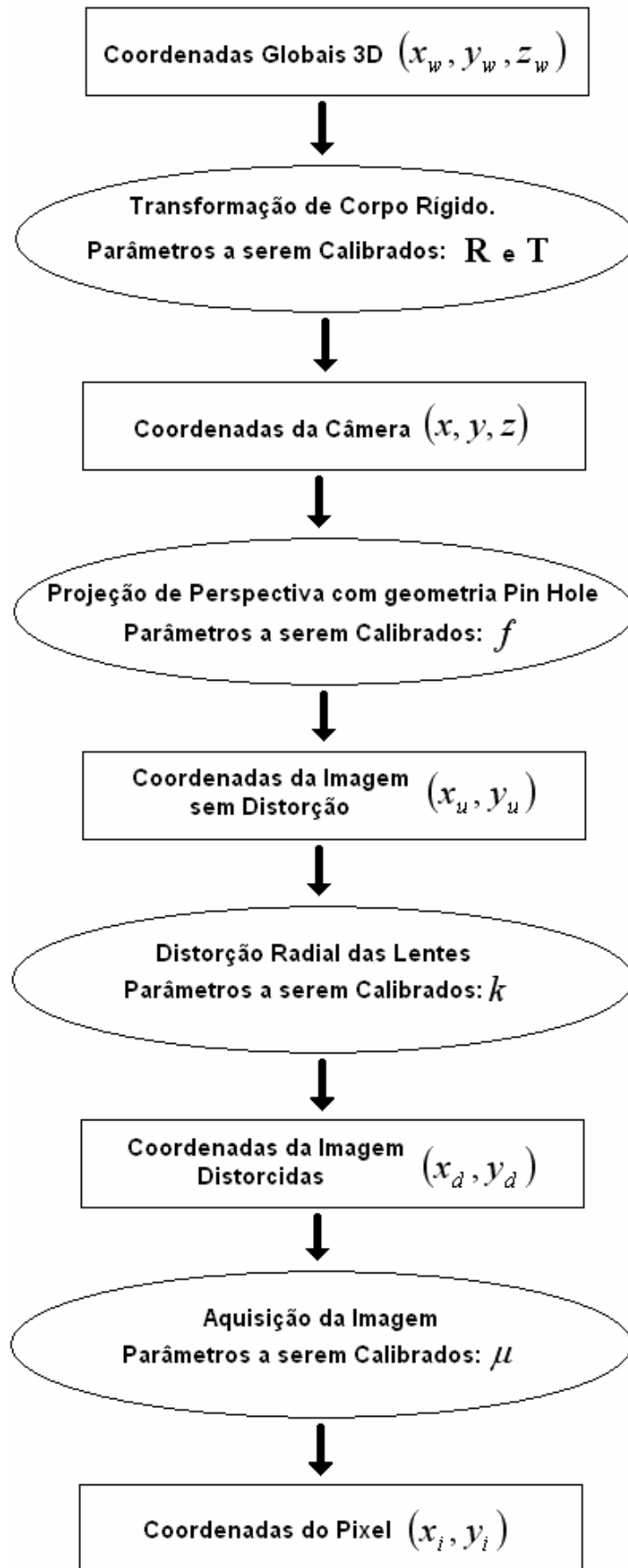


Figura 2.4. Transformações de Coordenadas do Modelo RAC.

A primeira transformação a ser realizada é a transformação de corpo rígido que altera o referencial do ponto P do sistema de coordenadas globais para o sistema de coordenadas da câmera. Essa transformação é realizada conforme ilustrado na Equação (2.2).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{T}, \quad (2.2)$$

onde R é uma matriz de rotação 3x3 definida como:

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \quad (2.3)$$

e T é um vetor de translação.

$$\mathbf{T} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (2.4)$$

Logo, nessa primeira transformação existem dois parâmetros envolvidos, a matriz R e o vetor T .

A transformação seguinte realiza a projeção das coordenadas 3D do ponto P em relação ao sistema de coordenadas da câmera (x, y, z) para as coordenadas do plano da imagem (x_u, y_u) , utilizando a projeção perspectiva e a geometria de câmera do tipo *pinhole*.

$$\begin{cases} x_u = f \frac{x}{z} \\ y_u = f \frac{y}{z} \end{cases} \quad (2.5)$$

O parâmetro a ser calibrado nessa etapa é a distância focal efetiva da câmera f .

A terceira etapa de transformação de coordenadas a ser realizada é a inclusão dos efeitos de distorção de lente. Essa distorção é modelada como sendo exclusivamente radial, sendo definida como:

$$\begin{cases} x_d + D_x = x_u \\ y_d + D_y = y_u \end{cases}, \quad (2.6)$$

onde (x_d, y_d) representa a coordenada real do ponto da imagem e D_x e D_y são os valores de distorção dados por: (Tsai, 1987, Zhuang e Roth, 1996)

$$\begin{cases} D_x = x_d(k_1 r^2 + k_2 r^4 + \dots) \\ D_y = y_d(k_1 r^2 + k_2 r^4 + \dots) \end{cases} \quad (2.7)$$

$$r = \sqrt{x_d^2 + y_d^2} \quad (2.8)$$

Entretanto, é possível desconsiderar os termos de ordem superior das equações de distorção sem nenhum prejuízo considerável de precisão (Sun e Cooperstock, 2005, Tsai, 1987, Zhuang e Roth, 1996), de forma que:

$$\begin{cases} D_x = k_1 x_d r^2 \\ D_y = k_1 y_d r^2 \end{cases} \quad (2.9)$$

O parâmetro a ser calibrado nessa etapa é o coeficiente de distorção k_1 .

A transformação final envolve a transformação das coordenadas (x_d, y_d) da projeção distorcida do ponto P para uma posição de memória.

$$\begin{cases} X_i = \frac{s_x}{dd_x} x_d + C_x \\ Y_i = -\frac{1}{dd_y} y_d + C_y \end{cases}, \quad (2.10)$$

onde:

$$\begin{aligned} (X_i, Y_i) &= \text{Coordenadas em Pixeis de um Ponto na Imagem} \\ (C_x, C_y) &= \text{Centro da Imagem} \\ (dd_x, dd_y) &= \text{Distância entre dois Elementos Sensores (Pixeis) Adjacentes} \\ s_x &= \text{Fator de Erros de Aquisição} \end{aligned} \quad (2.11)$$

O sinal negativo que multiplica o valor y_d é necessário devido ao fato do eixo Y do sistema de coordenadas da câmera ter o sentido invertido com relação ao do sistema de coordenadas utilizado normalmente por computadores para o armazenamento de imagens na memória conforme pode ser observado na Figura 2.5.



Figura 2.5. Sistema De coordenadas da Imagem e de Armazenamento no Computador.

Os fatores de escala (dd_x, dd_y) são os responsáveis pela transformação das coordenadas em pixels e são normalmente fornecidos pelos fabricantes de câmeras. Já o fator s_x é responsável pela inclusão no modelo dos efeitos causados na aquisição de imagens pelos sistemas de conversão A/D devido a pequenos desvios de sincronização e frequência entre os conversores e as câmeras (Tsai, 1987, Zhuang e Roth, 1996, Lenz e Tsai, 1987). Nessa etapa final de transformação, descrita pela Equação (2.10), o fator s_x é o parâmetro que deve ser calibrado.

Dentro do modelo RAC proposto por Tsai (1987), entretanto, existem somente dois parâmetros independentes no conjunto de parâmetros intrínsecos (s_x, dd_x, dd_y, f) . Utilizando-se dessa característica do modelo, Zhuang e Roth (1996) propõe uma definição diferente de parâmetros, onde a conversão de unidades para pixels é realizada já na segunda etapa de transformação. Nesse trabalho o modelo RAC modificado, introduzido por Zhuang e Roth (1996) foi utilizado.

2.1.4 O Modelo RAC Modificado

O modelo de câmera utilizado para o desenvolvimento do sistema VISSCAN-3D foi proposto por Motta (1999) e por Zhuang e Roth (1996). Utilizando-se de uma série de características do modelo RAC, Motta (1999) e Zhuang e Roth (1996) propuseram algumas alterações desse modelo.

O primeiro passo para a apresentação desse modelo é a definição das coordenadas da projeção distorcida de um ponto P qualquer:

$$\begin{cases} X_d = \frac{s_x}{dd_x} x_d = X_i - C_x \\ Y_d = \frac{1}{dd_y} y_d = -Y_i + C_y \end{cases}, \quad (2.12)$$

onde (X_d, Y_d) são as coordenadas de tela em pixels da projeção distorcida do ponto P com origem no centro (C_x, C_y) da imagem.

Definidas essas coordenadas, a primeira alteração introduzida no modelo, levando-se em consideração que dentro do conjunto de parâmetros intrínsecos dd_x , dd_y , s_x e f somente dois parâmetros são independentes, é a definição dos fatores de escala f_x e f_y conforme a Equação (2.13).

$$\begin{cases} f_x = f \frac{s_x}{dd_x} \\ f_y = f \frac{1}{dd_y} \end{cases} \quad (2.13)$$

A segunda alteração proposta está no cálculo da distância radial de cada ponto (Equação (2.8)). Enquanto no modelo RAC proposto por Tsai (1987) essa distância é calculada em unidades S.I. (mm), no modelo proposto por Motta (1999) e por Zhuang e Roth (1996) esse raio é calculado em pixels, de forma que:

$$r' = \sqrt{\left(\frac{dd_x}{s_x} X_d\right)^2 + (dd_y Y_d)^2} \quad (2.14)$$

Definindo-se então um novo coeficiente de distorção radial k e a razão μ entre os fatores de escala f_x e f_y , tem-se:

$$\begin{cases} k \equiv k_1 dd_y \\ \mu \equiv \frac{f_y}{f_x} = \frac{dd_x}{dd_y s_x} \end{cases} \quad (2.15)$$

$$r = \sqrt{\mu^2 X_d^2 + Y_d^2} \quad (2.16)$$

Logo, o modelo da câmera pode ser escrito como:

$$\begin{cases} X_d(1+kr^2) = f_x \frac{r_1x_w + r_2y_w + r_3z_w + T_x}{r_7x_w + r_8y_w + r_9z_w + T_z} \\ Y_d(1+kr^2) = f_y \frac{r_4x_w + r_5y_w + r_6z_w + T_y}{r_7x_w + r_8y_w + r_9z_w + T_z} \end{cases} \quad (2.17)$$

É nessa etapa que a modificação mais significativa é realizada sobre o modelo RAC. Utilizando-se da aproximação matemática (Kreyzig, 2006, Motta, 1999, Zhuang e Roth, 1996):

$$(1+kr^2) \approx \frac{1}{1-kr^2}, kr^2 \ll 1 \quad (2.18)$$

É possível então utilizar uma variante do modelo de distorção radial da lente conforme a Equação (2.19).

$$\begin{cases} \frac{X_d}{1-kr^2} = f_x \frac{r_1x_w + r_2y_w + r_3z_w + T_x}{r_7x_w + r_8y_w + r_9z_w + T_z} \\ \frac{Y_d}{1-kr^2} = f_y \frac{r_4x_w + r_5y_w + r_6z_w + T_y}{r_7x_w + r_8y_w + r_9z_w + T_z} \end{cases} \quad (2.19)$$

2.2 LENTES

Lentes são componentes ópticos com simetria axial que permitem a transmissão e a refração da luz, convergindo ou divergindo feixes.

Existe uma infinidade de tipos de lentes e de materiais que podem ser utilizados para a fabricação dessas, cada uma tendo diferentes aplicações e características, além da possibilidade de várias lentes simples para a construção de complexos sistemas compostos de lentes.

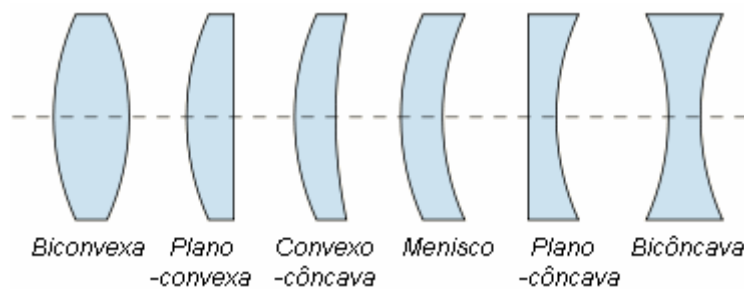


Figura 2.6. Alguns Tipos de Lentes Simples.

Lentes são largamente utilizadas em equipamentos de imageamento como microscópios, binóculos, câmeras, telescópios, teodolitos e muitos outros, e é exatamente no seu uso em imageamento que as lentes apresentam uma grande importância para a visão computacional.

2.3 FILTROS

Um filtro óptico é um instrumento que transmite seletivamente a luz. Ele bloqueia toda a luz que não apresenta determinadas características, transmitindo somente uma faixa do espectro.

O tipo mais comum de característica da luz utilizada para realizar essa seleção é seu comprimento de onda.

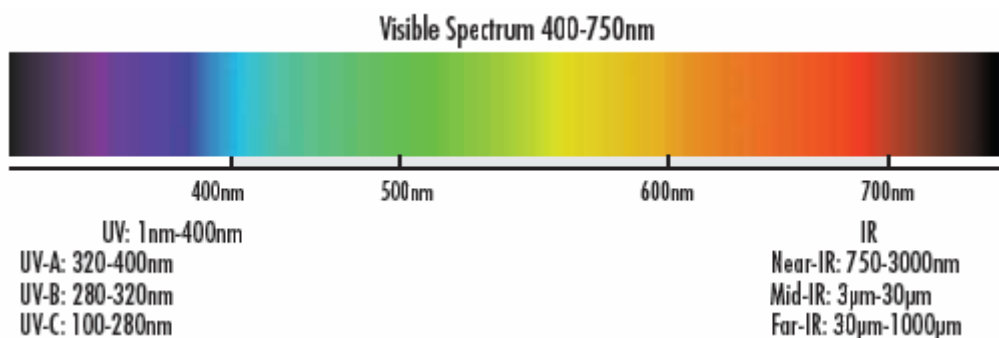


Figura 2.7. Espectro Visível e Comprimento de Onda.

Os filtros ópticos variam de acordo com o material de que são fabricados, com a forma e a faixa que bloqueiam o espectro e muitos outros parâmetros, entretanto os três tipos básicos de filtros são os filtros passa-baixas, passa-altas e passa-banda. Esses filtros são comumente referidos na literatura como filtros passa-curtas e filtros passa-longas devido à relação direta entre frequência e comprimento de onda para a radiação eletromagnética.

$$\text{Comprimento de Onda} = \frac{\text{Velocidade da Luz}}{\text{Frequência}} \quad (2.20)$$

Os filtros passa-longas são filtros que, dado um comprimento de onda de corte, todo o espectro eletromagnético que possuir um comprimento de onda inferior a esse comprimento de corte é bloqueado, conforme pode ser observado na Figura 2.8. Já os filtros passa-curtas bloqueiam todo o espectro eletromagnético que possuir um comprimento de onda superior a esse comprimento de corte e os filtros passa-banda, definidos dois comprimentos de onda de corte, somente o espectro que possuir comprimento de onda entre esses dois comprimentos de corte é transmitido.

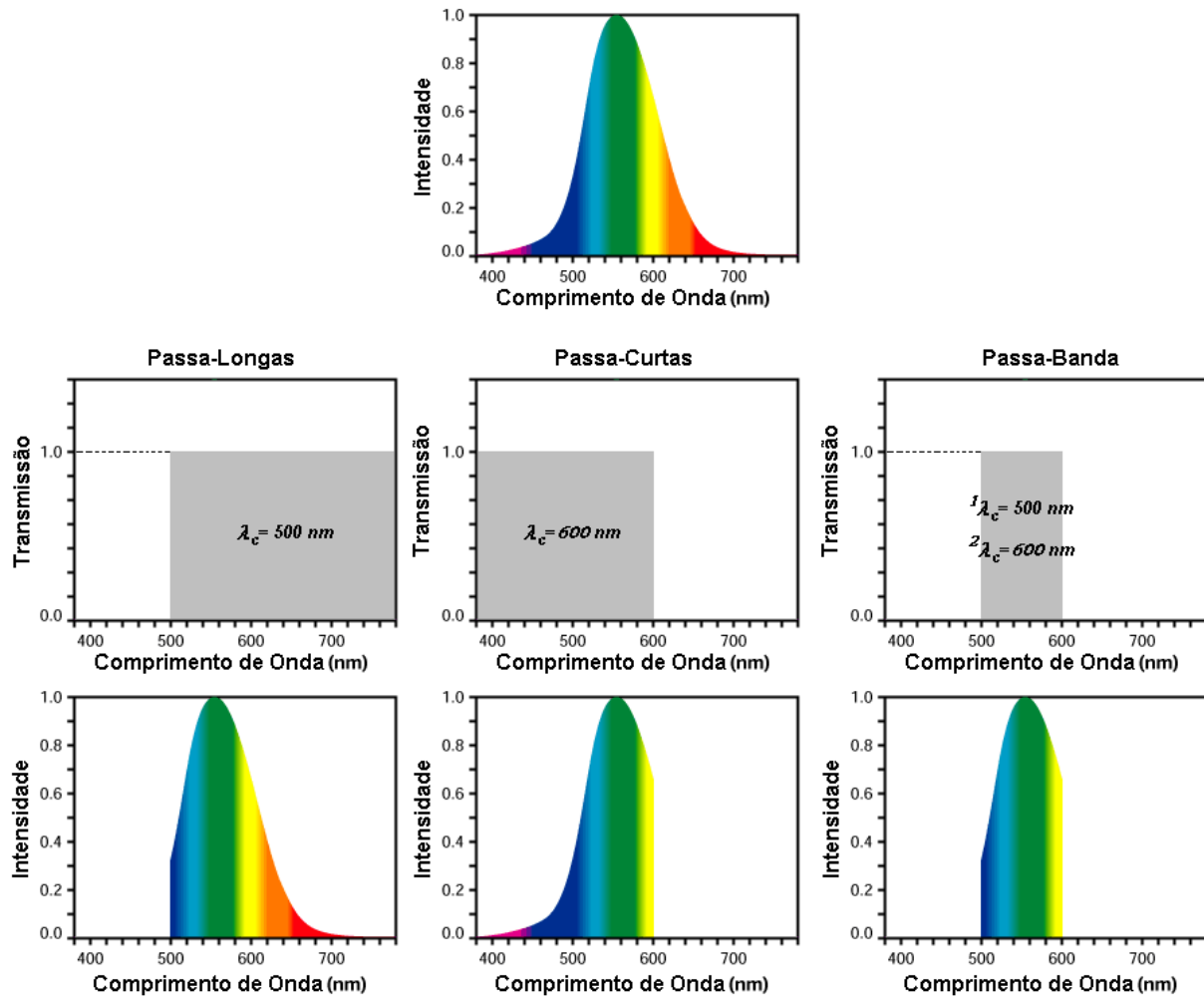


Figura 2.8. Tipos Básicos de Filtros Ópticos Ideais e seu Efeito.

Os filtros ópticos são utilizados principalmente para melhora do contraste de imagens e para a remoção de comprimentos de onda indesejados, facilitando o processamento das imagens capturadas.

2.4 LASERS

Um laser, ou amplificador de luz por emissão estimulada de radiação (*Light Amplification by Stimulated Emission of Radiation*), é um dispositivo óptico que produz um tipo bem definido de radiação eletromagnética. Ele produz uma radiação monocromática, coerente e colimada, ou seja, ele produz uma radiação que possui uma frequência bem definida, relações de fase bem definidas e que se propaga como um feixe.

De uma forma simplificada um laser pode ser visto como dois espelhos paralelos, um deles semi-transparente, e um meio ativo gasoso ou semi-condutor entre eles. Ao energizar-se esse meio, a radiação oscila entre os espelhos até acumular energia suficiente para atravessar o espelho semi-transparente, emitindo um feixe de energia eletromagnética.

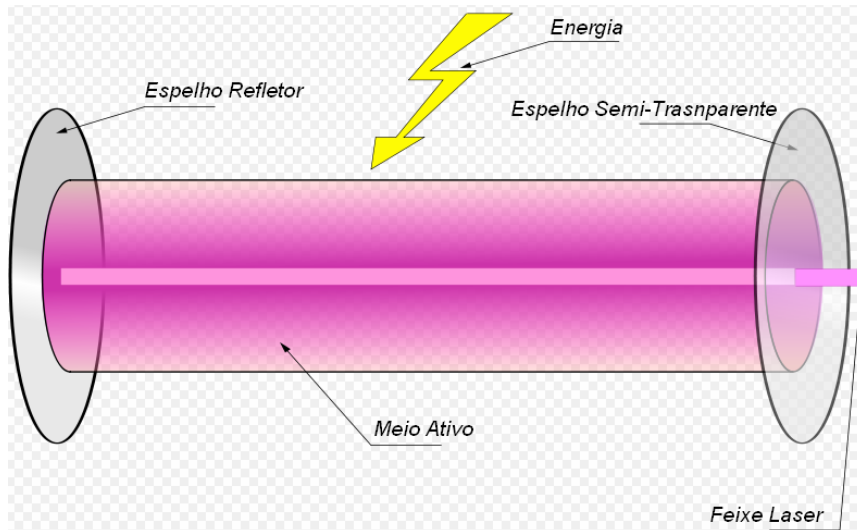


Figura 2.9. Princípio de Funcionamento de um Diodo Laser.

O laser é utilizado em várias aplicações na área médica, na área industrial, na pesquisa científica e em uma infinidade de equipamentos e tecnologias comerciais como os aparelhos de CD e DVD, os leitores de códigos de barras, as fibras ópticas e muitos outros.

Apesar de lasers emitirem um único feixe de energia eletromagnética, é possível, através do uso de elementos ópticos, transformar esse feixe em uma projeção de formas mais complexas, como linhas ou matrizes de pontos, ampliando ainda mais a gama de aplicações desses dispositivos.

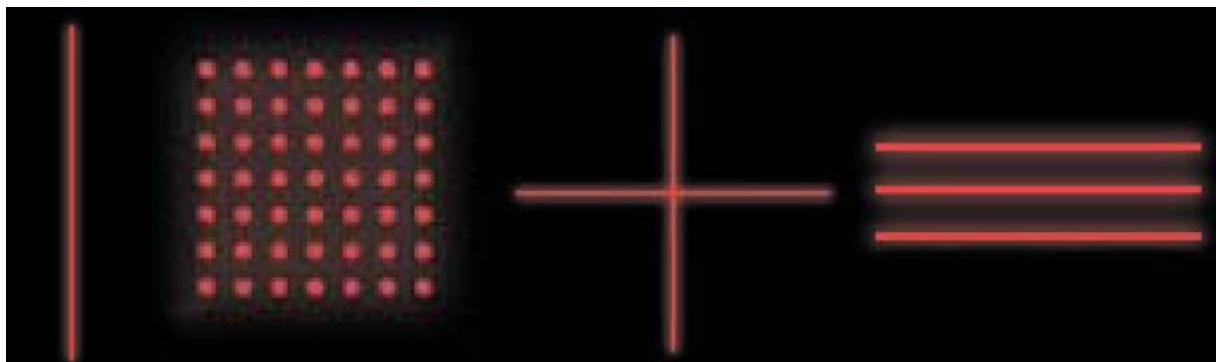


Figura 2.10. Alguns Padrões de Projeção de Lasers.

3 O SISTEMA VISSCAN-3D

Uma das principais vantagens do sistema de digitalização de superfícies VISSCAN-3D está no fato dele não depender de sensores de posicionamento para medir o deslocamento angular da fonte laser. Entretanto, a determinação desse deslocamento angular é necessária para a obtenção do mapa de profundidade da superfície a ser digitalizada. A idéia básica para realizar essa medição está na substituição desse sensor por uma fonte laser, sendo o sistema como um todo composto por dois diodos lasers e uma câmera conforme ilustrado no esquemático simplificado na Figura 3.1.

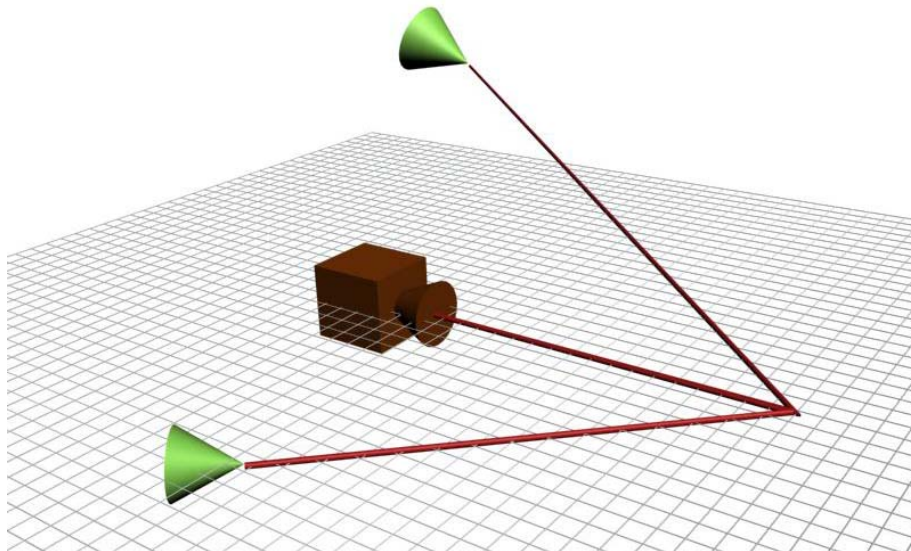


Figura 3.1. Esquemático do Sistema VISSCAN-3D.

Além do uso de um diodo laser adicional, para tornar a triangulação possível, ainda é necessário impor duas restrições de montagem ao sistema.

A primeira restrição é relacionada à posição relativa entre os dois planos de luz projetados pelos lasers. Eles devem ser perpendiculares para que as equações de triangulação apresentadas na Seção 3.1 sejam válidas. A segunda restrição impõe que um dos diodos lasers é fixo, não possuindo nenhum grau de liberdade enquanto o outro diodo possui somente uma movimentação angular.

A razão de cada uma dessas restrições e as implicações do não cumprimento dessas são detalhadamente explicadas na Seção 3.1.1.

3.1 O SISTEMA ÓPTICO

Devido às restrições de montagem do sistema é possível definir o conjunto de parâmetros necessários para a triangulação e a reconstrução do mapa de profundidade da superfície digitalizada.

Cada um dos diodos lasers gera um plano de luz que, projetado sobre uma superfície qualquer, gera uma curva sobre essa superfície, conforme ilustrado na Figura 3.2.

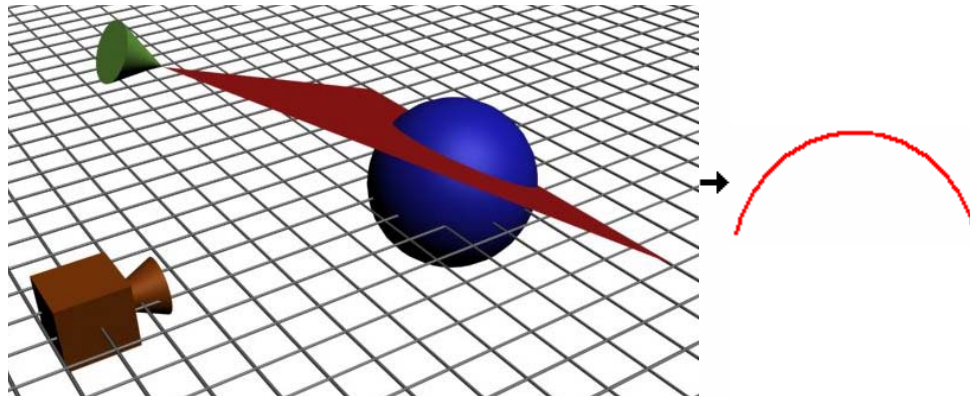


Figura 3.2. Projeção de um Plano de Luz Laser sobre uma Superfície.

Considerando, sem perda de generalidade, que o plano de luz que gera a linha laser ilustrada na Figura 3.2 é paralelo ao eixo-X do sistema de coordenadas da câmera, e realizando um corte de visualização no plano ZY desse sistema de coordenadas, é possível escolher um único ponto P da linha laser e analisar a formação da imagem desse ponto utilizando o modelo de perspectiva central.

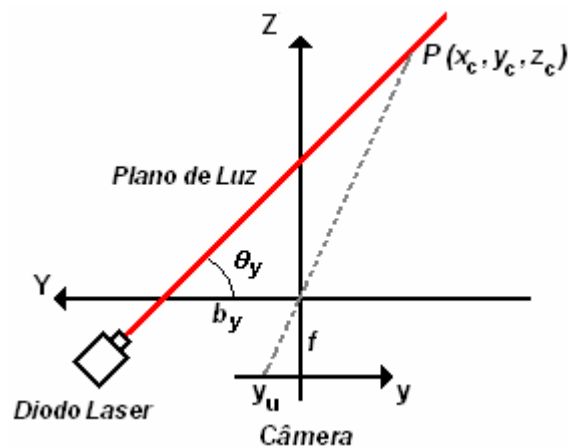


Figura 3.3. O Primeiro Plano de Luz e a Geração da Imagem de um Ponto da Linha Laser.

Observando esse corte, tem-se dois triângulos, através dos quais é possível extrair uma relação entre as coordenadas tridimensionais do ponto P e as coordenadas bidimensionais de sua imagem.

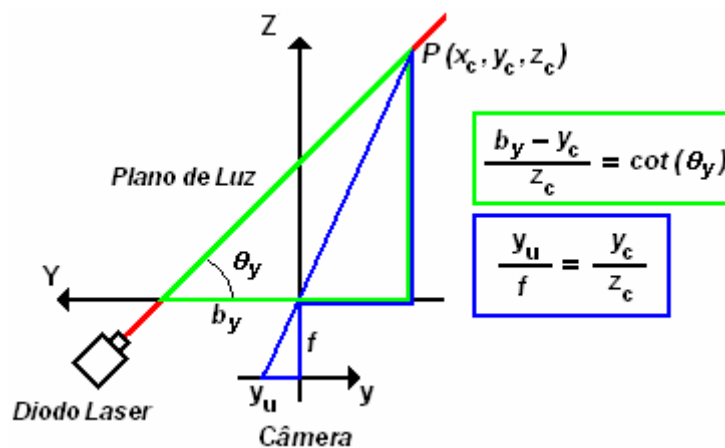


Figura 3.4. Triangulação através da Projeção do Laser na Superfície.

Logo,

$$\begin{cases} y_c = \frac{y_u b_y}{f \cot(\theta_y) + y_u} \\ z_c = \frac{f b_y}{f \cot(\theta_y) + y_u} \end{cases}, \quad (3.1)$$

onde:

- $(x_c, y_c, z_c) \rightarrow$ Coordenadas 3D do ponto P em relação à câmera em unidades S.I.;
- $(x_u, y_u) \rightarrow$ Coordenadas da imagem do ponto P em unidades S.I.;
- $b_y \rightarrow$ Distância base no eixo-Y entre a câmera e o plano laser paralelo ao eixo-X.
- $\theta_y \rightarrow$ Ângulo entre o plano laser paralelo ao eixo-X e o eixo-Y.
- $f \rightarrow$ Distância focal da câmera.

Utilizando-se então as equações de perspectiva (Equação (2.1)) é possível determinar-se ainda o valor de x_c como:

$$x_c = \frac{x_u b_y}{f \cot(\theta_y) + y_u}, \quad (3.2)$$

de forma que as coordenadas tridimensionais do ponto P ficam completamente definidas pelas coordenadas bidimensionais da imagem desse ponto através da Equação (3.3).

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{b_y}{f \cot(\theta_y) + y_u} \begin{bmatrix} x_u \\ y_u \\ f \end{bmatrix} \quad (3.3)$$

Sabendo-se então que, devido às restrições de montagem do sistema, ambos os planos lasers gerados são perpendiculares entre si, tem-se que o segundo plano laser gerado será paralelo ao eixo-Y da câmera. Realizando-se novamente os procedimentos utilizados para a derivação das equações da primeira linha laser, um corte no plano XZ é feito, e a formação da imagem de um único ponto P dessa linha é analisado sobre a ótica do modelo de perspectiva.

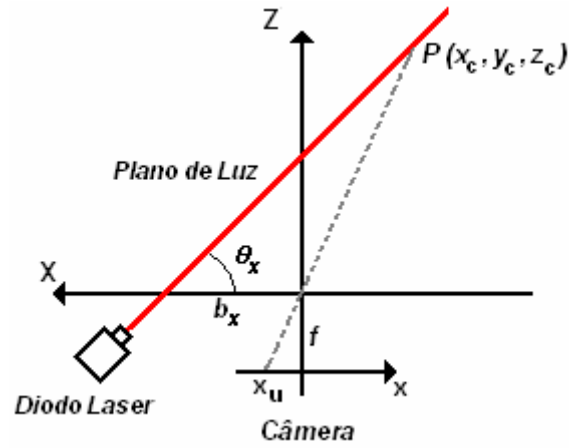


Figura 3.5. O Segundo Plano de Luz e a Geração da Imagem de um Ponto da Linha Laser.

Logo, de maneira similar à ilustrada na Figura 3.4, por meio de triangulação, tem-se:

$$\begin{cases} x_c = \frac{x_u b_x}{f \cot(\theta_x) + x_u}, \\ z_c = \frac{f b_x}{f \cot(\theta_x) + x_u} \end{cases}, \quad (3.4)$$

onde:

- $(x_c, y_c, z_c) \rightarrow$ Coordenadas 3D do ponto P em relação à câmera em unidades S.I.;
- $(x_u, y_u) \rightarrow$ Coordenadas da imagem do ponto P em unidades S.I.;
- $b_x \rightarrow$ Distância base no eixo-X entre a câmera e o plano laser paralelo ao eixo-Y;
- $\theta_x \rightarrow$ Ângulo entre o plano laser paralelo ao eixo-Y e o eixo-X;
- $f \rightarrow$ Distância focal da câmera;

E, através das equações de perspectiva:

$$y_c = \frac{y_u b_x}{f \cot(\theta_x) + x_u}, \quad (3.5)$$

de forma que as coordenadas tridimensionais do ponto P ficam completamente definidas pelas coordenadas bidimensionais da imagem desse ponto através da Equação (3.6).

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{b_x}{f \cot(\theta_x) + x_u} \begin{bmatrix} x_u \\ y_u \\ f \end{bmatrix} \quad (3.6)$$

A Equação (3.3) e a Equação (3.6) definem uma relação direta entre as coordenadas tridimensionais de um ponto P e as coordenadas bidimensionais de sua imagem, mas essas equações não são válidas para quaisquer pontos da imagem. A Equação (3.3) é válida somente para uma das linhas lasers, e a Equação (3.6) é válida somente para a outra linha laser.

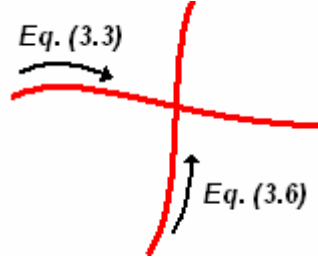


Figura 3.6. Validade das Equações das Linhas Lasers.

Entretanto, existe um ponto onde ambas as equações são válidas, o ponto de interseção P_{int} entre as duas linhas lasers projetadas na superfície.

Extraindo-se da imagem as coordenadas (x_{int}, y_{int}) desse ponto de interseção e sabendo que as coordenadas tridimensionais desse ponto são as mesmas independente de qual equação é utilizada para seu cálculo, é possível obter uma relação entre os ângulos dos dois diodos laser θ_x e θ_y a partir das Equação (3.3) e da Equação (3.6):

$$\frac{b_x}{f \cot(\theta_x) + x_{int}} = \frac{b_y}{f \cot(\theta_y) + y_{int}} \Rightarrow \begin{cases} \cot(\theta_y) = \frac{1}{f} \left[\frac{b_y}{b_x} (f \cot(\theta_x) + x_{int}) - y_{int} \right] \\ \cot(\theta_x) = \frac{1}{f} \left[\frac{b_x}{b_y} (f \cot(\theta_y) + y_{int}) - x_{int} \right] \end{cases} \quad (3.7)$$

Considerando-se novamente as restrições de montagem do sistema, sabe-se que um dos diodos laser não possui nenhum grau de movimentação, de forma que o valor de $\cot(\theta_x)$ ou de $\cot(\theta_y)$ é constante e conhecido, assim como os valores de b_x , b_y e f que são calibrados previamente. Logo, o valor da cotangente do diodo laser móvel pode ser obtido através da Equação (3.7) e a Equação (3.3) ou a Equação (3.6) pode ser utilizada para, a partir das coordenadas bidimensionais da imagem, determinarem-se as coordenadas tridimensionais de cada um dos pontos que pertencem à linha projetada na superfície pelo diodo laser móvel.

Entretanto, ao rotacionar-se o diodo móvel observa-se que o modelo descrito pela Equação (3.3), Equação (3.6) e Equação (3.7) não consegue descrever completamente a geometria do sistema. Conforme pode ser observado na Figura 3.3 e na Figura 3.5 as distâncias de base b_x e b_y são

consideradas constantes para a derivação das equações de digitalização (Equação (3.3), Equação (3.6) e Equação (3.7)). Entretanto, conforme pode ser observado na Figura 3.7, caso o diodo laser móvel não esteja alinhado com o sistema de coordenadas da câmera, sua distância base não se mantém constante durante a varredura da superfície.

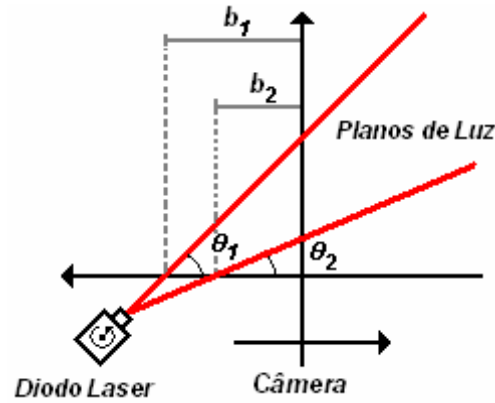


Figura 3.7. Efeito do Desalinhamento do Laser Móvel.

Para considerar-se esse efeito nas equações de digitalização é necessária a inclusão de um parâmetro de desalinhamento d , sendo então possível a realização de uma correção sobre a distância base do laser para cada posição angular conforme a Equação (3.8).

$$\begin{cases} b_y' = b_y + d_y \cot(\theta_y) \\ b_x' = b_x + d_x \cot(\theta_x) \end{cases} \quad (3.8)$$

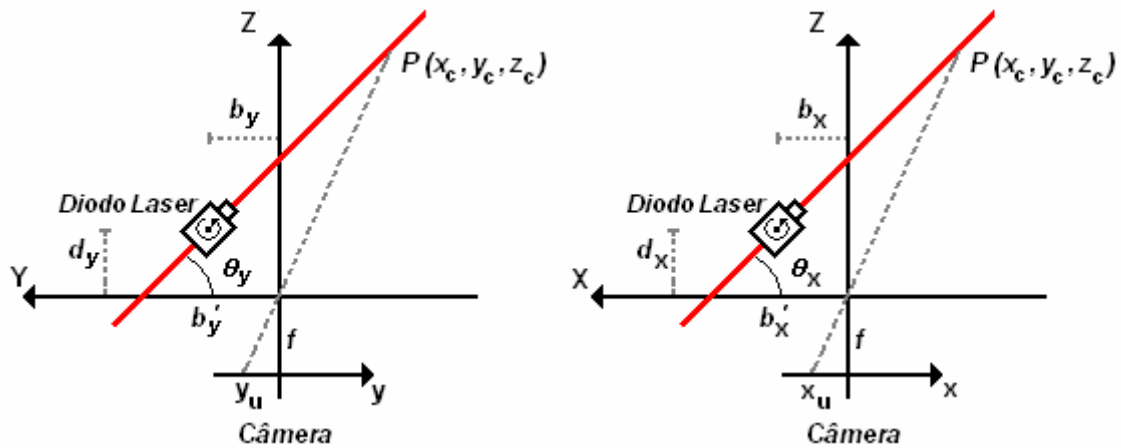


Figura 3.8. Parâmetros de Desalinhamento.

É importante observar que, apesar desse desalinhamento poder ocorrer em ambos os diodos, ele gera uma variação somente da distância base do feixe laser móvel. Para o laser fixo, independente do desalinhamento, a distância base b' permanece constante. Ou seja, após a determinação dessa distância, não é necessário realizar-se nenhuma compensação devido à variação da posição do feixe móvel.

Reescrevendo as equações de digitalização, incluindo o efeito do desalinhamento do laser móvel:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{b_y'}{f \cot(\theta_y) + y_u} \begin{bmatrix} x_u \\ y_u \\ f \end{bmatrix} \quad (3.9)$$

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{b_x'}{f \cot(\theta_x) + x_u} \begin{bmatrix} x_u \\ y_u \\ f \end{bmatrix}, \quad (3.10)$$

onde b_x' e b_y' são dados pela Equação (3.8), e:

$$\begin{cases} \cot(\theta_y) = \frac{1}{f} \left[\frac{b_y'}{b_x} (f \cot(\theta_x) + x_{int}) - y_{int} \right] \left[1 - \frac{d_y}{fb_x} (f \cot(\theta_x) + x_{int}) \right]^{-1} \\ \cot(\theta_x) = \frac{1}{f} \left[\frac{b_x'}{b_y} (f \cot(\theta_y) + y_{int}) - x_{int} \right] \left[1 - \frac{d_x}{fb_y} (f \cot(\theta_y) + y_{int}) \right]^{-1} \end{cases} \quad (3.11)$$

Logo, é possível definir cada um dos passos necessários para a digitalização completa de uma superfície e descrevê-los através de um fluxograma. É importante observar que, o modelo da câmera e os parâmetros b_x , b_y , d_x , d_y , $\cot(\theta_x)$ e $\cot(\theta_y)$ são calibrados previamente e que, dependendo de qual diodo laser é utilizado como o laser móvel, a equação a ser utilizada para o cálculo das coordenadas tridimensionais será a Equação (3.9) ou a Equação (3.10) e o conjunto de parâmetros que devem ser calibrados é alterado.

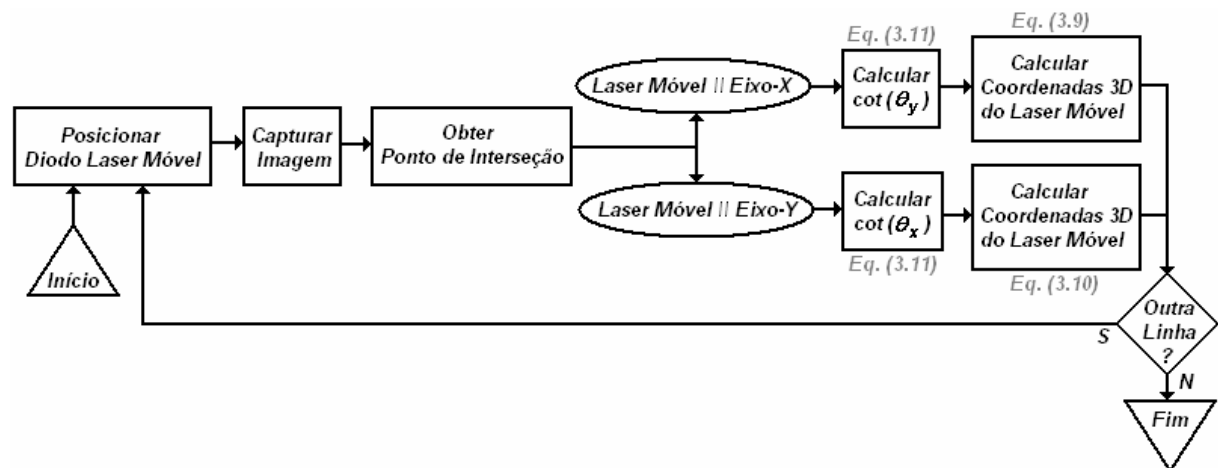


Figura 3.9. Algoritmo de Digitalização de Superfícies.

Esses procedimentos constituem a parte central do sistema de digitalização proposto e são a base de todo o desenvolvimento apresentado aqui.

3.1.1 As Restrições de Montagem do Sistema

As restrições de montagem do sistema são condições necessárias para que os cortes ilustrados na Figura 3.3 e na Figura 3.5 sejam válidos, possibilitando a realização do processo de digitalização apresentado na Seção 3.1. São duas essas restrições:

- Os feixes lasers projetados devem ser perpendiculares entre si;
- Somente um dos feixes lasers é móvel.

A restrição de mobilidade imposta a um dos diodos é necessária para tornar possível a determinação da posição do diodo móvel sem a necessidade de um sensor de posição rotacional. O cálculo dessa posição angular pode ser feito utilizando-se a Equação (3.11), os parâmetros do sistema (b_x , b_y , d_x , d_y), o valor de posição de um dos diodos laser ($\cot(\theta_x)$ ou $\cot(\theta_y)$) e as coordenadas do ponto de interseção (x_{int} , y_{int}) entre os lasers conforme ilustrado na Figura 3.10.

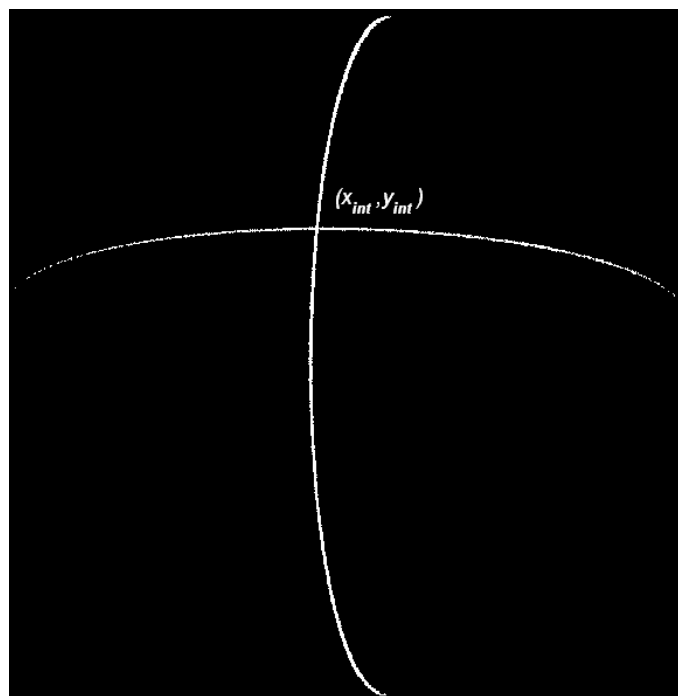


Figura 3.10. Ponto de Cruzamento dos Feixes Lasers.

Entretanto, se ambos os diodos tivessem liberdade de movimento, ambos os valores de cotangente seriam incógnitas, e, não seria possível a determinação desses valores diretamente através da Equação (3.11).

Já a restrição de perpendicularidade entre ambos os feixes lasers projetados também é necessária para que os cortes ilustrados na Figura 3.3 e na Figura 3.5 sejam válidos. Conforme apresentado anteriormente, é possível, sem perda de generalidade, assumir que um dos feixes lasers seja paralelo ao eixo-X da câmera. Supondo então que o segundo feixe, ao invés de perpendicular, tenha uma

inclinação α em relação ao primeiro diodo laser, o corte da Figura 3.5 não é mais válido, pois para diferentes pontos P pertencentes ao plano de luz laser, serão obtidos diferentes cortes.

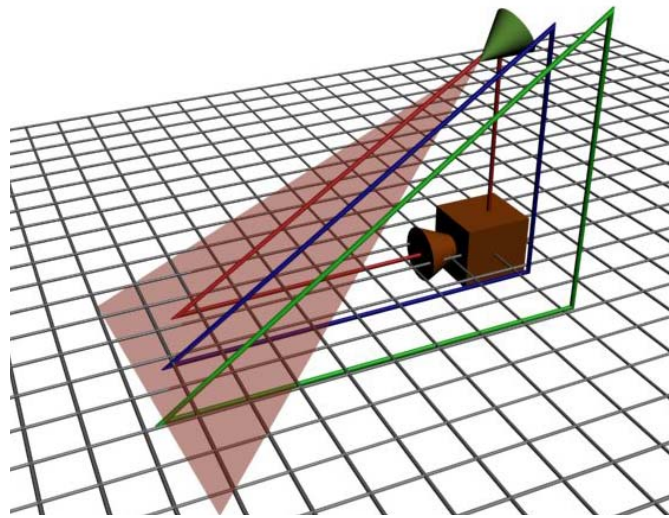


Figura 3.11. Efeito da Não-Perpendicularidade dos Feixes Lasers.

Isso significa que, ao realizar-se um corte em um plano paralelo ao eixo-Y não serão obtidos somente um único triângulo e uma distância base como ilustrado na Figura 3.5, mas sim infinitos triângulos e distâncias base conforme ilustrado na Figura 3.11, onde a distância base irá depender da posição do ponto P no espaço.

3.2 O SISTEMA ELETRO-MECÂNICO

Conhecidos os fundamentos básicos que regem o sistema óptico de triangulação ativa e as restrições que devem ser impostas sobre a posição e movimentação dos diodos laser é possível definir-se o sistema eletro-mecânico correspondente. Para o desenvolvimento do sistema VISSCAN-3D foram criadas duas versões diferentes de hardware.

A versão inicial do sistema foi desenvolvida para uso em laboratório. Essa versão fornece acesso a cada uma das variáveis descritas na Seção 3.1 e é possível modificá-las com facilidade, de forma a tornar possível a avaliação experimental de diferentes configurações, oferecendo uma grande flexibilidade ao sistema.

Já a segunda versão do sistema foi desenvolvida para uso em campo, tendo como principais características a robustez, o tamanho e peso e a precisão, de forma que o hardware do sistema é completamente protegido por um invólucro, tornando possível seu uso em ambientes agressivos.

A seguir, cada um desses sistemas é apresentado.

3.2.1 Versão de Laboratório

A versão de laboratório do sistema VISSCAN-3D foi criada com a finalidade de tornar possível a realização de vários experimentos com diferentes configurações e arranjos de parâmetros. Na Figura 3.12 é possível observar a estrutura implementada.

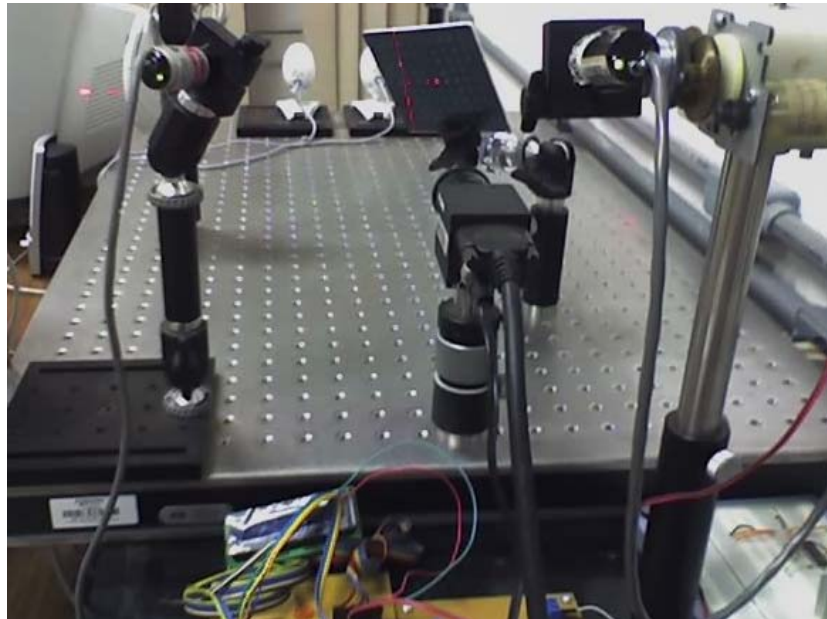


Figura 3.12. Versão de Laboratório do Hardware do Sistema VISSCAN-3D.

O sistema foi construído utilizando-se estruturas disponíveis comercialmente para a montagem de sistemas ópticos, e na Tabela 3.1 podem ser observados os valores de cada um dos parâmetros físicos dessa montagem. Esses parâmetros foram calibrados utilizando as técnicas apresentadas na Seção 5.2.

Tabela 3.1. Parâmetros Físicos do Sistema VISSCAN-3D Versão de Laboratório.

Parâmetro	Valor
b_y	165 mm
d_y	-24 mm
b_x'	146 mm
θ_x	58.5°

3.2.2 Versão de Campo

A versão de campo do sistema VISSCAN-3D, ao contrário da versão de laboratório, foi criada com a finalidade de não permitir a variação dos parâmetros do sistema, fornecendo uma estrutura rígida que ofereça proteção às partes sensíveis do sistema enquanto mantém peso e tamanho mínimos de forma a facilitar a integração com outros sistemas. Na Figura 3.13 é possível observar o *hardware* implementado.

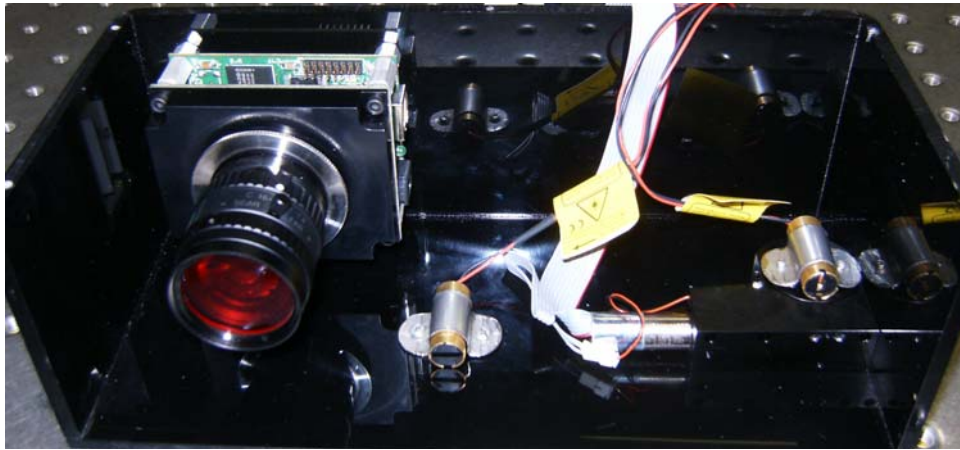


Figura 3.13. Versão de Campo do Hardware do Sistema VISSCAN-3D.

Assim como a versão de laboratório, a versão de campo também foi construída utilizando-se estruturas disponíveis comercialmente, o que torna a montagem do sistema mais rápida, reduzindo o tempo entre o projeto e a produção final e facilitando a produção em escala. A estrutura básica do sistema para o desenvolvimento de um protótipo inicial apresentou um custo aproximado de US\$ 5000,00 e um peso de 300 g distribuídos entre cada um dos principais componentes conforme ilustrado na Figura 3.14 e na Figura 3.15.

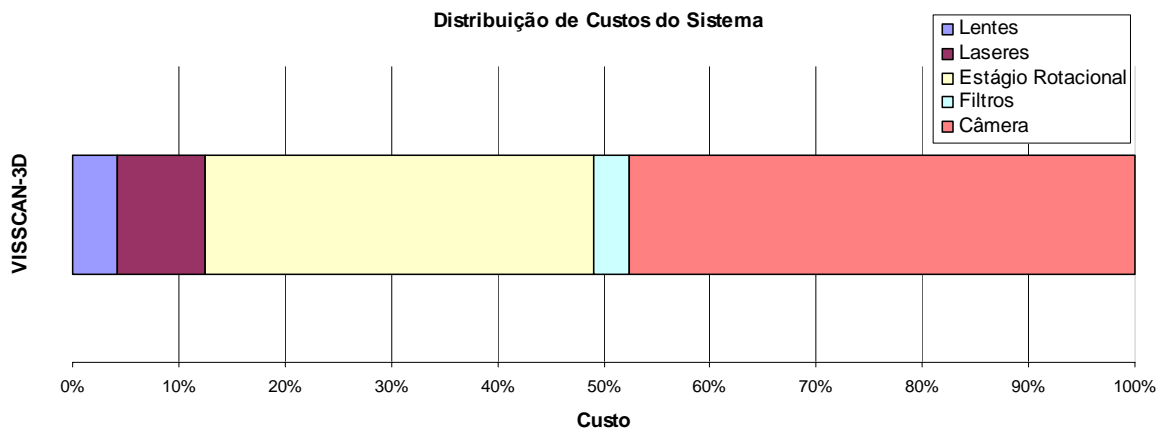


Figura 3.14. Distribuição do Custo do Sistema.

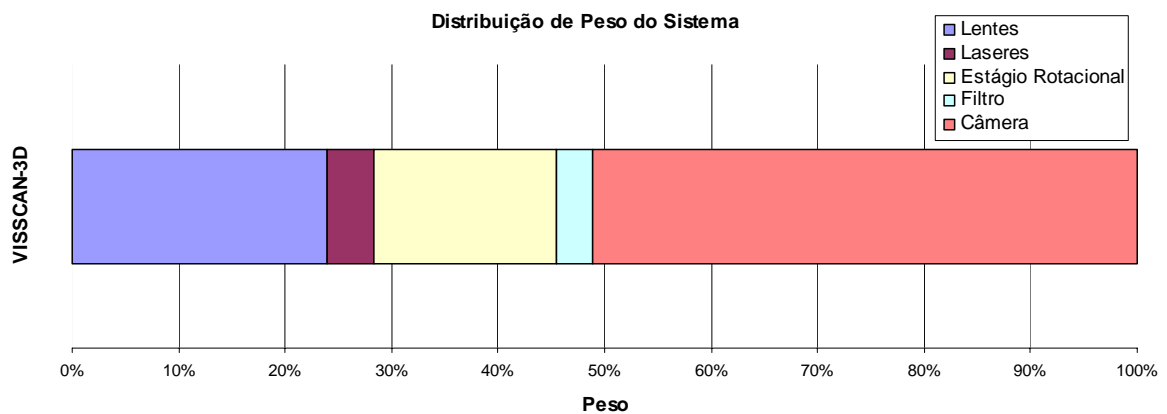


Figura 3.15. Distribuição do Peso do Sistema.

A interface de câmera escolhida foi o padrão USB. Esta escolha deveu-se principalmente ao fato do padrão USB ser um padrão amplamente difundido e utilizado e, ao contrário do padrão CameraLink utilizado na versão de laboratório, não necessita de uma placa de aquisição, o que torna possível a integração da versão de campo do sistema VISSCAN-3D à qualquer microcomputador que possua uma interface padrão USB.

Na Tabela 3.2 podem ser observados os valores de cada um dos parâmetros físicos dessa montagem. Esses parâmetros foram determinados durante o projeto mecânico do sistema e foram, após a montagem, calibrados utilizando as técnicas apresentadas na Seção 5.2.

Tabela 3.2. Parâmetros Físicos do Sistema VISSCAN-3D Versão de Campo.

Parâmetro	Valor
b_y	<i>140 mm</i>
d_y	<i>-10 mm</i>
b_x'	<i>30 mm</i>
θ_x	<i>90.0°</i>

3.3 O SISTEMA DE CALIBRAÇÃO

O sistema de calibração tem como objetivo a determinação de todos os parâmetros do sistema descritos na Seção 3.1. A calibração envolve o uso de uma placa de calibração, conforme ilustrado na Figura 3.16 e o uso do Algoritmo de Restrição de Alinhamento Radial para a calibração da câmera.

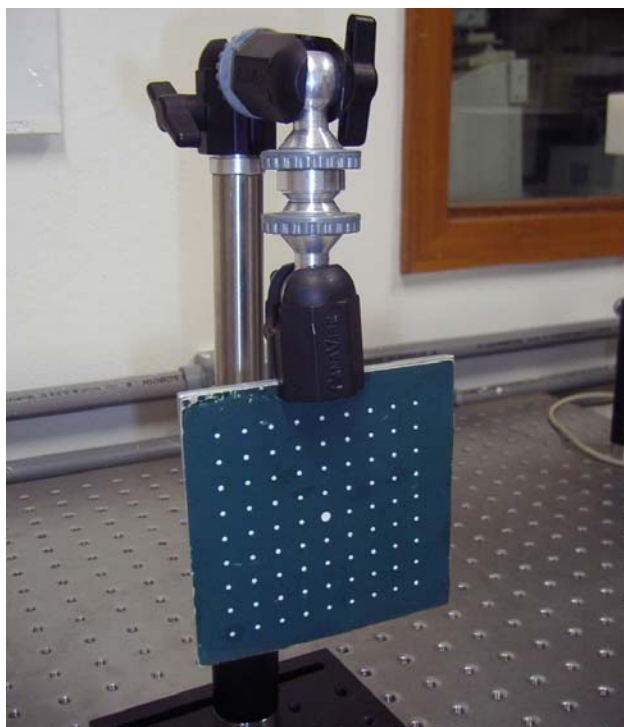


Figura 3.16. Placa de Calibração.

O procedimento de calibração do sistema VISSCAN-3D pode ser dividido em duas etapas principais. A primeira etapa é a calibração da câmera utilizando o algoritmo RAC e a segunda é a calibração dos diodos lasers, onde os parâmetros $\cot(\theta_x)$, $\cot(\theta_y)$, d_x , d_y , b_x e b_y são determinados a partir da projeção dos lasers sobre a placa de calibração.

A calibração do sistema é apresentada detalhadamente no Capítulo 5.

3.4 O SISTEMA COMPUTACIONAL

Visando tornar o processo de digitalização fácil e amigável a um usuário do sistema, dois *softwares* principais foram desenvolvidos e uma série de sistemas periféricos de apoio, sendo todo o projeto desenvolvido em linguagem C/C++. A criação dos ambientes de visualização 3D foi realizada utilizando-se as bibliotecas de programação OPEN-GL e para o desenvolvimento das interfaces de usuário e de toda a codificação necessária foi utilizado o compilador Borland C++ Builder 6.

Os dois principais *softwares* do pacote computacional desenvolvido são:

- Sistema de Calibração de Câmeras e Lasers;
- Sistema de Digitalização-3D de Superfícies;

Já os *softwares* de apoio são:

- *Software* de Processamento de Imagens;
- *Software* de Simulação de Câmeras e Lasers;
- *Software* de Geração de Superfícies Sintéticas;

Dentre os *softwares* principais, o sistema de calibração é responsável pela calibração de todas as características do sistema, incluindo não apenas as características físicas do sistema, mas também as características da câmera. Já sistema de digitalização é responsável pelo processo de digitalização de superfícies. Na Figura 3.17 e na Figura 3.18 é possível observar as janelas principais de cada um desses *softwares*

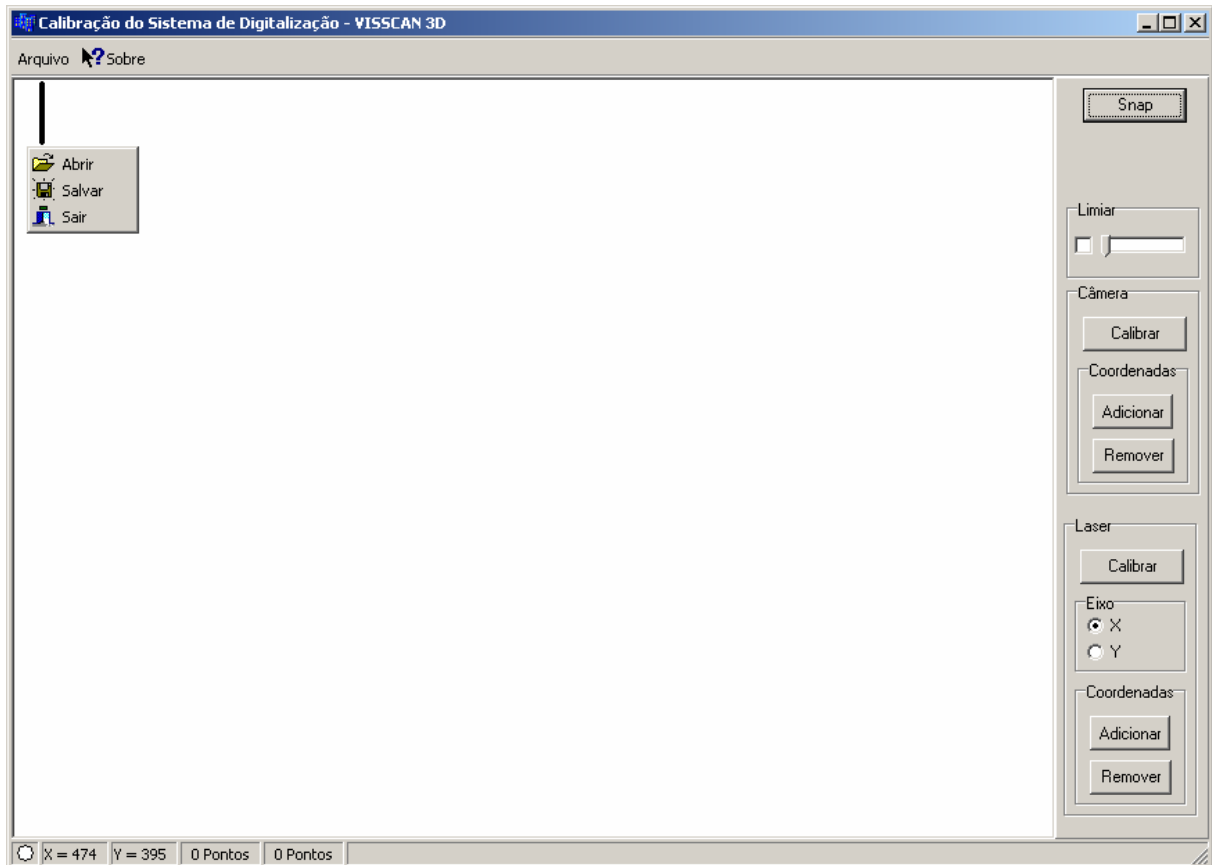


Figura 3.17. Tela Inicial do Software de Calibração.

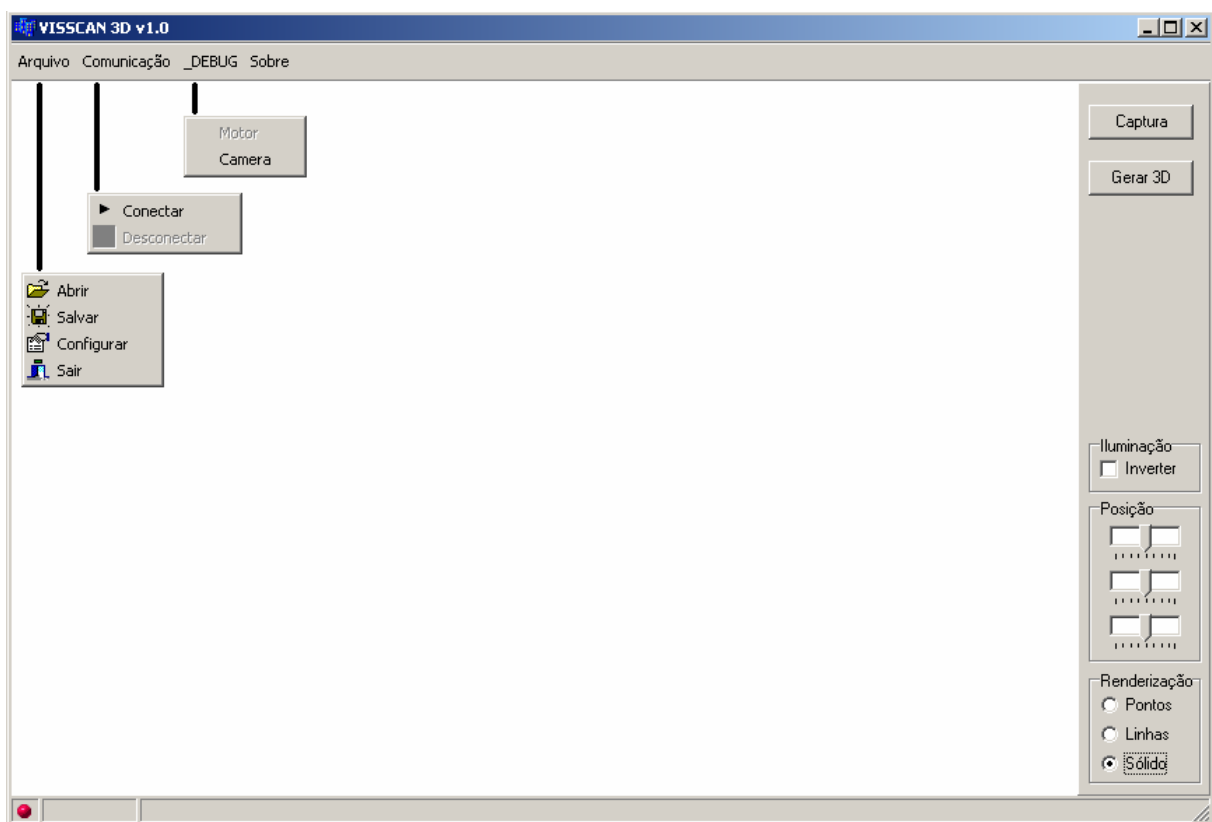


Figura 3.18. Tela Inicial do Software de Digitalização.

4 SOFTWARE E PROGRAMAÇÃO

4.1 PROCESSAMENTO DE IMAGENS

O processamento de imagens é uma vasta área de pesquisa e desenvolvimento que tem como principal objetivo a melhora de imagens para a percepção humana e a manipulação de imagens para a armazenagem, transmissão e representação dessas (Gonzales e Woods, 2002, Trucco e Verri, 1998).

Atualmente, o processamento de imagens está presente em quase todas as áreas tecnológicas, sendo utilizado nas mais diversas aplicações e não apenas no espectro visível da luz (Gonzales e Woods, 2002). Algumas dessas áreas são o imageamento médico, o imageamento meteorológico, o monitoramento de áreas urbanas, a delimitação de terras agrícolas, a inspeção e controle de qualidade industrial, segurança, aplicações espaciais e muitas outras (Gonzales e Woods, 2002, Trucco e Verri, 1998, Parker, 1997).

Apesar do principal objetivo da visão computacional ser o mundo real tridimensional, enquanto o processamento de imagens se foca nas próprias imagens, ambas essas grandes áreas do conhecimento apresentam muito em comum. Grande parte das técnicas e procedimentos de visão computacional necessitam do processamento das imagens utilizadas (Trucco e Verri, 1998, Klette *et.al.*, 1996). E no caso particular do sistema de digitalização VISSCAN-3D não é diferente. O sistema utiliza uma grande variedade de técnicas de processamento de imagens visando a extração de informações destas que possibilitem a reconstrução de superfícies. No Apêndice D cada uma dessas funções de processamento e suas implementações são apresentadas detalhadamente.

A principal necessidade do sistema VISSCAN-3D no que se refere ao processamento de imagens são quatro tarefas. Duas referentes à calibração do sistema e duas referentes ao processo de digitalização tridimensional.

- Extração das coordenadas dos pontos da placa de calibração do sistema (Figura 4.1);
- Extração de pontos da linha laser projetada sobre a placa de calibração (Figura 4.1);
- Extração do ponto de interseção das linhas lasers projetadas em uma superfície (Figura 4.2);
- Extração de uma linha laser projetada sobre uma superfície (Figura 4.2);

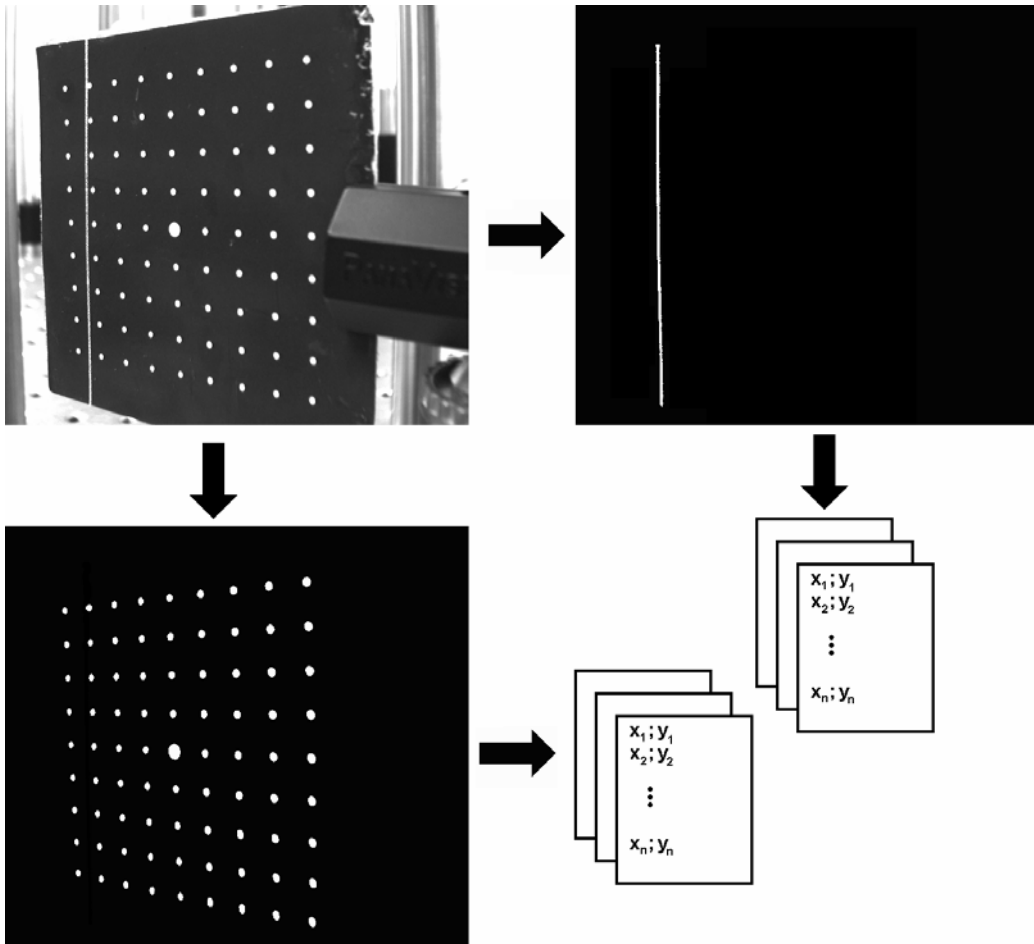


Figura 4.1. Processamento de Imagens na Etapa de Calibração.

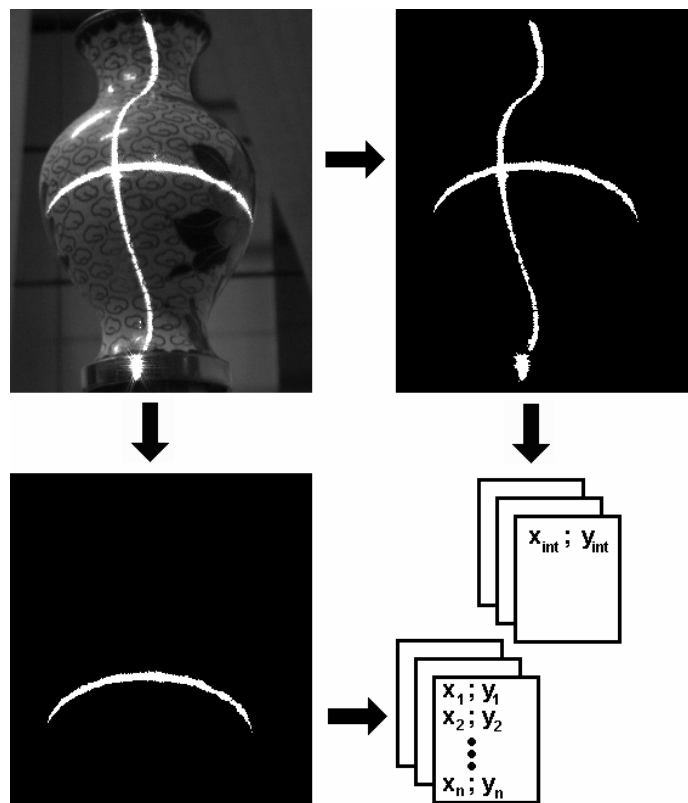


Figura 4.2. Processamento de Imagens na Etapa de Digitalização.

Cada uma dessas tarefas é apresentada detalhadamente na Seção 4.1.1 e no Apêndice F.

As tarefas de extração das coordenadas dos pontos da placa de calibração e de extração da linha laser projetada sobre a placa de calibração são duas tarefas do sistema que necessitam de uma maior interação com o usuário do sistema e que, devido a isso, são apresentadas posteriormente no Apêndice F.

Já as operações de extração do ponto de interseção das linhas lasers e de extração de uma linha laser projetada sobre uma superfície qualquer são operações mais complexas cuja correta execução depende de uma série de variáveis das imagens capturadas e, uma vez que o sistema VISSCAN-3D não é um sistema projetado para uso em ambientes totalmente controlados a alteração e adaptação dos procedimentos utilizados para a execução dessas tarefas pode vir a ser necessária. Logo, de forma a facilitar a alteração desses procedimentos e fornecer ao usuário do sistema uma ferramenta de auxílio nessa tarefa, todas as operações de processamento de imagens foram implementadas utilizando-se uma linguagem de programação de alto nível baseada em *scripts* que são interpretados pelo sistema VISSCAN-3D. Ou seja, cada uma das operações realizadas para a execução de cada dessas tarefas de processamento do sistema são descritas seqüencialmente através de um *script* de processamento.

Cada uma das funções de processamento de imagens implementadas pelo sistema VISSCAN-3D podem ser observadas na Tabela 4.1 juntamente com sua sintaxe.

Tabela 4.1. Lista de Funções de Processamento de Imagens

Funções de Processamento de Imagens						
Função	Sintaxe	Parâmetros				
Adição	ADICAO	<i>Imagem a ser Adicionada</i>				
Blobs	BLOBS	<i>Limite Superior</i>		<i>Limite Inferior</i>		
Vértices	CORNERS	<i>Arquivo de Vértices</i>	<i>Número de Vértices</i>	<i>Distância de Vértices</i>	<i>Fator de Qualidade</i>	
Centro Geométrico	CENTRO	<i>Centro X</i>	<i>Centro Y</i>	<i>Largura X</i>	<i>Largura Y</i>	
Dilatação	DILATAR	<i>Número de Repetições</i>				
Erosão	EROSAO	<i>Número de Repetições</i>				
Excentricidade	EXCENT	<i>Limite Superior</i>		<i>Limite Inferior</i>		
Transformada de Hough	HOUGH	<i>Nome Arquivo</i>	<i>Resolução ρ</i>	<i>Resolução θ</i>	<i>Número de Linhas</i>	<i>Zona Nula θ</i>
Interseção	INTER	<i>Nome Arquivo</i>	<i>Resolução ρ</i>	<i>Resolução θ</i>	<i>Zona Nula θ</i>	
Limiar	LIMIAR	<i>Limiar</i>				
Raio Máximo	MAXR	<i>Limite Superior</i>		<i>Limite Inferior</i>		
Raio Mínimo	MINR	<i>Limite Superior</i>		<i>Limite Inferior</i>		
Mediana	MEDIANA	<i>Largura do Filtro</i>				
Seleção de Região Válida	REGIAO	<i>Centro X</i>	<i>Centro Y</i>	<i>Largura X</i>	<i>Largura Y</i>	
Subtração	SUBTRAIR	<i>Imagem a ser Subtraída</i>				

A estrutura dos *scripts* de processamento foi definida de forma a tornar a tarefa de definição e programação dessas operações simples. O *script* é composto somente pela seqüência de funções, uma por linha, e pela palavra de controle “FINAL” no final da seqüência de funções. Na Figura 4.3 é possível observar a estrutura básica de uma *script* de processamento da nuvem de pontos e na Figura 4.4 um exemplo de *script*, onde é aplicado sobre a nuvem um filtro de média e em seguida são detectadas as regiões da superfície onde existe uma variação brusca. Na Figura 4.5 é possível observar o efeito desse *script* sobre uma imagem de teste amplamente utilizada na literatura (Parker, 1997).

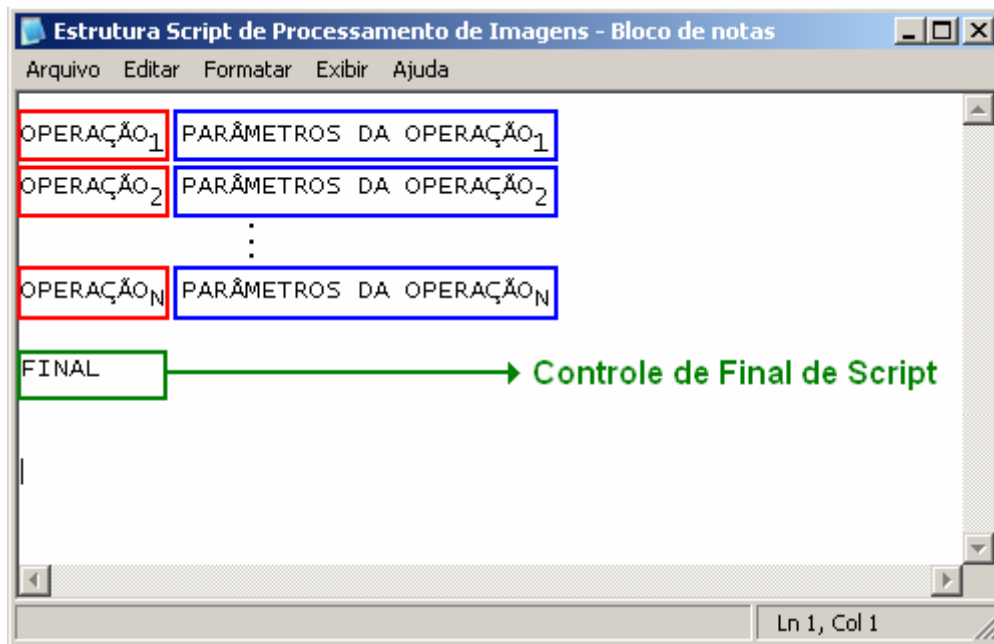


Figura 4.3. Estrutura Básica do Script de Processamento de Imagens.

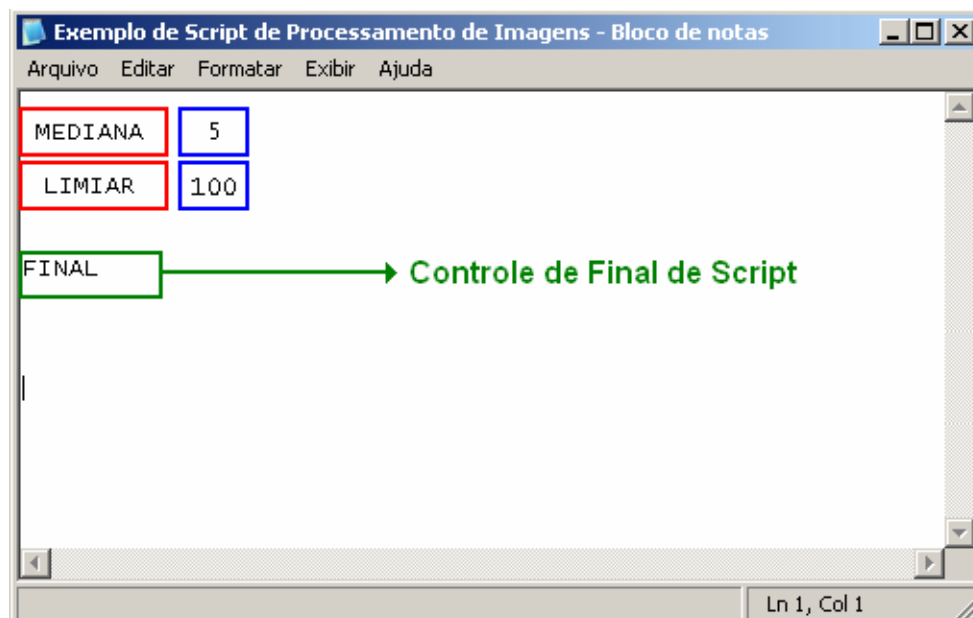


Figura 4.4. Exemplo de Script de Processamento de Imagens.



Figura 4.5. Efeito do Script sobre uma Imagem de Teste.

Apesar de desenvolvida visando seu uso com o sistema VISSCAN-3D, a descrição de tarefas de processamento através de *scripts* de alto nível é uma ferramenta para o auxílio no desenvolvimento de quaisquer projetos voltados para a áreas de processamento de imagens ou que utilizem essas técnicas.

É importante observar que todas as funcionalidades de processamento de imagens do sistema VISSCAN-3D foram desenvolvidas para utilização sobre imagens monocromáticas, ou seja, as funções foram implementadas visando a operação sobre imagens em escala de cinza, e não sobre imagens coloridas.

De forma a facilitar a implementação de cada uma dessas funções de processamento de imagens e ao mesmo tempo oferecer ao usuário uma ferramenta para a alteração, caso necessária, dos *scripts* de processamento responsáveis pelas operações fundamentais de processamento do sistema VISSCAN-3D descritas acima, foi desenvolvido uma plataforma de *software* que trata somente com o processamento de imagens.

Esse *software* tem como principal função a execução de *scripts* de processamento sobre uma imagem qualquer. Na Figura 4.6 é possível observar sua interface principal.

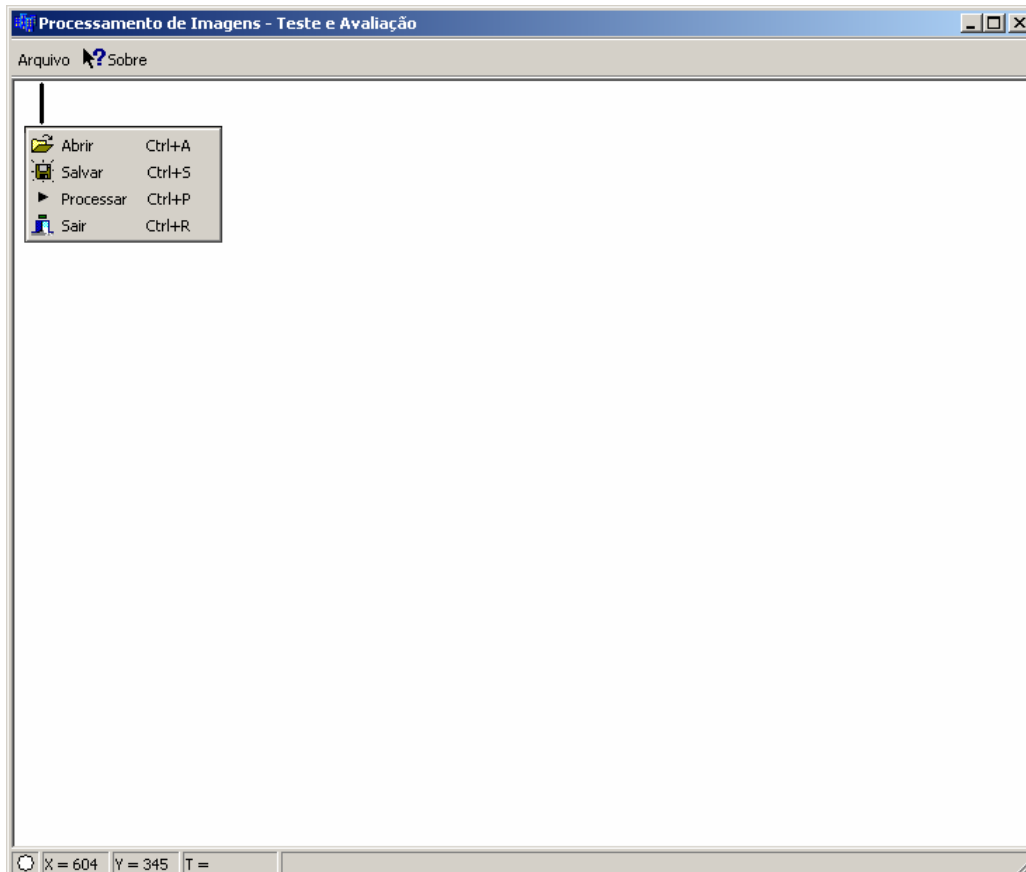


Figura 4.6. Software de Avaliação de Scripts de Processamento de Imagens.

O *software* tem um uso simples e oferece ao usuário somente as opções de abrir imagens e *scripts* e processar esses *scripts*. Para a edição de *scripts* deve ser utilizado um *software* de edição de textos simples como o Bloco de Notas da Microsoft, ou o TextPad da Helios *Software* Solutions, ou qualquer *software* desse tipo disponível.

4.1.1 Principais Tarefas de Processamento do Sistema VISSCAN-3D

Conforme apresentado na Tabela 4.1 o sistema GRACO 3-D possui uma grande variedade de funções de processamento de imagens disponibilizadas para o usuário. O principal objetivo dessas funções é a extração de informações das imagens capturadas para a calibração do sistema e para a digitalização de superfícies. As principais tarefas de processamento de imagens que o sistema VISSCAN-3D tem que realizar são:

- Extração do ponto de interseção das linhas lasers projetadas em uma superfície;
- Extração de uma linha laser projetada sobre uma superfície;

Cada uma dessas tarefas representa um ponto crucial para o processo de digitalização. Nessa seção cada uma dessas tarefas e a maneira como elas são realizadas através de *scripts* de processamento é apresentada, entretanto, é importante observar que, devido às estratégias de processamento serem descritas através desses *scripts*, elas podem ser modificadas a qualquer instante por um usuário. Aqui

são apresentados os *scripts* que foram criados e utilizados durante o desenvolvimento do sistema e que mostraram resultados satisfatórios dentro do ambiente de laboratório.

4.1.1.1 Extração do Ponto de Interseção das Linhas Lasers

A tarefa de extração do ponto de interseção das linhas lasers projetadas sobre uma superfície é realizada em duas etapas distintas utilizando dois *scripts* de processamento. Um *script* de pré-processamento da imagem capturada e um *script* de finalização. A união desses dois *scripts* em um único não acarretaria em alterações no resultado final da operação, entretanto optou-se pelo uso de dois *scripts* para fornecer ao sistema uma maior maleabilidade e uma facilidade para a localização de falhas nesses *scripts*.

Ambos os *scripts* em sua versão padrão são apresentados na Figura 4.7.

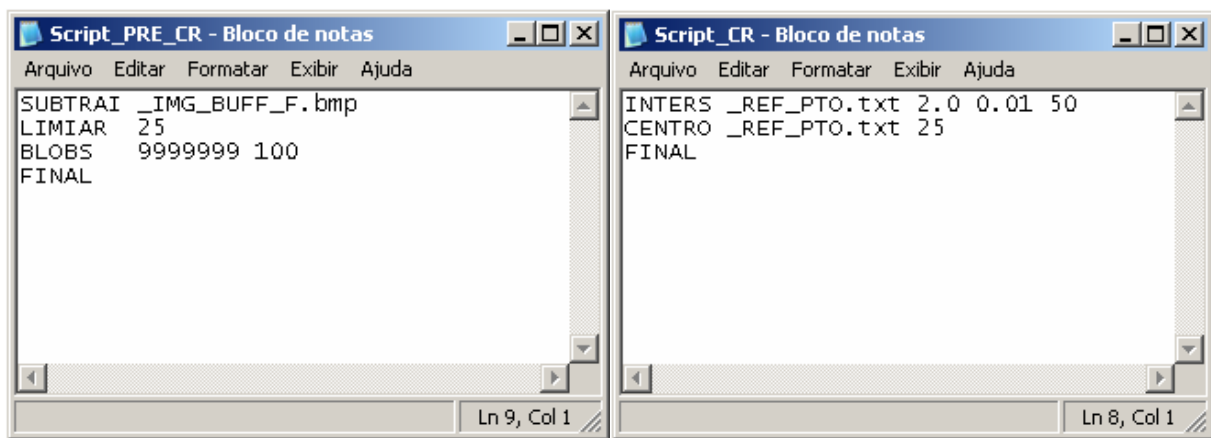


Figura 4.7. Scripts de Pré e Pós Processamento para Extração do Ponto de Interseção.

Antes de detalhar cada uma das operações de processamento desses *scripts* é importante realizar algumas observações acerca dos procedimentos de captura de imagens do sistema VISSCAN-3D. Ao iniciar-se um procedimento de digitalização o sistema VISSCAN-3D captura uma série de imagens da superfície que será digitalizada e salva essas imagens utilizando nomes padrões definidos pelo *software*. A captura de imagens ocorre conforme descrito no fluxograma da Figura 4.8 e os nomes padrões dessas imagens capturadas encontram-se detalhados na Tabela 4.2, junto com exemplos de cada um desses tipos de imagens ilustrados na Figura 4.9.

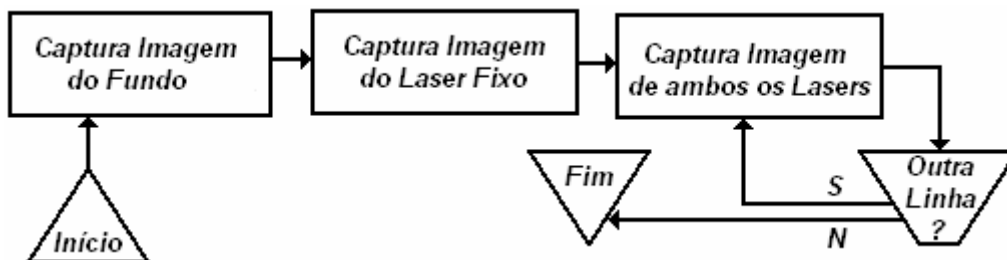


Figura 4.8. Fluxograma dos Procedimentos de Captura de Imagens.

Tabela 4.2. Nomes Padrões das Imagens Capturadas.

Tipo de Imagem	Nome Padrão
Fundo	<i>_IMG_BUFF_F.bmp</i>
Somente Laser Fixo	<i>_IMG_BUFF_L.bmp</i>
Ambos os Lasers	<i>_IMG_BUFF_*.bmp</i>

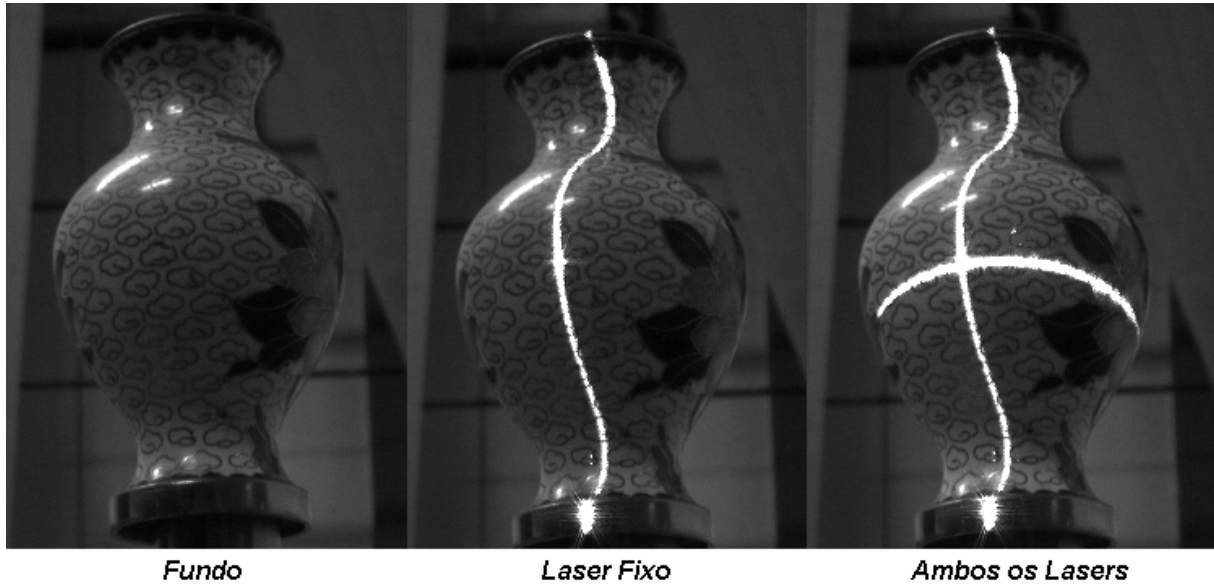


Figura 4.9. Tipos de Imagens Capturadas pelo Sistema VISSCAN-3D.

O primeiro procedimento realizado pelo *script* é a subtração do fundo da imagem. Isso é realizado subtraindo-se a imagem “_IMG_BUFF_F.bmp”. Em seguida é aplicado sobre a imagem um valor de limiar de 25 visando remover pequenos ruídos e manter somente as partes da imagem onde existam alterações mais bruscas, que é exatamente o que acontece ao projetar-se o laser sobre a superfície. Concluindo o pré-processamento da imagem é aplicado um operador de *blobs* sobre a imagem, retirando pequenos pontos que podem ser considerados pequenos demais para serem provenientes da projeção do laser sobre a superfície. O valor de área máxima desse operador não é utilizado e, para evitar que ele atue sobre a imagem, atribui-se um valor elevado a esse parâmetro.

Ao final desse processamento espera-se possuir uma imagem binária que contenha somente as linhas lasers projetadas sobre a superfície e um nível de ruído baixo que não será capaz de influenciar significativamente o pós-processamento.

Concluído o pré-processamento da imagem, o *script* de pós-processamento é inicializado. Esse *script* não realiza mais nenhuma alteração sobre a imagem. Ele é responsável somente pela extração de informação. Seu primeiro procedimento é a execução da função de interseção descrita no Apêndice D. É importante entretanto lembrar que as linhas projetadas pelos diodos lasers podem não ser retas, de forma que essa função fornece somente uma primeira estimativa do ponto de interseção que é então aprimorada utilizando-se a função de cálculo do centro geométrico.

O *script* de pós-processamento gera como saída um arquivo de texto que informa ao sistema o ponto de interseção. Ao alterar-se esse *script* é importante manter o nome do arquivo de saída “_REF_PTO.txt”, pois é nesse local que o sistema VISSCAN-3D espera localizar essa informação e, caso ele não a encontre, a digitalização da superfície irá falhar.

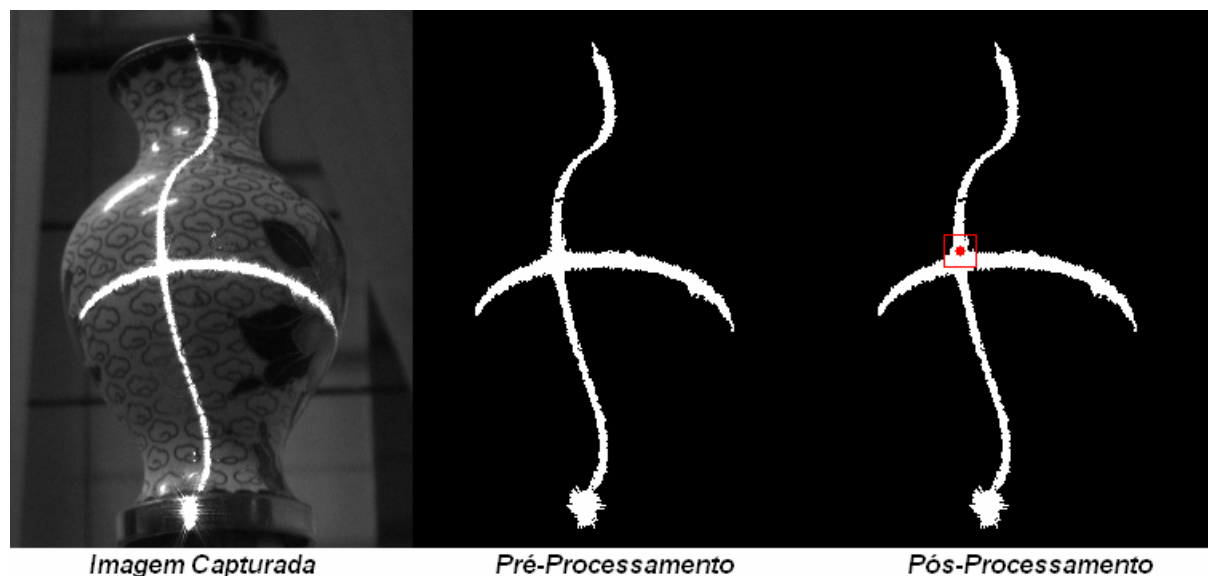


Figura 4.10. Etapas de Pré e Pós-Processamento para Extração do Ponto de Interseção.

4.1.1.2 Extração da Linha Laser Projetada sobre a Superfície

Assim como a tarefa de extração do ponto de interseção, a tarefa de extração da linha laser projetada sobre uma superfície é realizada em duas etapas de processamento. Ambos os *scripts* de processamento podem ser observados na Figura 4.11.

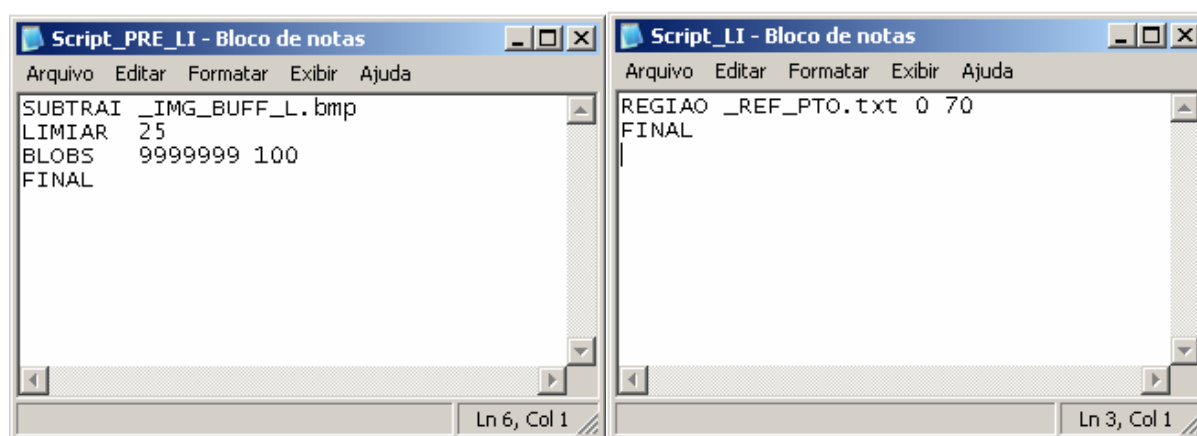


Figura 4.11. Scripts de Pré e Pós Processamento para Extração da Linha Laser.

O primeiro procedimento realizado pelo *script* é a subtração da imagem com o laser fixo. Isso é realizado subtraindo-se a imagem “_IMG_BUFF_L.bmp”, de forma que não apenas o fundo da imagem é removido, mas também a linha de laser projetada pelo diodo fixo. Em seguida é aplicado sobre a imagem um valor de limiar de valor 25 visando remover pequenos ruídos da imagem e manter

somente as partes da imagem onde existem alterações mais bruscas, o que ocorre na região onde o laser móvel é projetado sobre a superfície. O pré processamento é então concluído com a aplicação do operador de *blobs* sobre a imagem.

Ambos os procedimentos padrões de pré-processamento utilizados na extração do ponto de interseção dos lasers e na extração da linha laser são os mesmos. Logo, ao concluir-se o pré-processamento da imagem para extração da linha laser espera-se também possuir uma imagem binária. Entretanto, a imagem binária obtida conterá somente a linha laser projetada pelo diodo móvel sobre a superfície.

Já o pós-processamento é utilizado para se minimizar ao máximo o nível de ruído na imagem. Ele realiza somente uma seleção de região válida em torno do ponto de cruzamento determinado pelo processamento apresentado na Seção 4.1.1.1. O *script* aplica uma restrição de região somente na dimensão Y da imagem, conforme ilustrado na Figura 4.12. Essa restrição de região válida impõe uma restrição quanto a variação de profundidade esperada nas superfícies digitalizadas. Em superfícies suaves é esperado que não haja variações muito bruscas. Essa restrição, apesar de ajudar na remoção de possíveis ruídos, age limitando a área de trabalho do sensor, logo, caso se deseje maximizar essa área e as variações de profundidade esperadas basta alterar o *script* de pós-processamento, entretanto é importante lembrar que a presença de ruído nessa imagem levará a geração de pontos espúrios na medição.

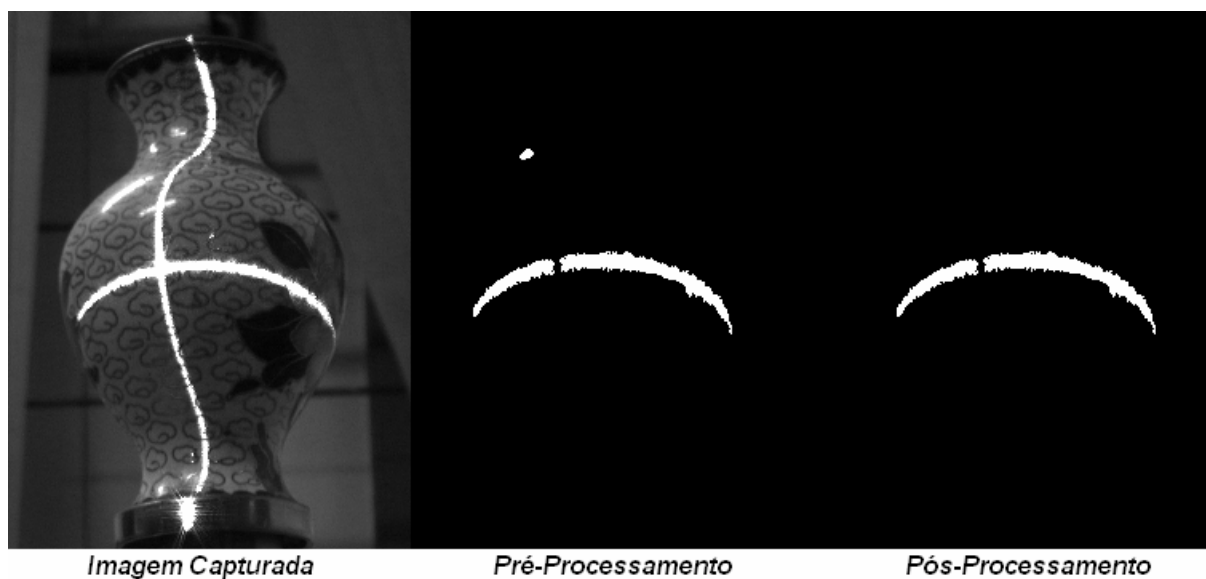


Figura 4.12. Exemplo das Etapas de Processamento para Extração da Linha Laser.

A presença de pontos espúrios é comum na digitalização de superfícies utilizando lasers, onde uma grande massa de dados é obtida e é nesse contexto que técnicas de redução e segmentação de dados podem ser utilizadas (Banhak e Fischer, 2001, Lee et.al., 2001). Essas técnicas visam a remoção de pontos espúrios ou sua desconsideração durante a reconstrução da superfície. O estudo dessas técnicas, entretanto, não é parte do escopo desse trabalho.

Concluída a etapa de pós-processamento espera-se possuir uma imagem binária que contenha somente a linha laser projetada sobre a superfície pelo diodo laser móvel, sendo necessário somente a extração das coordenadas dessa linha.

A extração das coordenadas de cada um dos pontos que formam essa linha laser é realizada através do particionamento da linha de formas distintas de acordo com a resolução desejada e com a orientação do plano laser projetado pelo diodo móvel.

A orientação do plano laser móvel influencia a direção em que a varredura da imagem é realizada, conforme ilustrado na Figura 4.13.

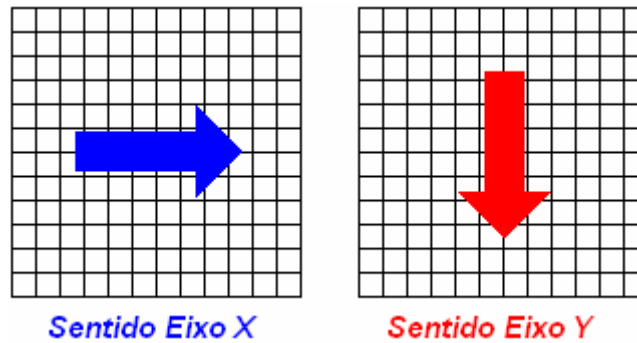


Figura 4.13. Direções de Varredura da Imagem.

No caso do particionamento da linha laser com a máxima resolução possível no sentido do eixo-X da imagem, para cada coluna i da imagem é calculada a média das coordenadas de linha dos pixels de valor “1”, sendo obtido um dos pontos da linha laser conforme a Equação (4.1)

$$P_{Laser} = \left[\frac{1}{m} \sum_{k=0}^m j \right], \quad (4.1)$$

onde m é o número de pixels de valor “1” na coluna i e j é a coordenada da linha onde cada um desses pixels estão. Caso não haja nenhum pixel “1” em uma coluna, nenhum ponto é adicionado e a próxima coluna da imagem é avaliada.

Já para a varredura da imagem no sentido do eixo-Y cada linha j da imagem é avaliada e a média das coordenadas de coluna dos pixels de valor “1” é calculada conforme a Equação (4.2)

$$P_{Laser} = \left[\frac{1}{m} \sum_{k=0}^m i \right]_j \quad (4.2)$$

Esse método de varredura gera uma nuvem de pontos que pode não ser uniforme e não fornece ao usuário nenhum controle do número de pontos que serão obtidos. Logo, de forma a oferecer a possibilidade da escolha do número de pontos que se deseja obter na nuvem uma segunda técnica de particionamento também foi desenvolvida e implementada no sistema VISSCAN-3D. Uma vez que o usuário defina o número de pontos n que ele deseja ter em cada posição da linha laser móvel, é calculado um fator de escala λ para o particionamento. No caso do particionamento da linha laser no sentido do eixo-X da imagem tem-se:

$$\lambda = \frac{i_{\max} - i_{\min}}{n - 1}, \quad (4.3)$$

onde i_{\max} é a maior coordenada de coluna onde existe um pixel “1” e i_{\min} a mínima coordenada de coluna onde existe um pixel “1”.

Em seguida, os pontos adicionados ao sistema são dados pela Equação (4.1) onde as colunas avaliadas são as colunas:

$$i = i_{\min} + \lambda k, \quad (4.4)$$

onde $k = 1, 2, \dots, n$.

O mesmo procedimento ocorre quando o particionamento da linha laser é realizado no sentido do eixo-Y, entretanto os pontos da linha laser serão obtidos a partir da Equação (4.2) e o fator de escala e as linhas avaliadas serão dadas por:

$$\lambda = \frac{j_{\max} - j_{\min}}{n - 1} \quad (4.5)$$

$$j = j_{\min} + \lambda k, \quad (4.6)$$

onde j_{\max} é a maior coordenada de linha onde existe um pixel “1” e j_{\min} a mínima coordenada de linha onde existe um pixel “1”.

4.2 PROCESSAMENTO DA NUVEM DE PONTOS

O sistema de digitalização VISSCAN-3D tem como principal objetivo a captura de informação tridimensional de uma superfície, obtendo-se uma nuvem de pontos que represente essa superfície com precisão suficiente. Entretanto, o *software* desenvolvido apresenta uma série de outras funcionalidades para facilitar a manipulação e a visualização dos dados contidos nessa nuvem de pontos.

Uma dessas funcionalidades é a execução de algumas operações de processamento sobre essa malha de pontos, como a aplicação de filtros e a detecção de variações bruscas. Com o uso dessas operações é possível detectar regiões específicas dentro da nuvem de pontos como bordas, excentricidades e descontinuidades.

De forma a oferecer total liberdade ao usuário do sistema durante a realização dessas operações foi criada uma linguagem de programação de pequeno porte e de alto nível baseada em *scripts* que são interpretados pelo sistema VISSCAN-3D. Ou seja, todas as operações que serão aplicadas sobre uma nuvem de pontos são descritas seqüencialmente através de um *script* de processamento. Em seguida esse *script* é lido por um interpretador de código dentro do sistema VISSCAN-3D, identificando-se cada uma das operações que se deseja realizar sobre a nuvem e aplicando-as.

O processamento das nuvens de pontos visando a extração de características não é o foco principal desse trabalho, de forma que somente algumas funções de processamento foram implementadas e o assunto não será abordado detalhadamente aqui. Alguns resultados obtidos são apresentados, entretanto o tema não foi amplamente explorado experimentalmente. A ampliação dessas funções de processamento pode ser realizada facilmente, expandindo-se as funcionalidades do *software* VISSCAN-3D e tornando-o não apenas um ambiente de digitalização, mas também um sistema de detecção e classificação de características.

As funções implementadas para processamento e detecção de características em nuvens de pontos foram divididas em duas classes principais. Funções de processamento e funções de detecção.

As funções de processamento realizam alterações na nuvem de pontos, modificando as coordenadas 3D dos pontos. Já as funções de detecção realizam somente a detecção de características, destacando as regiões da superfície onde essas características podem ser observadas.

Cada um desses grupos de funções juntamente com sua sintaxe podem ser observados na Tabela 4.3 e na Tabela 4.4. No Apêndice E cada uma dessas funções e suas implementações são apresentadas.

Tabela 4.3. Lista de Funções de Processamento da Nuvem de Pontos.

Funções de Processamento		
Função	Sintaxe	Parâmetros
Média	MEDIA	<i>Largura do Filtro</i>
Mediana	MEDIANA	<i>Largura do Filtro</i>
Saturação	SATURACAO	<i>Limite Superior</i> <i>Limite Inferior</i>

Tabela 4.4. Lista de Funções de Detecção de Características da Nuvem de Pontos.

Funções de Detecção			
Função	Sintaxe	Parâmetros	
Limiar	LIMIAR	<i>Limite Superior</i>	<i>Limite Inferior</i>
Gradiente	GRADIENTE	<i>Limite</i>	
Laplaciano	LAPLACIANO	<i>Limite</i>	
Curvatura	CURVATURA	<i>Operação</i>	<i>Limite</i>
Plano Médio	PLANO	<i>Limite</i>	
Planicidade	ERRO	<i>Arquivo de Saída</i>	

A estrutura dos *scripts* de processamento foi definida de forma a tornar a tarefa de definição e programação dessas operações o mais simples possível. O *script* é composto somente pela seqüência de funções, uma por linha, e pela palavra de controle “FINAL” no final da seqüência de funções. Na Figura 4.14 é possível observar a estrutura básica de um *script* de processamento da nuvem de pontos e na Figura 4.15 um exemplo de *script*, onde é aplicado sobre a nuvem um filtro de média e em seguida são detectadas as regiões da superfície onde existe uma variação brusca. Na Figura 4.16 é possível observar o efeito desse *script* sobre uma superfície de teste.

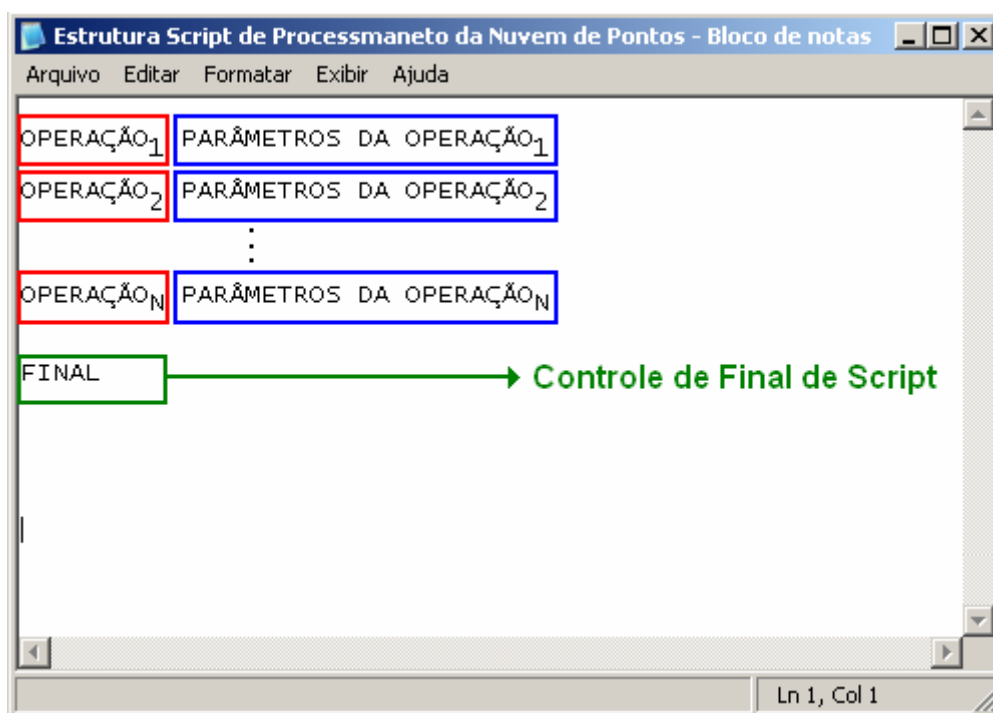


Figura 4.14. Estrutura Básica do Script de Processamento da Nuvem de Pontos.

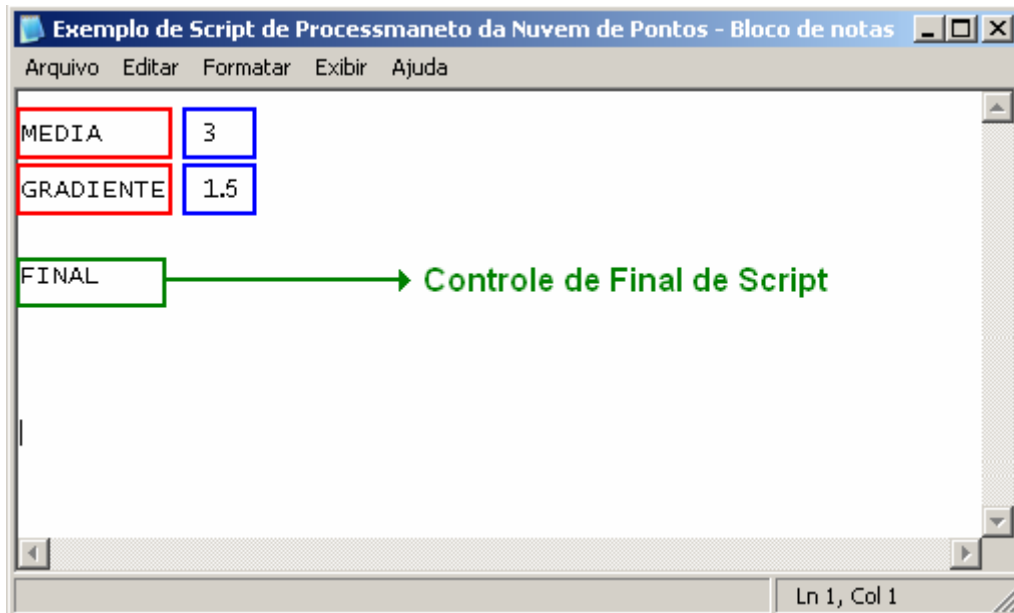


Figura 4.15. Exemplo de Script de Processamento da Nuvem de Pontos.

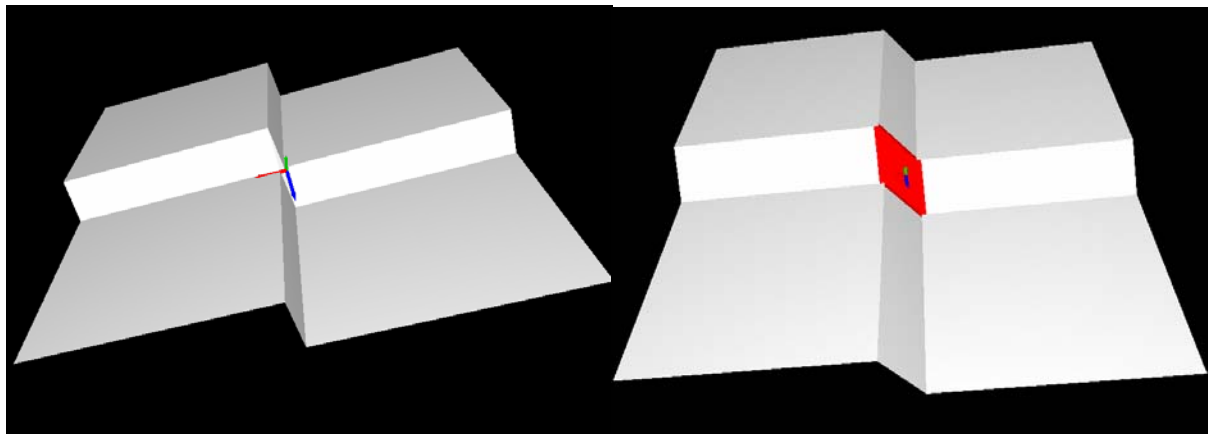


Figura 4.16. Efeito do Script sobre uma Superfície de Teste.

A grande maioria das funções de processamento aqui implementadas, como o cálculo do gradiente e do Laplaciano, dependem diretamente do cálculo das derivadas da malha de pontos. Essas malhas entretanto são matrizes discretas de pontos que podem não ser uniformemente distribuídos. Logo, para o cálculo dessas derivadas é necessário o uso de técnicas de derivação numérica conforme as ilustradas no Apêndice A.

4.3 O SISTEMA GRÁFICO DE VISUALIZAÇÃO

4.3.1 OpenGL

O OpenGL (Open Graphics Library) é um padrão de especificação que define uma API multiplataforma para o desenvolvimento de aplicações gráficas bidimensionais e tridimensionais (Neider *et.al.*, 1993, OpenGL Architecture Review Board, 1992). Ele define uma interface que consiste em cerca de 250 funções que podem ser utilizadas para desenhar cenas tridimensionais complexas a partir de primitivas, constituindo uma poderosa biblioteca de renderização e modelagem de baixo nível. Desenvolvido em 1992 pela Silicon Graphics Inc., o OpenGL é utilizado amplamente em uma grande variedade de aplicações como ferramentas CAD, simulações e visualizações científicas, jogos eletrônicos e sistemas de realidade virtual.

A funcionalidade básica do OpenGL é a transformação de primitivas como pontos, linhas e polígonos em pixels (OpenGL Architecture Review Board, 1992), deixando o programador livre para se concentrar na geração das primitivas e em como essas primitivas serão processadas pela máquina gráfica.

A grande vantagem no uso de uma API gráfica para a visualização está na facilidade de se realizar o interfaceamento com os sistemas de aceleração gráficos através de uma única biblioteca de funções, independente do fabricante do acelerador gráfico.

Neider *et.al.*(1993) e OpenGL Architecture Review Board(1992) apresentam informações detalhadas a respeito do padrão OpenGL e seu uso para o desenvolvimento de aplicativos.

4.3.2 A Máquina Gráfica Implementada

Nessa fase de desenvolvimento, a visualização dos dados, apesar de importante, não é a principal finalidade do sistema. Dessa forma, a máquina gráfica implementada para a geração de gráficos tridimensionais a partir de dados obtidos do sistema de digitalização utiliza uma estrutura básica e não foram realizadas considerações significativas a respeito de técnicas de interpolação ou suavização de superfícies.

Existe uma infinidade de técnicas de reconstrução de superfícies a partir de uma nuvem de pontos. Várias publicações vem sendo feitas principalmente no uso de técnicas como NURBS e SPLINES (Dimas e Briassoulis, 1999), aqui entretanto optou-se por técnicas simples de preenchimento, sem o uso de nenhum tipo complexo de interpolação.

A máquina gráfica do sistema VISSCAN-3D é um exemplo de aplicativo que utiliza o OpenGL para a visualização de dados. O sistema gera uma série de pontos que fazem parte da superfície digitalizada e esses pontos são utilizados como primitivas para a máquina gráfica.

O sistema gráfico implementado, apesar de simples, permite a visualização dos dados obtidos de uma maneira fácil e concisa, oferecendo cinco diferentes formas de visualização dos dados, dependendo

das opções do usuário. A máquina gráfica oferece duas opções básica de visualização de superfícies. O tipo de polígono a ser utilizado e a forma como esse polígono deve ser reconstruído, conforme ilustrado na Tabela 4.5.

Tabela 4.5. Opções de Renderização da Superfície Digitalizada.

Tipo de Polígono	Tipo de Renderização
Triângulos	Pontos
Quadrados	Wireframe
	Preenchimento

Logo, é possível a visualização somente dos pontos obtidos, a visualização de uma estrutura tipo *wireframe* formada com esses pontos conectados por quadrados ou triângulos, ou a visualização de uma superfície interpolada com quadrados ou triângulos.

Na Figura 4.17 é possível observar uma superfície simples visualizada com cada uma dessas cinco formas de renderização.

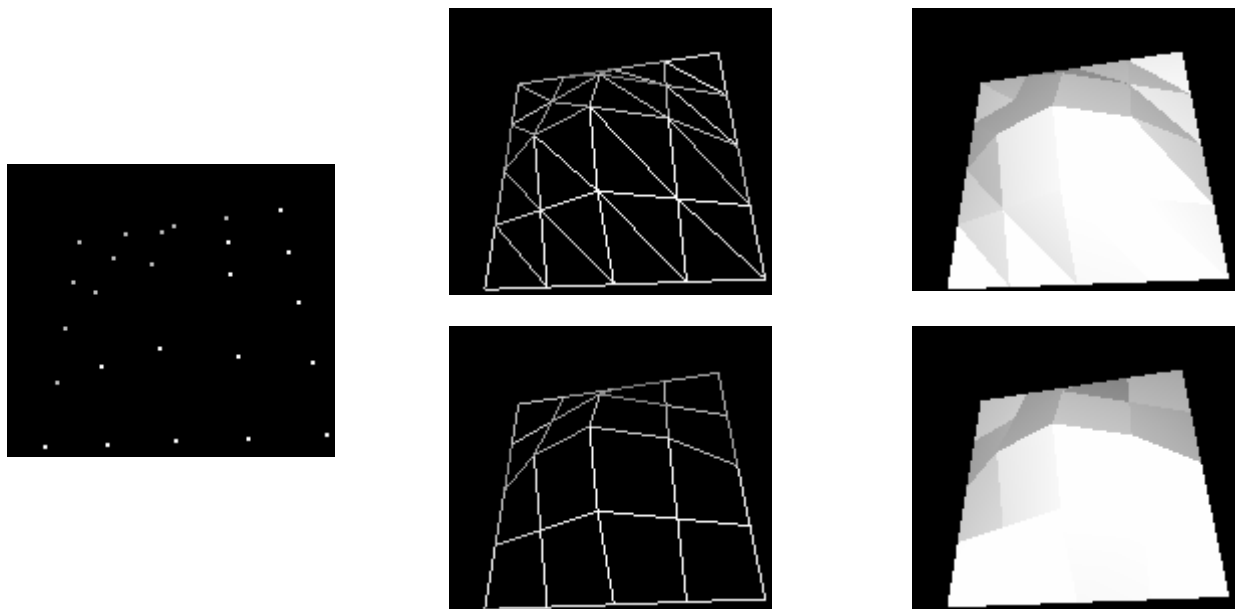


Figura 4.17. Diferentes Formas de Renderização Disponíveis.

A geração desses gráficos é realizada através da inserção dos pontos obtidos na máquina gráfica, entretanto para tornar possível a visualização tridimensional desses dados e a geração das estruturas *wireframe* e das superfícies ilustradas na Figura 4.17 é necessário definir-se os polígonos que formam essas estruturas.

Antes de entrar em detalhes a respeito da geração desses polígonos é necessário destacar uma característica do processo de digitalização baseado em varredura laser. Devido à seqüência do processo de varredura, os pontos da superfície digitalizada são obtidos seqüencialmente conforme ilustrado na Figura 4.18.

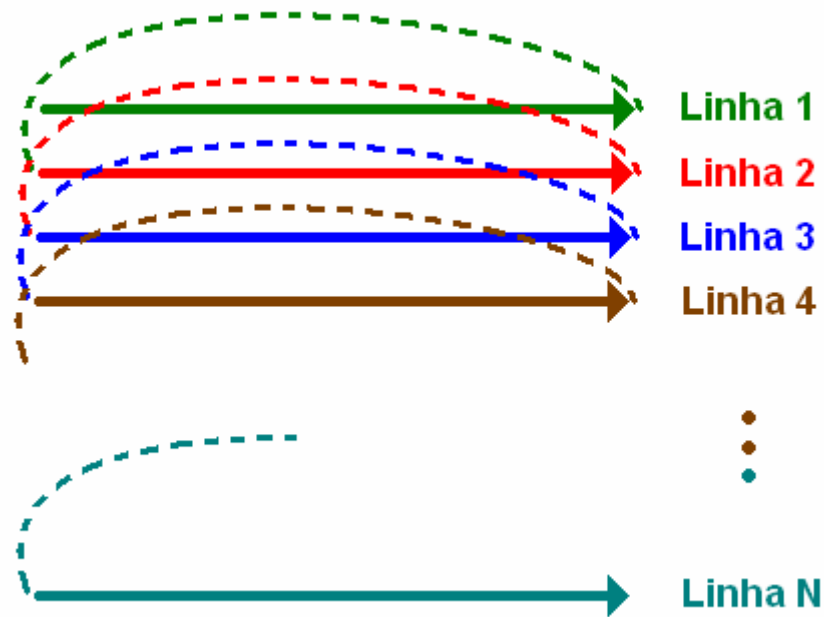


Figura 4.18. Sequência da Varredura Laser.

A máquina gráfica responsável pela geração dos polígonos e pela renderização das superfícies digitalizadas foi implementada utilizando-se desse fato para definir quais pontos fazem parte de cada polígono a ser formado. Observando duas linhas sequenciais de digitalização, sabe-se que essas linhas definem pontos vizinhos, logo a estrutura de polígonos pode ser definida conforme ilustrado na Figura 4.19.

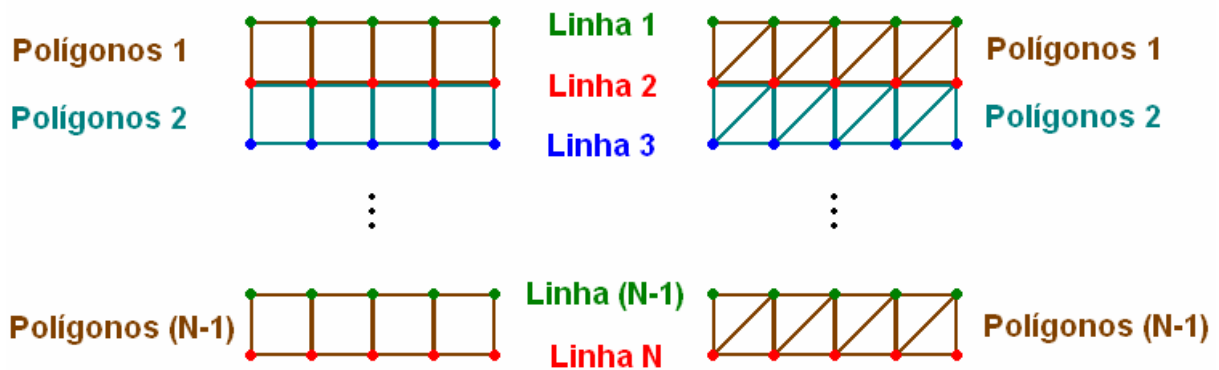


Figura 4.19. Renderização baseada em Triângulos e Quadrados.

Uma vez definido um polígono é ainda necessário definir o vetor normal a esse, de forma que o sistema OpenGL seja capaz de realizar os cálculos de iluminação e criar corretamente o efeito de profundidade durante a visualização da superfície na tela bidimensional do computador.

Conhecendo-se quatro pontos P_1, P_2, P_3, P_4 que definem um quadrado conforme ilustrado na Figura 4.19, sua normal pode ser definida como:

$$\vec{N} = (P_3 - P_1) \otimes (P_2 - P_1) \quad (4.7)$$

Já para o triângulo, com os três pontos P_1, P_2, P_3 , que o formam conforme ilustrado na Figura 4.19, tem-se também:

$$\vec{N} = (P_3 - P_1) \otimes (P_2 - P_1) \quad (4.8)$$

Através da Equação (4.7) e da Equação (4.8) observa-se que para o cálculo da normal de um plano somente três pontos são necessários (Edwards e Penney, 1998). Ao definir-se um quadrado como polígono de interpolação entre quatro pontos que podem não ser coplanares, obtêm-se na realidade dois triângulos com uma única normal e isso gera uma iluminação não suave na superfície quando essa é renderizada. Ou seja, uso de triângulos para a interpolação de superfícies é preferível ao uso de quadrados, entretanto, para a visualização de uma superfície em *wireframe*, o uso de quadrados normalmente gera uma estrutura de mais fácil visualização.

5 CALIBRAÇÃO AUTOMÁTICA

5.1 CALIBRAÇÃO DA CÂMERA

A calibração de câmeras vem recebendo uma grande atenção da comunidade de visão computacional nas últimas décadas (Sun e Cooperstock, 2005). No contexto de visão, a calibração de câmeras consiste na determinação das características geométricas e ópticas como a posição e orientação de uma câmera em relação a um referencial externo. O principal propósito dessa calibração é inferir informações tridimensionais a partir de imagens, ou inferir informações bidimensionais a partir de dados sobre uma cena tridimensional (Tsai, 1987).

Nesse trabalho o modelo de câmera utilizado foi apresentado na Seção 2.1.4 e aqui são discutidos os procedimentos de calibração desse modelo. A calibração completa do modelo de câmera utilizado divide-se em cinco etapas.

- Obtenção dos Pontos de Calibração;
- Calibração da Razão μ ;
- Calibração do Centro da Imagem (C_x, C_y) ;
- Calibração do Modelo RAC;
- Otimização Numérica do Resultado Obtido;

Cada uma dessas etapas é de grande importância para a obtenção de um modelo preciso que descreva a câmera e são apresentadas detalhadamente a seguir.

5.1.1 Obtenção dos Pontos de Calibração

Os pontos de calibração formam o conjunto básico de dados a partir do qual todo o modelo da câmera é calibrado. Ele é formado pelas coordenadas em unidades métricas de uma série de pontos em relação a um referencial global qualquer (x_w, y_w, z_w) e as coordenadas em pixels resultantes na imagem (X_i, Y_i) .

Entretanto, de forma a obter-se essa relação de maneira precisa, são utilizados alvos de calibração. Esses alvos podem assumir as mais diversas configurações como tabuleiros de xadrez ou malhas de pontos e podem ainda possuir diferentes formas como cubos, placas planas, tetraedros e outras. O importante em um alvo de calibração é que seja possível diferenciar facilmente os pontos de calibração e que, dada a geometria do alvo, a posição relativa entre eles seja conhecida.

Em todas as técnicas aqui desenvolvidas, utilizou-se um alvo de calibração constituído por uma malha de pontos distribuída em uma placa plana. A malha de pontos utilizada é constituída por 81 pontos

distribuídos em 9 linhas e 9 colunas conforme ilustrado na Figura 3.16, e o espaçamento entre cada um dos pontos é de 9 mm.

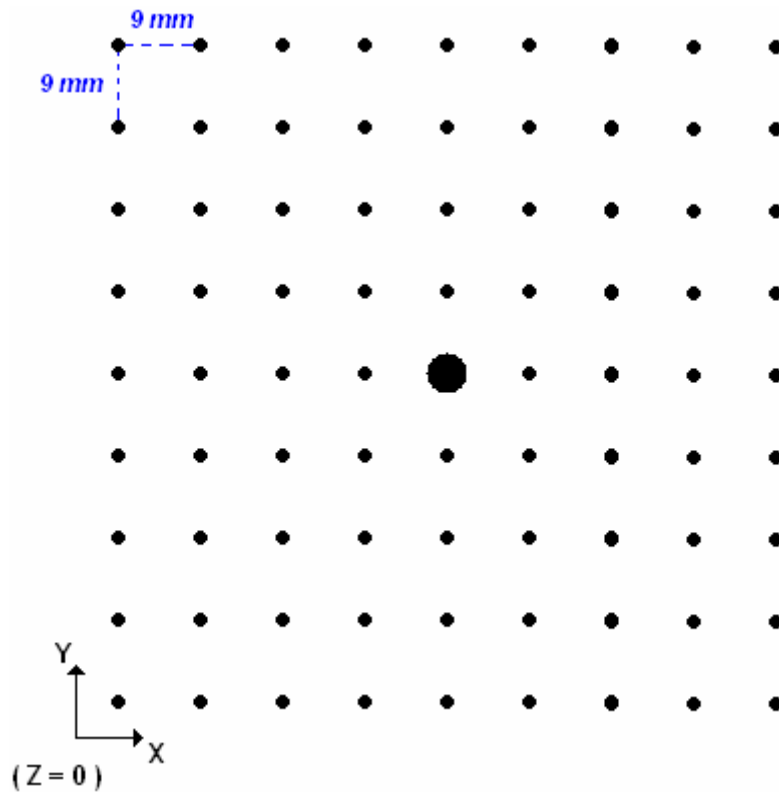


Figura 5.1. Distribuição dos Pontos de Calibração.

Em relação ao posicionamento da origem do sistema de referência global, a escolha é da pessoa que está realizando o processo de calibração do sistema e influencia diretamente sobre os parâmetros extrínsecos que serão determinados para a câmera.

A escolha dessa origem, entretanto, deve ser realizada de forma que o parâmetro extrínseco t_y não seja nulo, o que invalidaria o processo de calibração aqui apresentado (Zhuang e Roth, 1996, Tsai, 1987)

Durante todas as etapas de desenvolvimento desse projeto essa origem foi sempre posicionada de forma que o plano formado pela placa de calibração fosse descrito como:

$$z_w = 0 \tag{5.1}$$

Já a obtenção das coordenadas dos pontos (X_i, Y_i) na imagem capturada pela câmera é realizada através de um série de funções de processamento de imagens e é descrita no Apêndice F.

5.1.2 Calibração do Modelo RAC

Supondo inicialmente que as coordenadas do centro da imagem (C_x, C_y) e a razão μ entre os fatores de escala (f_x, f_y) foram calibradas previamente (Seção 5.1.3 e Seção 5.1.4) e já são conhecidas, é possível determinar os outros parâmetros do modelo da câmera a partir de um conjunto coplanar de pontos de calibração.

O algoritmo de calibração consiste em duas etapas. Na primeira, a matriz de rotação R e os parâmetros translacionais t_x e t_y são calculados. Já na segunda etapa os parâmetros restantes são estimados a partir desses resultados.

O primeiro passo para a calibração da câmera é o cálculo de um conjunto de coordenadas intermediárias de apoio dadas por:

$$\begin{cases} X_j = {}^jX - C_x \\ Y_j = {}^jY - C_y \end{cases}, \quad (5.2)$$

onde $({}^jX, {}^jY)$ são as coordenadas em pixels na imagem do j -ésimo ponto de calibração e $j = 1 \dots N$ com N igual ao número total de pontos de calibração.

A seguir, como o conceito da restrição de alinhamento radial independe da distorção da lente e da distância focal desta (Zhuang e Roth, 1996, Tsai, 1987), é possível definir o seguinte sistema de equações a partir da Equação (2.19):

$$\begin{bmatrix} {}^jx_w Y_j & {}^jy_w Y_j & Y_j & -{}^jx_w \mu X_j & -{}^jy_w \mu X_j \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \mu X_j, \quad (5.3)$$

onde

$$\begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 \end{bmatrix} \equiv \begin{bmatrix} \frac{r_1}{t_y} & \frac{r_2}{t_y} & \frac{t_x}{t_y} & \frac{r_4}{t_y} & \frac{r_5}{t_y} \end{bmatrix} \quad (5.4)$$

e $({}^jx_w, {}^jy_w, {}^jz_w)$ são as coordenadas globais do j -ésimo ponto de calibração.

O número mínimo de equações para a solução desse sistema é cinco. Isso implica na necessidade de pelo menos cinco pontos de calibração não-colineares. Entretanto, na prática N deve ser um número

maior de forma a gerar um sistema sobre-determinado de equações lineares que pode ser solucionado através de um método de otimização de mínimos quadrados. Sun e Cooperstock (2005) apresentam um estudo detalhado da influência do número de pontos utilizados na calibração .

Outra observação importante a cerca desse sistema é o termo t_y . No desenvolvimento dessas equações a partir das equações de alinhamento radial é suposto que $t_y \neq 0$. Caso isso não ocorra é ainda possível realizar o desenvolvimento desse sistema em termos de t_x . Entretanto, como no procedimento de calibração tem-se controle das coordenadas globais $(^j x_w, ^j y_w, ^j z_w)$, é possível escolher um sistema de referência global de forma a evitar esse problema.

Logo, uma vez determinados os parâmetros v_1, v_2, v_3, v_4 e v_5 , é possível expressar o parâmetro t_y em função desses parâmetros. Definindo-se uma matriz C dada por:

$$C \equiv \begin{bmatrix} v_1 & v_2 \\ v_4 & v_5 \end{bmatrix} \quad (5.5)$$

Se nenhuma coluna ou linha dessa matriz for nula, t_y pode ser determinado como:

$$t_y^2 = \frac{S_r - \sqrt{S_r^2 - 4(v_1 v_5 - v_4 v_2)^2}}{2(v_1 v_5 - v_4 v_2)^2}, \quad (5.6)$$

onde $S_r \equiv v_1^2 + v_2^2 + v_4^2 + v_5^2$

Já, caso uma linha ou coluna de C seja nula t_y pode ser obtido diretamente da relação:

$$t_y^2 = \frac{1}{v_i^2 + v_j^2}, \quad (5.7)$$

onde v_i e v_j são os elementos não nulos da matriz C .

Uma vez determinado o valor de t_y é ainda necessário determinar seu sinal. Para tanto utiliza-se o fato de que os sinais de x e X e de y e Y devem ser consistentes conforme pode ser observado na Figura 2.3.

Supondo inicialmente que $t_y > 0$ tem-se:

$$\begin{aligned}
r_1 &= v_1 t_y \\
r_2 &= v_2 t_y \\
r_4 &= v_4 t_y \\
r_5 &= v_5 t_y \\
t_x &= v_3 t_y
\end{aligned} \tag{5.8}$$

Selecionando-se então um ponto de calibração arbitrário, calculam-se suas coordenadas (x, y) :

$$\begin{cases} x = r_1 x_w + r_2 y_w + t_x \\ y = r_4 x_w + r_5 y_w + t_y \end{cases} \tag{5.9}$$

Então, caso ambos os sinais de x e X e de y e Y sejam iguais, a suposição de que $t_y > 0$ está correta. Caso contrário o sinal de t_y é invertido assim como os sinais de r_1, r_2, r_4, r_5 e t_x .

Dada uma submatriz 2x2 qualquer da matriz de rotação R , existem duas possíveis soluções para essa matriz (Edwards e Penney, 1998, Zhuang e Roth, 1996). Cada uma dessas soluções fornece um valor diferente de f_x na segunda etapa de calibração. Um com valor negativo e outro positivo. Uma vez que o valor de f_x deve ser positivo, essa condição pode ser utilizada para se determinar a solução correta de R .

A primeira solução possível para a matriz R é dada por:

$$\begin{aligned}
r_3 &= \sqrt{1 - r_1^2 - r_2^2} \\
r_6 &= -s \sqrt{1 - r_4^2 - r_5^2} \\
[r_7 \quad r_8 \quad r_9]^T &= [r_1 \quad r_2 \quad r_3]^T \otimes [r_4 \quad r_5 \quad r_6]^T
\end{aligned} \tag{5.10}$$

onde $s = \text{sinal}(r_1 r_4 + r_2 r_5)$.

Caso esse resultado resulte um uma valor positivo de f_x na segunda etapa de calibração da câmera essa solução é mantida, caso contrário a matriz R é modificada como:

$$\begin{aligned}
r_3 &= -\sqrt{1 - r_1^2 - r_2^2} \\
r_6 &= s \sqrt{1 - r_4^2 - r_5^2} \\
[r_7 \quad r_8 \quad r_9]^T &= [r_1 \quad r_2 \quad r_3]^T \otimes [r_4 \quad r_5 \quad r_6]^T
\end{aligned} \tag{5.11}$$

A segunda etapa de calibração da câmera consiste na computação dos parâmetros f_x , f_y , t_z e k a partir dos resultados obtidos na primeira etapa.

Tendo-se os valores de R , t_x e t_y como conhecidos, para cada ponto de calibração tem-se (Motta, *et.al.*, 2001, Motta, 1999):

$$\begin{bmatrix} -X_j & jx & jx^j r^2 \\ -Y_j & \mu^j y & -\mu^j y r^2 \end{bmatrix} \begin{bmatrix} t_z \\ f_x \\ kf_x \end{bmatrix} = \begin{bmatrix} X_j w_j \\ Y_j w_j \end{bmatrix} \quad (5.12)$$

, onde

$$\begin{cases} jx \equiv r_1^j x_w + r_2^j y_w + t_x \\ w_j \equiv r_7^j x_w + r_8^j y_w \\ r^2 \equiv \mu^2 X_j^2 + Y_j^2 \end{cases} \quad (5.13)$$

Tendo-se um número de pontos de calibração N superior a três obtêm-se um sistema de equações sobre-determinado que pode ser solucionado utilizando-se métodos de solução baseados em mínimos quadrados.

Um vez determinados esses parâmetros é possível obter os parâmetros intrínsecos restantes.

$$\begin{cases} f_y = f_x \mu \\ k = \frac{kf_x}{f_x} \end{cases} \quad (5.14)$$

Logo, todo o modelo RAC é calibrado e pode ser utilizado para o mapeamento de pontos da imagem em pontos tridimensionais no espaço.

5.1.3 Calibração da Razão μ

Em um sistema de visão os fatores de escala f_x e f_y são fixos, de forma que para um dado conjunto de câmera, lente e sistema de aquisição a razão μ entre esses fatores pode ser determinada uma única vez (Zhuang e Roth, 1996). Na literatura, assim como existe uma grande variedade de modelos e técnicas de calibração de câmeras, também existem vários trabalhos que tratam da calibração dos fatores de escala e de sua razão (Motta *et.al.*, 2001, Motta, 1999, Trucco e Verri, 1998, Klette *et. al.*, 1996, Zhuang e Roth, 1996, Tsai, 1987, Lenz e Tsai, 1987).

A grande maioria dessas técnicas envolve o uso de elementos de calibração específicos para a geração de pontos de calibração não coplanares (Trucco e Verri, 1998), ou o uso de sistemas de posicionamento precisos (Zhuang e Roth, 1996, Klette *et. al.*, 1996, Tsai, 1987).

A metodologia adotada aqui para a calibração da razão μ dos fatores de escala é baseada no método proposto por Motta (1999). A escolha foi realizada levando-se em consideração os procedimentos experimentais e os resultados descritos por Motta e McMaster (2002), Motta *et.al.* (2001) e Motta (1999). O método de calibração proposto por Motta (1999), apesar de também necessitar de pontos de calibração não coplanares, pode ser realizado utilizando-se um padrão de calibração plano, sendo necessária somente a movimentação da placa de calibração, gerando uma série de diferentes planos e imagens desses planos.

Seguindo então os procedimentos descritos na Seção 5.1.2, realiza-se a calibração do modelo RAC a partir de um valor inicial de μ . Utilizando-se então o modelo RAC, os resíduos dessa calibração são dados a partir da Equação (2.19) por (Motta *et.al.*, 2001, Zhuang e Roth, 1996, Lenz e Tsai, 1987):

$$\frac{X_d}{Y_d} = \frac{1}{\mu} \frac{r_1^j x_w + r_2^j y_w + r_3^j z_w + t_x}{r_4^j x_w + r_5^j y_w + r_6^j z_w + t_y}, \quad (5.15)$$

de forma que:

$$-Y_d r_1^j x_w - Y_d r_2^j y_w - Y_d r_3^j z_w - Y_d t_x + \mu X_d r_4^j x_w + \mu X_d r_5^j y_w + X_d r_6^j z_w + \mu X_d t_y = 0 \quad (5.16)$$

onde j é o índice representando cada uma das diferentes posições assumidas pela placa de calibração e (X_d, Y_d) os pontos de calibração em pixels que representam a placa nessa posição.

Sabendo-se ainda que $z_w = 0$ (Equação (5.1)):

$$-Y_d r_1^j x_w - Y_d r_2^j y_w - Y_d t_x + \mu X_d r_4^j x_w + \mu X_d r_5^j y_w + \mu X_d t_y = 0 \quad (5.17)$$

Minimizando-se então o resíduo obtido para cada uma das imagens da placa de calibração utilizando a Equação (5.17) é possível calibrar o valor de μ .

Uma vez calibrado o valor de μ é possível repetir-se todo o procedimento, utilizando-se o novo valor de μ para a calibração dos parâmetros $[r_1, r_2, t_x, r_4, r_5, t_y]$ e os novos parâmetros calculados para a calibração de um novo valor de μ , tendo-se então um método iterativo de calibração da razão μ dos fatores de escala, conforme pode ser observado na Figura 5.2.

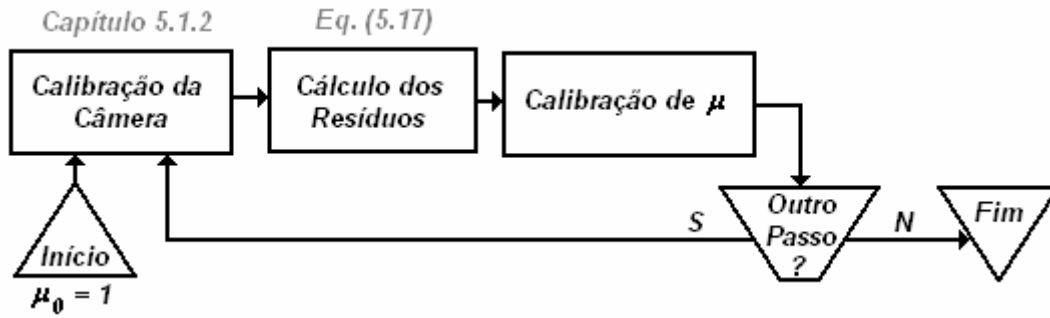


Figura 5.2. Calibração Iterativa da Razão dos Fatores de Escala.

5.1.4 Obtenção do Centro da Imagem

O centro da imagem, definido como as coordenadas (C_x, C_y) em pixels onde o eixo óptico e o plano da imagem se interceptam é um parâmetro de grande importância para o modelo de restrição radial. Ele influencia diretamente não apenas as equações de perspectiva, mas também é o centro da modelagem de distorção radial para as lentes e a calibração precisa desse ponto é uma etapa de grande importância para a realização de medições tridimensionais precisas utilizando-se uma câmera (Lenz e Tsai, 1987).

Lenz e Tsai (1987) apresentam um estudo detalhado da influência de erros na determinação do centro da imagem e propõe diferentes métodos para a determinação precisa desse ponto. Aqui adota-se um desses métodos que baseia-se no uso das equações de restrição de alinhamento radial. A escolha desse método foi realizada devido ao fato deste utilizar a mesma configuração física baseada em uma placa coplanar de pontos de calibração utilizada nas outras etapas de calibração realizadas.

A restrição de alinhamento radial estipula que, conforme ilustrado na Figura 2.3, os vetores \overrightarrow{OP} , $\overrightarrow{OP_u}$ e $\overrightarrow{OP_d}$ estão radialmente alinhados. Logo, com um conjunto coplanar de j pontos de calibração $P_j = (x_w^j, y_w^j, z_w^j)$ cujas coordenadas em pixels na imagem são dadas por (X^j, Y^j) , tem-se a partir da Equação (2.19) que:

$$\begin{cases} \frac{X^j - C_x}{1 - kr^2} - f_x \frac{r_1^j x_w^j + r_2^j y_w^j + r_3^j z_w^j + t_x}{r_7^j x_w^j + r_8^j y_w^j + r_9^j z_w^j + t_z} = 0 \\ \frac{Y^j - C_y}{1 - kr^2} - f_y \frac{r_4^j x_w^j + r_5^j y_w^j + r_6^j z_w^j + t_y}{r_7^j x_w^j + r_8^j y_w^j + r_9^j z_w^j + t_z} = 0 \end{cases} \quad (5.18)$$

, onde

$$r^2 = \mu^2 (X^j - C_x)^2 + (Y^j - C_y)^2 \quad (5.19)$$

Entretanto, devido a incertezas e erros na modelagem do sistema e nas medições para a calibração, a Equação (5.18) normalmente não é nula (Motta *et.al.*, 2001, Zhuang e Roth, 1996, Lenz e Tsai, 1987). Pode-se então tratar essa equação como uma equação de resíduo em cada ponto de calibração e, através de uma técnica de otimização não-linear, achar os valores de (C_x, C_y) que minimizam esse resíduo.

A calibração do centro da imagem (C_x, C_y) e da razão μ dos fatores de escalas aqui apresentada são procedimentos independentes. Esse tipo de estratégia pode ser adotada devido ao fato de que os parâmetros (C_x, C_y) e μ são relativamente independentes entre si, de forma que uma escolha inapropriada para um deles não prejudica a correta calibração do valor ótimo do outro (Zhuang e Roth, 1996).

5.2 CALIBRAÇÃO DO SISTEMA ÓPTICO

Uma vez que a câmera é calibrada utilizando-se a placa metrológica ilustrada na Figura 3.16, todos os parâmetros intrínsecos e extrínsecos desta estão completamente determinados e é possível a calibração do sistema óptico a partir desses parâmetros.

O primeiro passo para a calibração do sistema óptico é a definição dos parâmetros que devem ser determinados. Os parâmetros necessários para a reconstrução 3D de uma superfície foram apresentados na Seção 3.1 e podem ser observados detalhadamente na Figura 5.3.

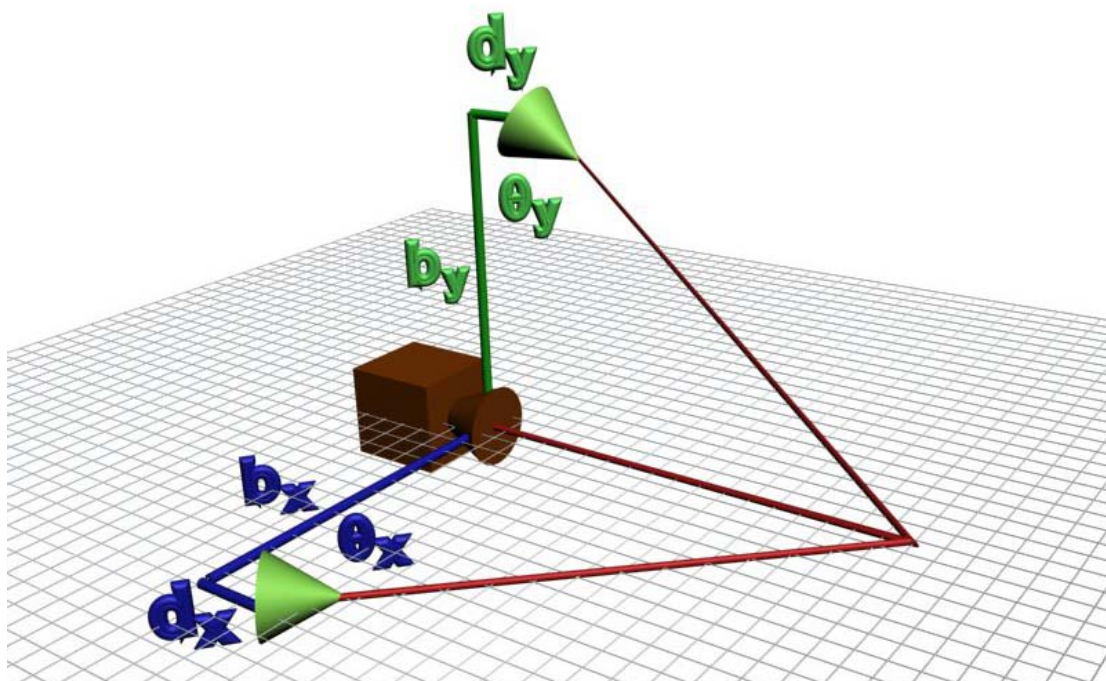


Figura 5.3. Parâmetros do Sistema de Digitalização.

A etapa inicial da calibração é realizada utilizando-se os parâmetros R e T que definem a transformação do sistema de referência global para o sistema de referência da câmera para determinar a equação do plano que caracteriza a placa de calibração.

Inicialmente, para a realização da calibração da câmera através do algoritmo RAC esse plano é definido em relação a um referencial global qualquer como:

$$z_w = 0 \quad (5.20)$$

Logo, sabendo que R e T , definidos na Equação (2.3) e na Equação (2.4), realizam a transformação desse sistema global para o sistema de referência da câmera é possível determinar a equação desse plano em relação à câmera:

$$0x_w + 0y_w + 1z_w + 0 = 0 \xrightarrow{R,T} Ax_c + By_c + Cz_c + D = 0 \quad (5.21)$$

Transformando inicialmente o vetor normal do plano $z_w = 0$ para o sistema de coordenadas da câmera têm-se:

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_3 \\ r_6 \\ r_9 \end{bmatrix} \quad (5.22)$$

Já para D , sabendo-se que o ponto $[T_x \ T_y \ T_z]^T$ pertence ao plano de calibração tem-se:

$$AT_x + BT_y + CT_z + D = 0 \rightarrow D = -r_3T_x - r_6T_y - r_9T_z \quad (5.23)$$

De forma que o plano de calibração em relação ao referencial da câmera fica completamente definido como:

$$Ax_c + By_c + Cz_c + D = 0 \quad (5.24)$$

, onde

$$\begin{aligned} A &= r_3 \\ B &= r_6 \\ C &= r_9 \\ D &= -r_3T_x - r_6T_y - r_9T_z \end{aligned} \quad (5.25)$$

O passo seguinte para a calibração do sistema óptico é a determinação dos planos gerados por cada um dos diodos lasers, conforme ilustrado na Figura 5.4, onde V_N^X e V_N^Y representam os vetores normais dos planos gerados e P_L^X e P_L^Y as posições dos diodos lasers em relação à câmera.

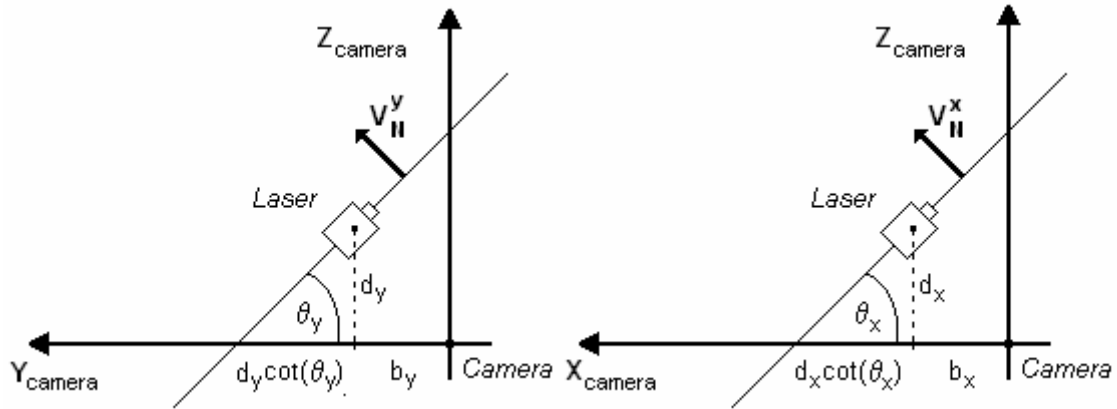


Figura 5.4. Planos Lasers Gerados e Parâmetros do Sistema.

Logo, as equações desses planos serão dadas por:

$$\begin{cases} V_N^Y = [0 & 1 & \cot(\theta_y)] \\ P_L^Y = [0 & b_y & d_y] \end{cases} \Rightarrow y_c + \cot(\theta_y)z_c - b_y - d_y \cot(\theta_y) = 0 \quad (5.26)$$

$$\begin{cases} V_N^X = [1 & 0 & \cot(\theta_x)] \\ P_L^X = [b_x & 0 & d_x] \end{cases} \Rightarrow x_c + \cot(\theta_x)z_c - b_x - d_x \cot(\theta_x) = 0 \quad (5.27)$$

Conhecendo-se a equação de cada um dos planos lasers gerados (Equação (5.26) e Equação (5.27)) e da placa de calibração (Equação (5.24)) é possível determinar a interseção desses planos com a placa de calibração. Essas interseções são as projeções dos planos de luz laser sobre a superfície da placa e são matematicamente descritas como retas no espaço.

Para o plano definido pela Equação (5.26) tem-se:

$$\begin{cases} Ax_c + By_c + Cz_c + D = 0 \\ y_c + \cot(\theta_y)z_c - b_y - d_y \cot(\theta_y) = 0 \end{cases} \quad (5.28)$$

Escolhendo-se x_c como parâmetro livre obtém-se a solução desse sistema que é a equação paramétrica da reta de interseção desses planos.

$$\begin{cases} x_c = t \\ y_c = \frac{A \cot(\theta_y)}{C - B \cot(\theta_y)} x_c + \frac{Cb_y + Cd_y \cot(\theta_y) + D \cot(\theta_y)}{C - B \cot(\theta_y)} \\ z_c = \frac{-A}{C - B \cot(\theta_y)} x_c + \frac{-Bb_y - Bd_y \cot(\theta_y) - D}{C - B \cot(\theta_y)} \end{cases} \quad (5.29)$$

Da mesma maneira, para o plano de luz descrito pela Equação (5.27):

$$\begin{cases} Ax_c + By_c + Cz_c + D = 0 \\ x_c + \cot(\theta_x)z_c - b_x - d_x \cot(\theta_x) = 0 \end{cases} \quad (5.30)$$

$$\begin{cases} x_c = \frac{B \cot(\theta_x)}{C - A \cot(\theta_x)} y_c + \frac{Cb + Cd_x \cot(\theta_x) + D \cot(\theta_x)}{C - A \cot(\theta_x)} \\ y_c = t \\ z_c = \frac{-B}{C - A \cot(\theta_x)} y_c + \frac{-Ab - Ad_x \cot(\theta_x) - D}{C - A \cot(\theta_x)} \end{cases} \quad (5.31)$$

A escolha dos parâmetros livres t nas equações de interseção dos planos é feita tendo como base as equações dos planos de luz laser para evitar divisões por zero, uma vez que é possível que os valores de x_c e y_c sejam constantes nos planos de luz paralelos aos eixos x e y respectivamente.

Obtendo-se então as coordenadas (X_i, Y_i) em pixels de um ponto da linha laser na imagem e sabendo que esse ponto pertence ao plano definido pela placa de calibração é possível obter as coordenadas (x_c, y_c, z_c) desse ponto em relação à câmera utilizando-se as equações do modelo da câmera apresentadas na Seção 2.1.4 e da equação do plano da placa de calibração:

$$\begin{cases} \frac{X_i - C_x}{1 - kr^2} = f_x \frac{x_c}{z_c} \\ \frac{Y_i - C_y}{1 - kr^2} = f_y \frac{y_c}{z_c} \\ Ax_c + By_c + Cz_c + D = 0 \\ r^2 = \mu^2 (X_i - C_x)^2 + (Y_i - C_y)^2 \end{cases} \quad (5.32)$$

Solucionando esse sistema, as coordenadas (x_c, y_c, z_c) do ponto da linha laser podem ser obtidas diretamente como:

$$\begin{cases} x_c = -\frac{A_x D}{AA_x + BB_y + C} \\ y_c = -\frac{B_y D}{AA_x + BB_y + C}, \\ z_c = -\frac{D}{AA_x + BB_y + C} \end{cases} \quad (5.33)$$

onde:

$$\begin{aligned} A_x &= \frac{1}{f_x} \frac{X_i - C_x}{1 - kr^2} \\ B_y &= \frac{1}{f_y} \frac{Y_i - C_y}{1 - kr^2} \\ r^2 &= \mu^2 (X_i - C_x)^2 + (Y_i - C_y)^2 \end{aligned} \quad (5.34)$$

Logo, utilizando-se essa coordenada (x_c, y_c, z_c) obtida e a equação da reta da projeção do plano laser no espaço (Equação (5.29) e Equação (5.31)) é possível obter-se um sistema linear de equações para $(b, \cot(\theta), d \cot(\theta))$:

$$\begin{bmatrix} Ax_c + By_c + D & C & C \\ Bz_c & -B & -B \end{bmatrix} \begin{bmatrix} \cot(\theta_y) \\ b_y \\ d_y \cot(\theta_y) \end{bmatrix} = \begin{bmatrix} Cy_c \\ Ax_c + Cz_c + D \end{bmatrix} \quad (5.35)$$

$$\begin{bmatrix} Ax_c + By_c + D & C & C \\ Az_c & -A & -A \end{bmatrix} \begin{bmatrix} \cot(\theta_x) \\ b_x \\ d_x \cot(\theta_x) \end{bmatrix} = \begin{bmatrix} Cx_c \\ By_c + Cz_c + D \end{bmatrix} \quad (5.36)$$

É bem claro nesses sistemas o fato de que as colunas 2 e 3 são idênticas, ou seja, independente do número de pontos utilizados o sistema sempre terá posto 2. Logo, os valores de desalinhamento d_x e d_y não podem ser obtidos diretamente desses sistemas.

Para a calibração de d_x e d_y , são utilizadas duas ou mais posições diferentes do laser móvel e o valor de $d \cot(\theta)$ e b são determinados conjuntamente. Alterando-se os sistemas da Equação (5.35) e da Equação (5.36):

$$\begin{bmatrix} Ax_c + By_c + D & C \\ Bz_c & -B \end{bmatrix} \begin{bmatrix} \cot(\theta_y) \\ b_y' \end{bmatrix} = \begin{bmatrix} Cy_c \\ Ax_c + Cz_c + D \end{bmatrix} \quad (5.37)$$

$$\begin{bmatrix} Ax_c + By_c + D & C \\ Az_c & -A \end{bmatrix} \begin{bmatrix} \cot(\theta_x) \\ b_x' \end{bmatrix} = \begin{bmatrix} Cx_c \\ By_c + Cz_c + D \end{bmatrix} \quad (5.38)$$

,onde

$$\begin{cases} b_x' = b_x + d_x \cot(\theta_x) \\ b_y' = b_y + d_y \cot(\theta_y) \end{cases} \quad (5.39)$$

Para a solução desses sistemas um único ponto da linha laser é suficiente, entretanto, a utilização de vários pontos da linha laser e uma otimização baseada em mínimos quadrados através da decomposição de valores singulares (SVD) é realizada visando um resultado mais preciso.

Logo, com N diferentes posições do laser móvel, N valores são obtidos para a cotangente e para a distância base do laser adicionada do termo de desalinhamento $d \cot(\theta)$, sendo então possível determinar a distância base real do diodo laser e seu valor de desalinhamento através de um sistema sobre-determinado, ficando todos os parâmetros do laser completamente calibrados:

$$\begin{cases} b_x + d_x^1 \cot(\theta_x) = b_x' \\ b_x + d_x^2 \cot(\theta_x) = b_x' \\ \vdots \\ b_x + d_x^N \cot(\theta_x) = b_x' \end{cases} \quad (5.40)$$

$$\begin{cases} b_y + d_y^1 \cot(\theta_y) = b_y' \\ b_y + d_y^2 \cot(\theta_y) = b_y' \\ \vdots \\ b_y + d_y^N \cot(\theta_y) = b_y' \end{cases} \quad (5.41)$$

Já para a calibração do laser fixo, o mesmo procedimento é realizado, entretanto, para o laser fixo, uma vez que seu ângulo de inclinação é constante, basta a determinação da distância base aparente b' .

Todo o procedimento de calibração do sistema óptico pode ser resumido através dos algoritmos ilustrados na Figura 5.5 e na Figura 5.6.

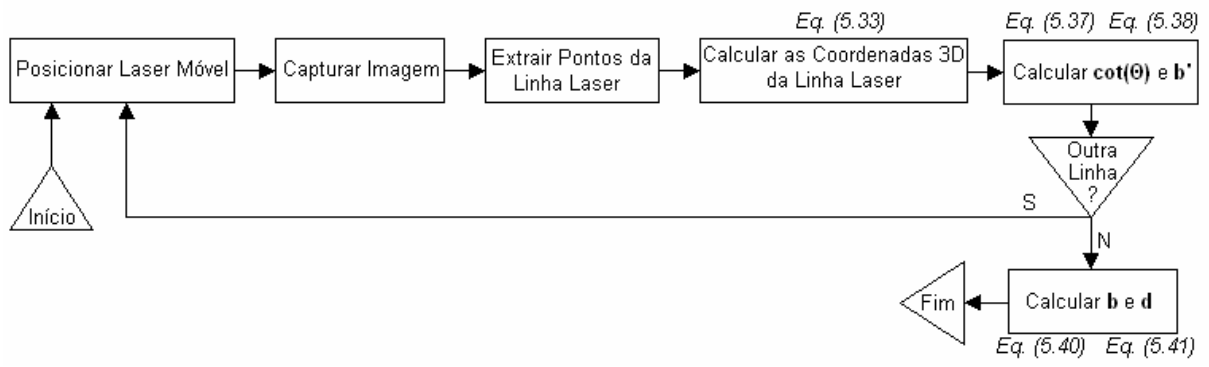


Figura 5.5. Algoritmo de Calibração do Laser Móvel.

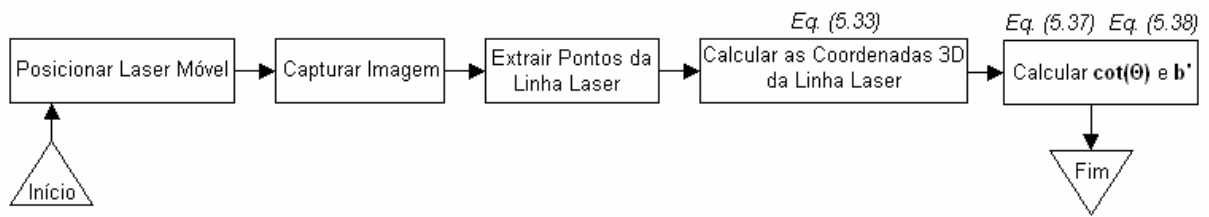


Figura 5.6. Algoritmo de Calibração do Laser Fixo.

6 SIMULAÇÃO

A simulação computacional é uma ferramenta importante para a avaliação e desenvolvimento de sistemas, facilitando as etapas de projeto e especificação de componentes.

Visando ter uma ferramenta de avaliação do sistema VISSCAN-3D, uma plataforma de simulação computacional foi desenvolvida.

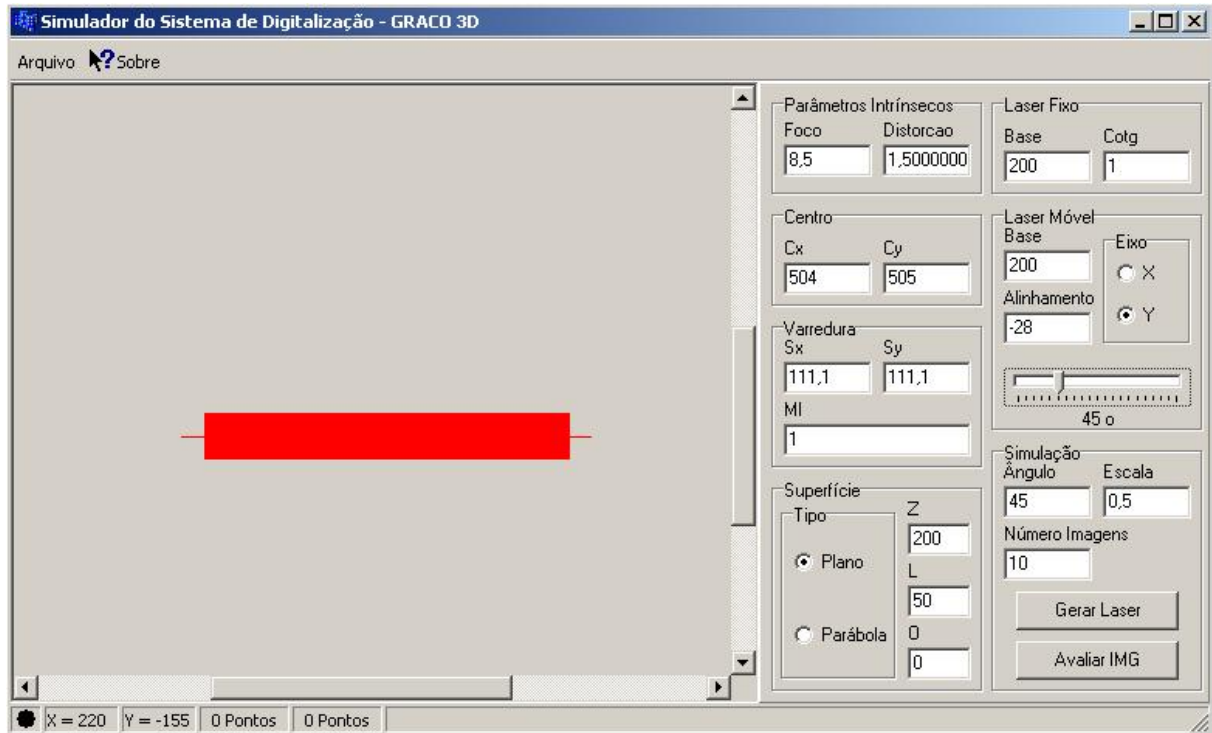


Figura 6.1. Plataforma de Simulação do Sistema VISSCAN-3D.

Nesse capítulo são apresentados cada um dos componentes dessa plataforma e sua importância para a simulação do sistema.

6.1 A CÂMERA RAC VIRTUAL

O conceito de câmera virtual é um conceito utilizado largamente em *softwares* de modelagem 3D e de realidade virtual (Barata e Santos, 1999, Neider et.al., 1993). Uma câmera virtual é um modelo matemático que engloba todo o conjunto de equações que realizam o mapeamento de um ponto tridimensional em um pixel de uma imagem, simulando assim o funcionamento de uma câmera real.

O modelo de câmera utilizado para o desenvolvimento do sistema VISSCAN-3D foi o modelo de restrição de alinhamento radial (Tsai, 1987), e, por essa razão, a câmera virtual desenvolvida para implementação de uma plataforma de simulação segue o mesmo modelo. Conforme esse modelo de câmera apresentado na Seção 2.1.4 um ponto $P = (x_c, y_c, z_c)$ representado no sistema de referência da

câmera é, através das equações de perspectiva, mapeado em um ponto bidimensional $P_u = (X_u, Y_u)$ no plano da imagem como:

$$\begin{cases} X_u = f_x \frac{x_c}{z_c} \\ Y_u = f_y \frac{y_c}{z_c} \end{cases} \quad (6.1)$$

A esse ponto é ainda necessário adicionar-se o efeito de distorção radial causado pelas lentes da câmera e definido pelo parâmetro k .

$$\begin{cases} X_d = (1 - kr^2)X_u \\ Y_d = (1 - kr^2)Y_u \end{cases}, \quad (6.2)$$

onde

$$r^2 = \mu^2 X_d^2 + Y_d^2 \quad (6.3)$$

Ou seja, a intensidade dessa distorção depende diretamente da distância do ponto distorcido até o centro da imagem. Entretanto, diferentemente da câmera real onde o ponto (X_d, Y_d) é conhecido e deseja-se compensar a distorção da lente para se obter o ponto (X_u, Y_u) , na câmera virtual somente o ponto (X_u, Y_u) é conhecido a priori, não sendo possível o cálculo direto da distância radial do ponto.

Para resolver esse problema, Klette *et.al.* (1996) propõem um cálculo iterativo onde cada iteração i é dada por:

$$\begin{cases} {}^0r^2 = \mu^2 X_u^2 + Y_u^2 \\ \begin{cases} {}^i r^2 = \mu^2 {}^i X_d^2 + {}^i Y_d^2 \\ {}^i X_d = (1 - k^i r^2)X_u \\ {}^i Y_d = (1 - k^i r^2)Y_u \end{cases} \end{cases} \quad (6.4)$$

Klette *et.al.* (1996) comenta que com cerca de oito iterações já é obtida uma boa precisão.

De posse agora do ponto $P_d = (X_d, Y_d)$ a última operação a ser realizada é o cálculo do ponto $P_i = (X_i, Y_i)$ em pixels cujo centro passa a ser o pixel $(0,0)$ e não mais o ponto (C_x, C_y) .

$$\begin{aligned} X_i &= X_d + C_x \\ Y_i &= -Y_d + C_y \end{aligned} \quad (6.5)$$

Esse conjunto de equações representa matematicamente a câmera virtual, definindo completamente a transformação de mapeamento de um ponto tridimensional em um pixel em uma imagem e é o elemento central da plataforma de simulação do sistema VISSCAN-3D.

6.2 O DIODO LASER VIRTUAL

Os diodos lasers atuam no sistema VISSCAN-3D através da geração de dois planos de luz. Esses planos de luz, conforme apresentado na Seção 5.2, são dados por:

$$y_c + \cot(\theta_y)z_c - b_y - d_y \cot(\theta_y) = 0 \Rightarrow {}^yA_L x_c + {}^yB_L y_c + {}^yC_L z_c + {}^yD_L = 0 \quad (6.6)$$

$$x_c + \cot(\theta_x)z_c - b_x - d_x \cot(\theta_x) = 0 \Rightarrow {}^xA_L x_c + {}^xB_L y_c + {}^xC_L z_c + {}^xD_L = 0 \quad (6.7)$$

Essas equações representam matematicamente todos os pontos de luz que fazem parte dos planos gerados por cada um dos diodos lasers do sistema e são responsáveis pela simulação desses diodos.

6.3 SUPERFÍCIES VIRTUAIS

A representação matemática de uma superfície consiste simplesmente em sua representação através de uma função na forma:

$$z = f(x, y) \quad (6.8)$$

A plataforma de simulação do sistema VISSCAN-3D traz somente uma superfície virtual implementada, a superfície de um plano. O número de superfícies disponibilizadas pela plataforma pode ser facilmente ampliado utilizando-se a mesma estratégia de simulação apresentada a seguir.

6.3.1 Plano Virtual

Um plano no espaço pode ser definido matematicamente como (Kreyzig, 2006, Edwards e Penney, 1998, Swokowsky, 1996, Alves, 1996):

$$Ax + By + Cz + D = 0, \quad (6.9)$$

onde $[A \ B \ C]^T$ representa o vetor normal desse plano e D é dado por:

$$Ax + By + Cz + D = 0, \quad (6.10)$$

onde $P_0 = [x_0 \ y_0 \ z_0]^T$ é um ponto pertencente ao plano.

A plataforma de simulação do sistema VISSCAN-3D utiliza três parâmetros para a geração de um plano virtual. Esses parâmetros, ilustrados na Tabela 6.1, definem um plano centrado no eixo óptico da câmera conforme ilustrado na Figura 6.2.

Tabela 6.1. Parâmetros para Geração do Plano Virtual.

Parâmetros	Descrição
z_0	Distância base da câmera ao plano
L	Comprimento do plano no eixo X
α	Ângulo de rotação do plano em torno do eixo Y

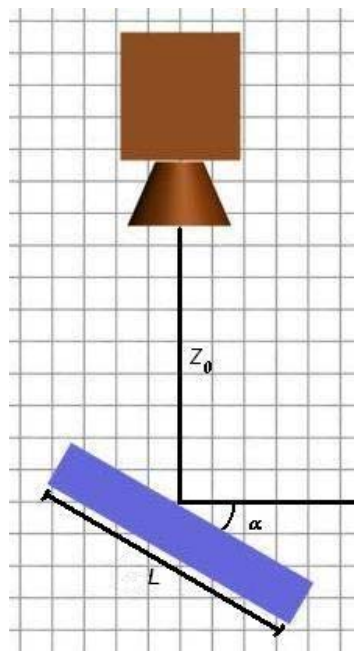


Figura 6.2. Geração de um Plano Virtual para Simulação.

Logo, a equação que descreve essa superfície será:

$$\begin{cases} \sin(\alpha)x + \cos(\alpha)z + \cos(\alpha)z_0 = 0 \\ -\frac{L}{2}\cos(\alpha) \leq x \leq \frac{L}{2}\cos(\alpha) \end{cases} \quad (6.11)$$

6.4 SIMULAÇÃO DO PROCESSO DE DIGITALIZAÇÃO

O processo de simulação da digitalização é realizado através da projeção dos planos lasers sobre a superfície virtual. Definida a angulação do laser móvel, cada um dos planos lasers é projetado, resultando em duas curvas no espaço. Essas curvas são em seguida avaliadas conjuntamente para a determinação do ponto de cruzamento entre elas. Esse ponto, ainda representado no espaço tridimensional da câmera, é então processado pela câmera virtual, resultando no pixel que representa o cruzamento das linhas lasers projetadas sobre a superfície virtual. A seguir, somente a projeção do plano de luz que representa o diodo laser móvel é considerado e, para cada ponto (x, y, z) dessa linha, o ponto (X_i, Y_i) na imagem é calculado utilizando a câmera virtual.

6.4.1 Digitalização do Plano Virtual

A digitalização do plano virtual inicializa-se com a projeção dos planos de luz laser sobre o plano virtual. Resolvendo-se a projeção desses lasers obtém-se duas retas no espaço:

$$\begin{cases} {}^y A_L x_c + {}^y B_L y_c + {}^y C_L z_c + {}^y D_L = 0 \\ A x_c + B y_c + C z_c + D = 0 \end{cases} \Rightarrow \begin{cases} x_c = t \\ y_c = {}^y m_y t + {}^y n_y \\ z_c = {}^z m_y t + {}^z n_y \end{cases} \quad (6.12)$$

$$\begin{cases} {}^x A_L x_c + {}^x B_L y_c + {}^x C_L z_c + {}^x D_L = 0 \\ A x_c + B y_c + C z_c + D = 0 \end{cases} \Rightarrow \begin{cases} x_c = {}^x m_x x + {}^x n_x \\ y_c = t \\ z_c = {}^z m_x x + {}^z n_x \end{cases}, \quad (6.13)$$

com

$$\begin{aligned} {}^y m_y &= \frac{{}^y D_L C_L - C^y D_L}{{}^y B_L C - B^y C_L} \\ {}^y n_y &= \frac{A^y C_L - C^y A_L}{{}^y B_L C - B^y C_L} \end{aligned} \quad (6.14)$$

$$\begin{aligned} {}^z m_y &= \frac{B^y A_L - A^y B_L}{{}^y B_L C - B^y C_L} \\ {}^z n_y &= \frac{B^y D_L - D^y B_L}{{}^y B_L C - B^y C_L} \end{aligned} \quad (6.15)$$

$$\begin{aligned} {}^x m_x &= \frac{B^y C_L - C^y B_L}{{}^y A_L C - A^y C_L} \\ {}^x n_x &= \frac{D^y C_L - C^y D_L}{{}^y A_L C - A^y C_L} \end{aligned} \quad (6.16)$$

$$\begin{aligned} {}^z m_x &= \frac{A^y B_L - B^y A_L}{{}^y A_L C - A^y C_L} \\ {}^z n_x &= \frac{A^y D_L - D^y A_L}{{}^y A_L C - A^y C_L} \end{aligned} \quad (6.17)$$

Uma vez de posse das equações dessas retas, calcula-se o ponto de interseção entre elas.

$$\begin{aligned} x_{int} &= \frac{{}^x m_x {}^y n_y + {}^x n_x}{{}^x m_x {}^y m_y - 1} \\ y_{int} &= \frac{{}^x n_x {}^y m_y + {}^y n_y}{{}^x m_x {}^y m_y - 1} \\ z_{int} &= \frac{{}^z m_y {}^x m_x {}^y n_y + {}^z m_y {}^x n_x + {}^z n_y - {}^x m_x {}^y m_y {}^z n_y}{{}^x m_x {}^y m_y - 1} \end{aligned} \quad (6.18)$$

Utilizando-se então a câmera virtual essas coordenadas tridimensionais são convertidas em coordenadas bidimensionais em pixels da imagem.

Em seguida utilizando-se somente a equação que representa a projeção do plano de luz do diodo laser móvel (Equação (6.12) ou Equação (6.13)) Essa equação é particionada em uma seqüência de pontos dentro do intervalo $\left[-\frac{L}{2}\cos(\alpha), \frac{L}{2}\cos(\alpha)\right]$ e cada um desses pontos é processado utilizando-se a câmera virtual.

Ao final desse procedimento obtem-se uma imagem sintética da projeção do laser sobre essa superfície virtual e um arquivo no formato PTO com esses pontos.

6.5 VALIDAÇÃO DO PROCESSO DE DIGITALIZAÇÃO

Concluído o desenvolvimento da plataforma de simulação do sistema VISSCAN-3D foi realizada uma série de simulações para demonstrar a funcionalidade do sistema e validar seu funcionamento. Para tanto, realizou-se a simulação de um plano virtual (Figura 6.2) em várias posições e, uma vez que ao gerar-se o plano virtual tem-se acesso às coordenadas tridimensionais de cada um dos pontos

digitalizados fechou-se um círculo e comparou-se os resultados obtidos pelo sistema VISSCAN-3D com os dados originais.

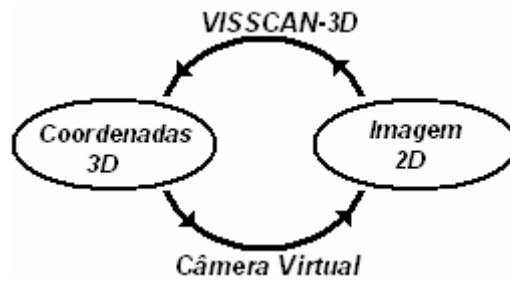


Figura 6.3. Simulação e Validação do Sistema VISSCAN-3D.

Seguindo-se essa estratégia de avaliação, foram utilizadas 25 imagens de um plano virtual, dispostas em 5 diferentes posições em 5 diferentes distâncias de trabalho (Z) conforme ilustrado na Figura 6.4. Cada imagem formada por 5000 pontos, e as coordenadas calculadas pelo sistema VISSCAN-3D foram comparadas com as coordenadas de entrada do sistema de simulação conforme a Equação (6.19)

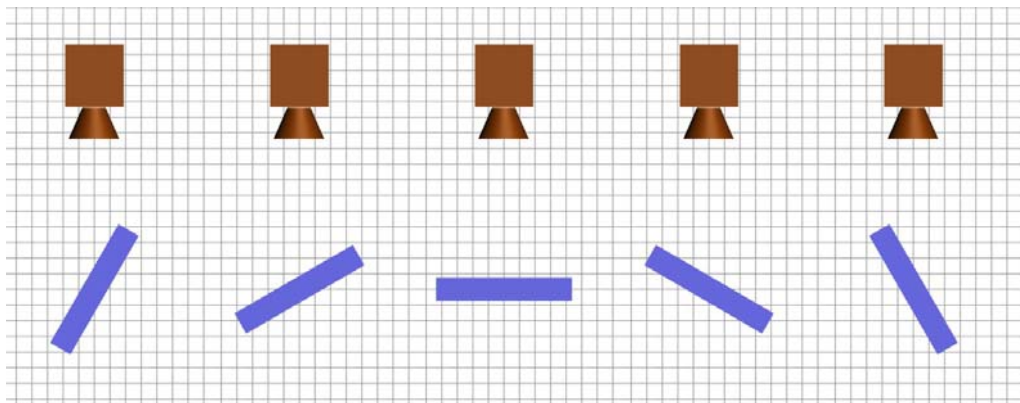


Figura 6.4. Diferentes Orientações do Plano Virtual.

$$\left\{ \begin{array}{l} E_x = |x_{calculado} - x_{original}| \\ E_y = |y_{calculado} - y_{original}| \\ E_z = |z_{calculado} - z_{original}| \\ E_T = \sqrt{E_x^2 + E_y^2 + E_z^2} \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \bar{E} = \frac{\sum_{i=1}^m E_i}{m} \\ E_{\max} = \max(E_i) \\ \sigma = \sqrt{\frac{\sum_{i=1}^m (E_i - \bar{E})^2}{m}} \end{array} \right. , \quad (6.19)$$

onde m é o número de pontos e E_T o erro total da digitalização.

Observando-se os resultados obtidos em função da distância base (Z) da câmera ao plano virtual têm-se as seguintes curvas de erro:

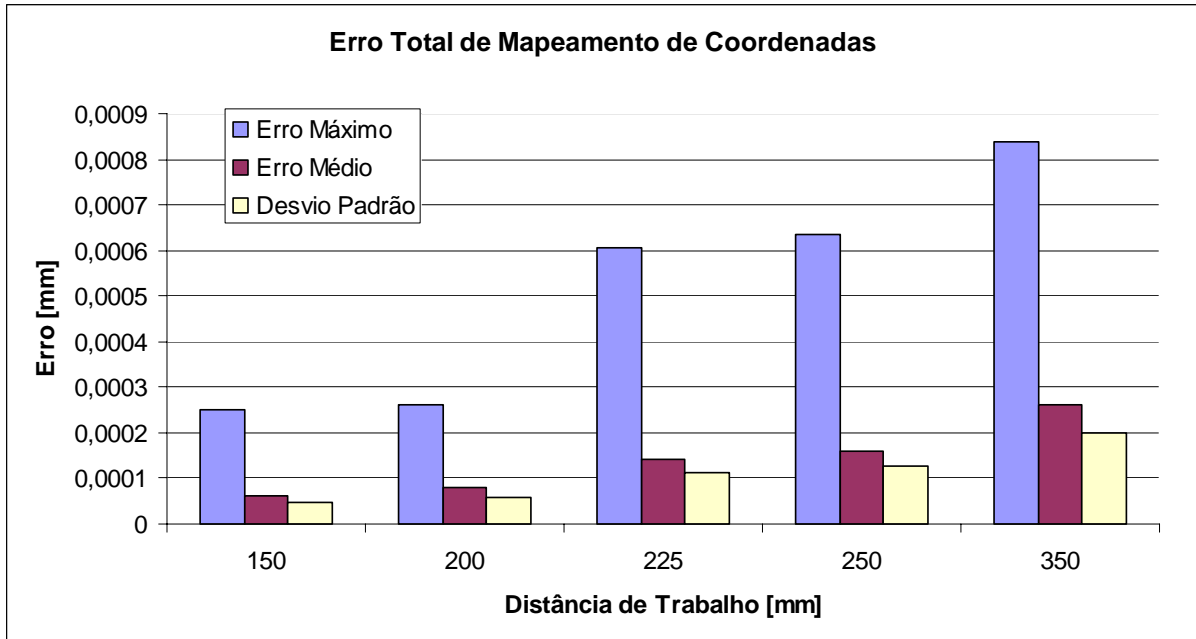


Figura 6.5. Erro Total de Mapeamento na Validação do Sistema.

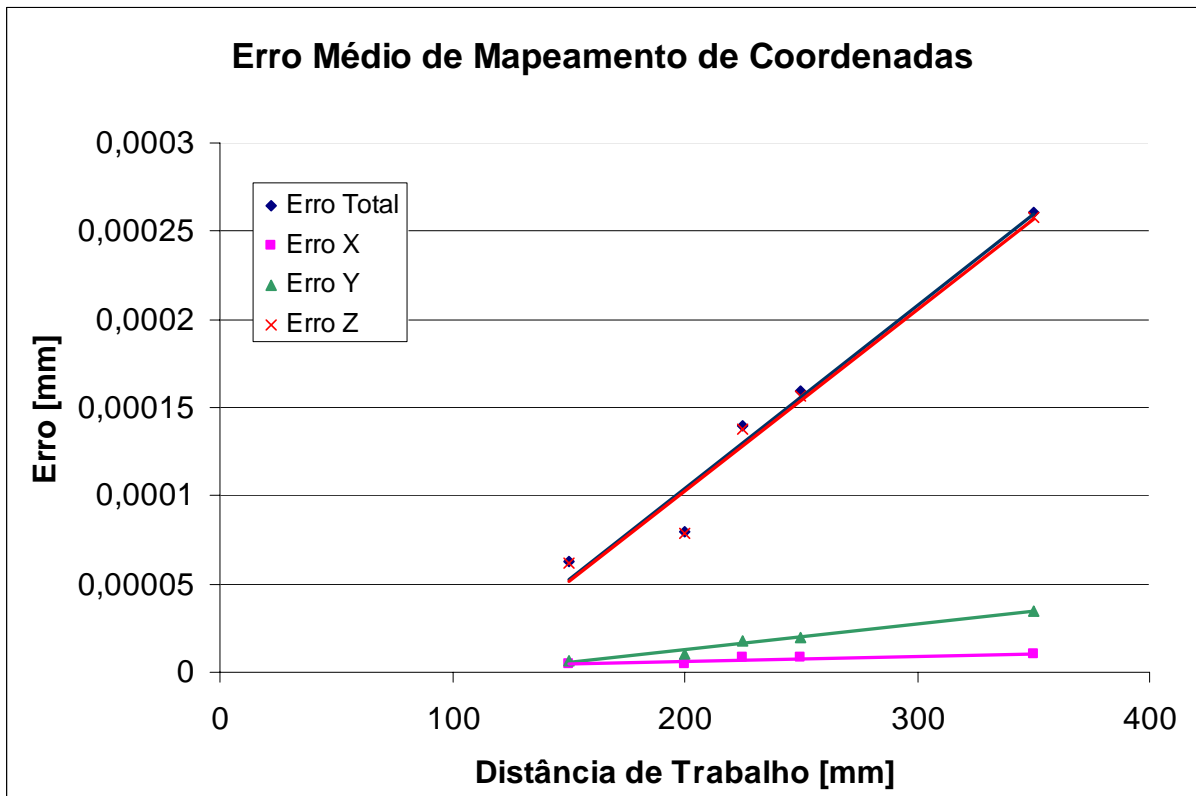


Figura 6.6. Erro Médio de Mapeamento na Validação do Sistema.

Conforme esperado, esses valores de erro, apesar de pequenos (na ordem de $10^{-4} mm$), não são nulos. Isso se deve à precisão numérica dos cálculos realizados e à representação em ponto flutuante das grandezas envolvidas. Todo o sistema VISSCAN-3D foi desenvolvido utilizando-se precisão simples (32 bits), o que gera uma precisão de 24 bits, aproximadamente 7 casas decimais.

Os resultados ilustrados na Figura 6.5 e na Figura 6.6 não apenas confirmam a validade das equações apresentadas na Seção 3.1 e o funcionamento do processo de digitalização, como também demonstram a influência que a precisão numérica exerce sobre sistemas de precisão.

Existe um grande número de publicações (Dinechin e Villard, 2006, Sufroniou e Spaletta, 2005, Alves, 1996) que tratam da precisão numérica e da propagação de erros na representação computacional de números reais, entretanto nessa etapa de desenvolvimento do sistema VISSCAN-3D não foram realizados estudos aprofundados desse tema.

7 ESTUDOS DE PRECISÃO

Independente do tipo de sistema que é utilizado para a medição de um parâmetro a precisão é um fator crítico (Alves, 1996, Petkovic, 1989).

Aqui é apresentado um estudo teórico das características de precisão do sistema desenvolvido, levando-se em consideração os aspectos ópticos e mecânicos deste, não sendo considerada a influência dos algoritmos de processamento de imagem utilizados para extração da linha laser da imagem e para o cálculo do ponto de cruzamento da projeção dos planos lasers.

Em seguida é apresentado um estudo experimental do desempenho do sistema através de uma série de experimentos para que os resultados obtidos com cada uma das diferentes abordagens sejam comparados e analisados.

7.1 PRECISÃO TEÓRICA DO SISTEMA

O sistema VISSCAN-3D é um sistema de digitalização baseado em visão computacional. Isso implica que sua resolução e precisão estão diretamente associadas não somente aos algoritmos de processamento e extração de dados das imagens, mas, principalmente, à resolução e precisão do sistema óptico utilizado para a captura dessas imagens e a todas as variáveis que envolvem essa captura.

O primeiro passo para realizar uma avaliação da resolução e precisão do sistema de digitalização como um todo é a determinação da resolução do sistema óptico utilizado e, a partir desses dados e das características particulares do sistema é possível realizar-se uma avaliação do sistema de digitalização e seu desempenho do ponto de vista metrológico.

7.1.1 Resolução do Sistema Óptico

A resolução de um sistema óptico é uma medida de sua capacidade de reproduzir detalhes dos objetos visualizados (Edmund Optics, 2007, Edmund Optics, 2004). Na Figura 7.1 é possível observar o significado dessa definição.

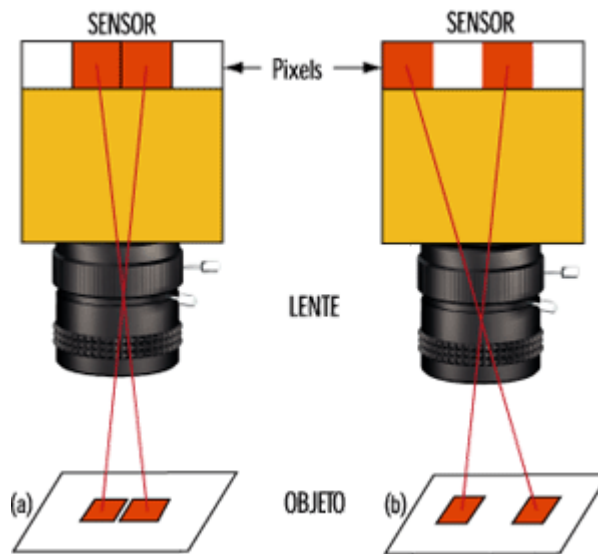


Figura 7.1. Resolução de um Sistema Óptico.

Para que um sistema óptico seja capaz de diferenciar entre dois objetos é necessário que exista um espaço mínimo entre eles. Em uma câmera digital esse espaço mínimo é definido com o tamanho de um pixel, ou seja, para que dois objetos sejam diferenciáveis entre si é necessário que ao mínimo um pixel represente cada objeto e que exista um pixel entre eles. Logo, a resolução de uma câmera digital, em unidades S.I., pode ser definida como:

$$Res_C = 2 \cdot (\text{Dimensão do Pixel}), \quad (7.1)$$

onde a dimensão do pixel é um valor normalmente fornecido pelos fabricantes de câmeras.

Antes de determinar a resolução do sistema óptico como um todo é ainda necessário conhecer o fator de amplificação das lentes utilizadas pelo sistema. Esse fator é definido como:

$$PMAG = \frac{\text{Dimensão do Sensor}}{FOV}, \quad (7.2)$$

onde FOV é o campo de visão da câmera (*Field of View*).

Esse fator de magnificação representa o trabalho feito pela lente sobre a imagem, e ilustra o efeito que essa lente causa sobre um objeto, aumentando ou reduzindo seu tamanho real.

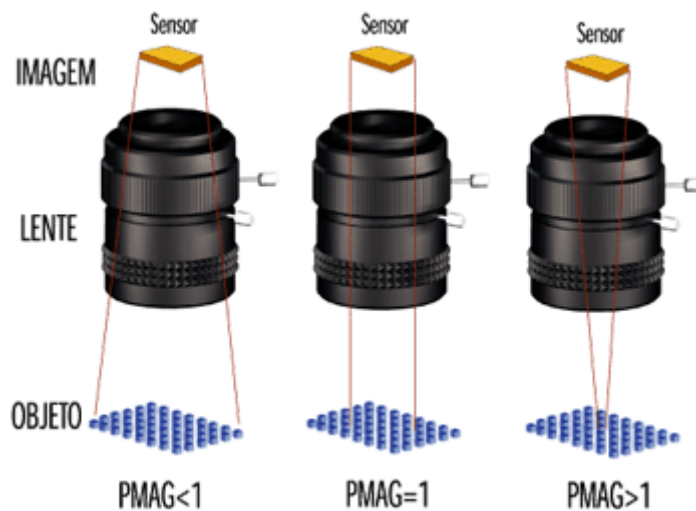


Figura 7.2. Fator de Amplificação de uma Lente.

O fator de amplificação primário de uma lente ($PMAG$) é um parâmetro comumente fornecido por fabricantes (Edmund Optics, 2007, Edmund Optics, 2004), entretanto, muitas vezes esse fator é fornecido como o ângulo de abertura da lente para um determinado tamanho padrão de sensor CCD.

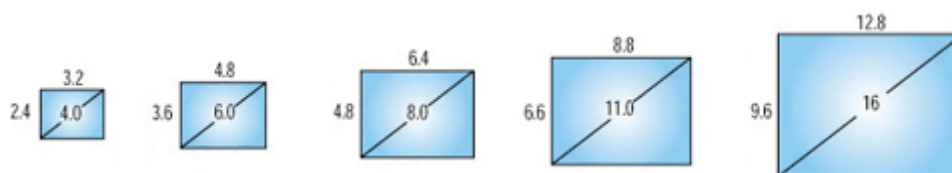


Figura 7.3. Tamanhos Padrões de Sensores CCD (Unidades em mm).

Sabendo-se o ângulo de abertura da lente é possível determinar o campo de visão em função da distância de trabalho e, conseqüentemente, o fator de amplificação primária da lente.

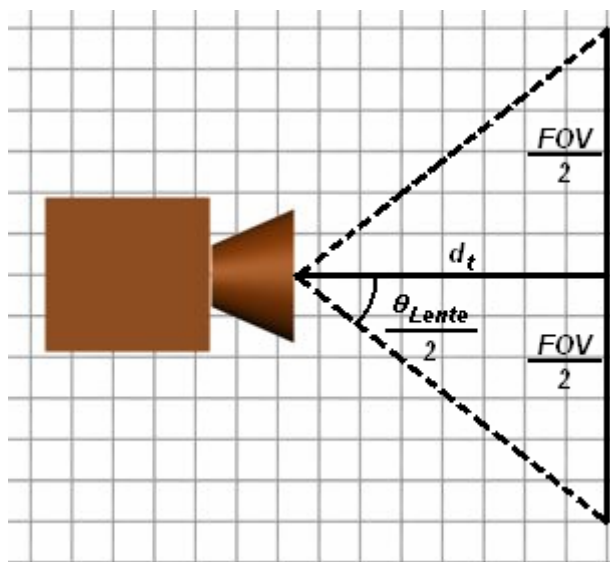


Figura 7.4. Determinação do Campo de Visão do Sistema Óptico.

$$FOV = 2k_{Sensor} \tan\left(\frac{\theta_{Lente}}{2}\right)d_t \quad (7.3)$$

, com

$$k_{Sensor} = \left(\frac{\text{Dimensão do Sensor}}{\text{Dimensão do Sensor Padrão}} \right) \quad (7.4)$$

O fator k_{Sensor} é responsável pela correção do campo de visão da câmera (FOV) em função do tamanho do sensor CCD utilizado, convertendo o campo de visão calculado com o fator de amplificação de um sensor de um tamanho diferente para o campo de visão do sensor utilizado.

Uma vez conhecidos os valores de $PMAG$ e de Res_C , a resolução global do sistema óptico pode ser definida como:

$$Res_o = \frac{Res_C}{PMAG} \quad (7.5)$$

A resolução do sistema óptico define a mínima característica de um objeto que pode ser observada, e conseqüentemente digitalizada, pelo sistema.

A partir do campo de visão da câmera é ainda possível calcular-se a área de trabalho da câmera:

$$A_{Trabalho} = FOV_{Vertical} FOV_{Horizontal} \quad (7.6)$$

Utilizando-se então as equações definidas, e conhecendo-se os parâmetros das câmeras e lentes utilizadas por cada um dos protótipos implementados, é possível obter os valores de resolução para ambos os sistemas ópticos construídos. Na Tabela 7.1 e na Tabela 7.2 são apresentados os parâmetros fornecidos pelos fabricantes e os parâmetros calculados para cada uma das câmeras utilizadas.

Tabela 7.1. Parâmetros da Câmera Utilizada no Sistema de Campo.

Câmera Lumenera LW230	
Número de Pixels [px]	1616 x 1216
Dimensões do Sensor CCD [mm]	7.1 x 5.4
Dimensões do Pixel [μm]	4.4 x 4.4
Resolução da Câmera (Res_C) [μm]	8.8 x 8.8

Campo de Visão (FOV) [mm]	$0.86d_t \times 0.87d_t$
Área de Trabalho ($A_{Trabalho}$) [mm^2]	$0.75d_t^2$
Amplificação ($PMAG$) [$Adimensional$]	$8.23/d_t \times 6.17/d_t$
Resolução do Sistema (Res_o) [mm]	$0.0010d_t \times 0.0014d_t$
Lentes MegaPixel	
Comprimento de Foco [mm]	8.0
Ângulo de Abertura	42.5°

Tabela 7.2. Parâmetros da Câmera Utilizada no Sistema de Laboratório.

Câmera Pulnix TM-1020-15CL	
Número de Pixeis [px]	1008×1018
Dimensões do Sensor CCD [mm]	9.1×9.2
Dimensões do Pixel [μm]	9.0×9.0
Resolução da Câmera (Res_c) [μm]	18.0×18.0
Campo de Visão (FOV) [mm]	$1.10d_t \times 1.49d_t$
Área de Trabalho ($A_{Trabalho}$) [mm^2]	$1.64d_t^2$
Amplificação ($PMAG$) [$Adimensional$]	$8.23/d_t \times 6.17/d_t$
Resolução do Sistema (Res_o) [mm]	$0.0022d_t \times 0.0029d_t$
Lentes MegaPixel	
Comprimento de Foco [mm]	8.5
Ângulo de Abertura	42.5°

Os valores de resolução do sistema são dependentes diretamente da distância de trabalho da câmera, ou seja, quanto mais distante o objeto se encontra da câmera, pior é a resolução do sistema óptico, conforme pode ser observado através dos gráficos na Figura 7.5 e na Figura 7.6.

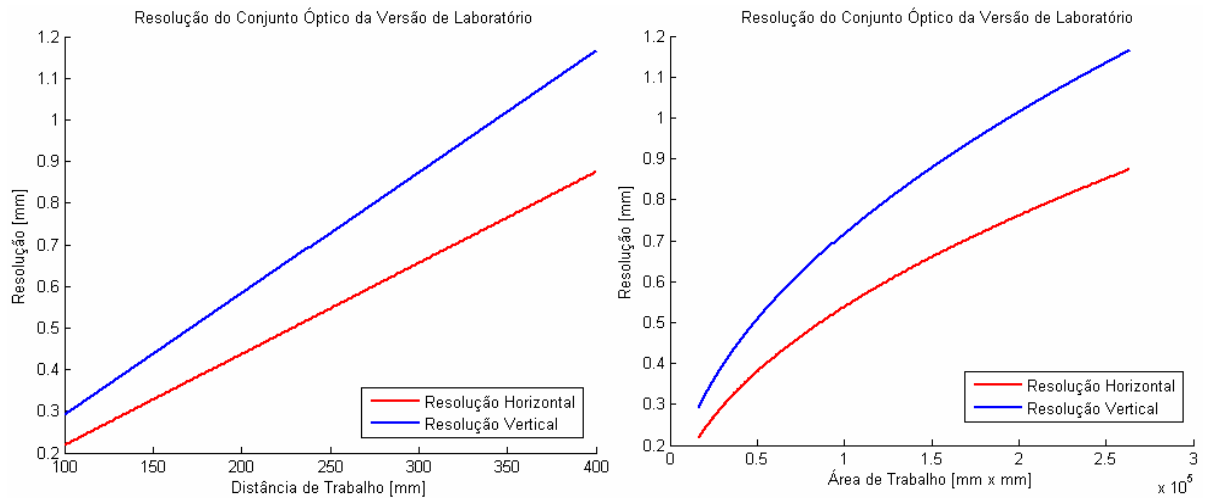


Figura 7.5. Resolução Teórica do Conjunto Óptico da Versão de Laboratório.

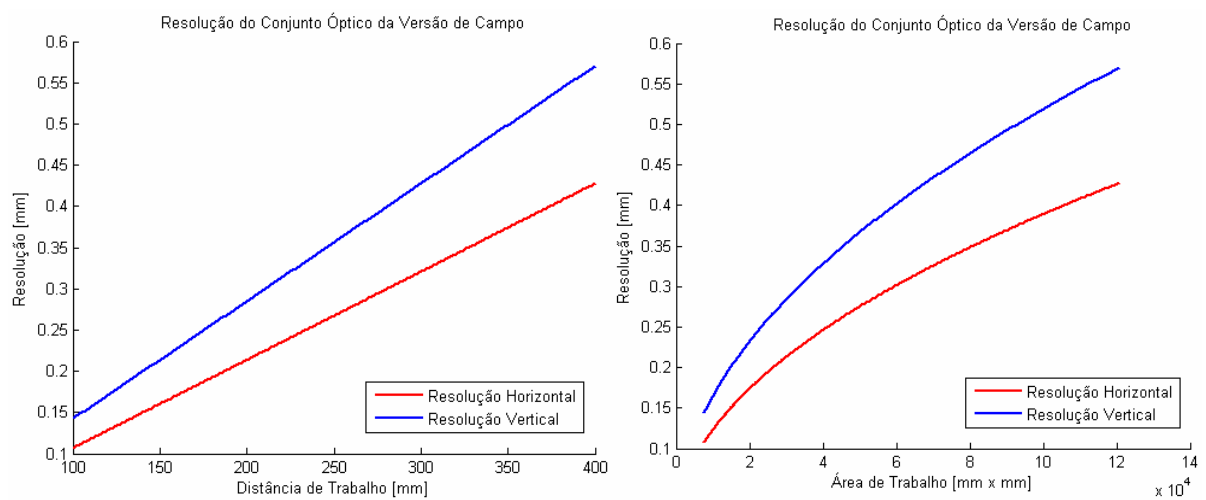


Figura 7.6. Resolução Teórica do Conjunto Óptico da Versão de Campo.

Outra observação que pode ser feita acerca do comportamento do sistema óptico através dos dados e equações apresentados é a variação da área de trabalho do sistema em função da distância de trabalho. Essa distância determina o limite teórico para área máxima que pode ser digitalizada, pois só é possível digitalizar-se a área que o sistema óptico consegue visualizar.

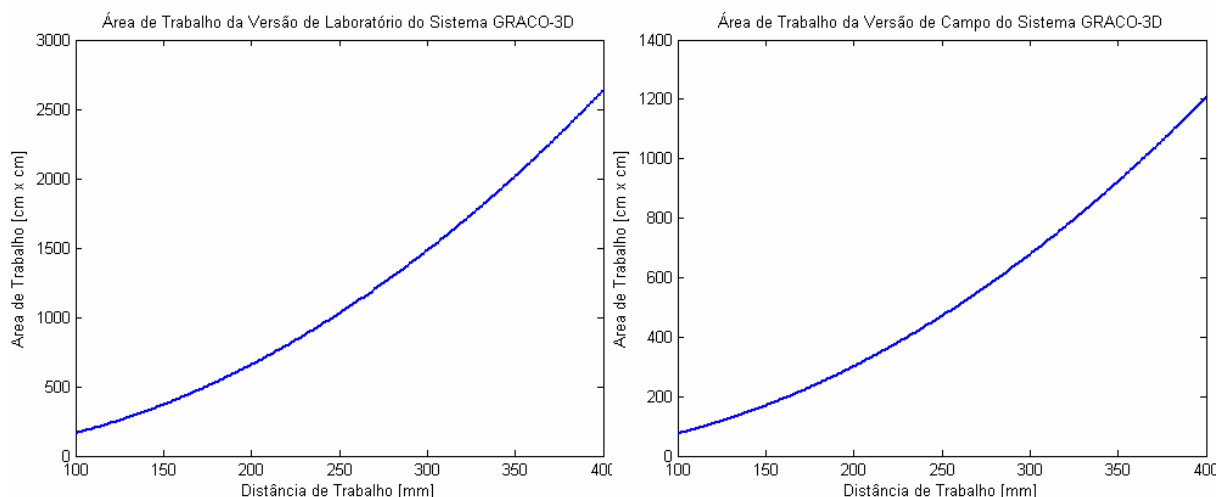


Figura 7.7. Área de Trabalho Teórica do Sistema VISSCAN-3D.

É importante observar que a avaliação da resolução do sistema óptico apresentada aqui não considera o efeito de aberrações ópticas ou da iluminação, que podem vir a prejudicar a resolução do sistema.

7.1.2 Precisão do Sistema de Digitalização

O passo seguinte para realizar uma avaliação teórica do sistema é a derivação das equações de propagação de erros do sistema. Toda a análise apresentada nessa seção é realizada considerando-se como laser móvel o laser paralelo ao plano X, de forma que, conhecendo-se as equação que regem o processo de digitalização através da triangulação ativa (Equação (3.9)) e derivando essa equação obtêm-se as seguintes relações para o erro em cada uma das coordenadas 3D de um ponto qualquer:

$$\begin{cases} \Delta x_c^2 = \left(\frac{\partial x_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial x_c}{\partial y_u}\right)^2 \Delta y_u^2 + \left(\frac{\partial x_c}{\partial f}\right)^2 \Delta f^2 + \left(\frac{\partial x_c}{\partial b_y}\right)^2 \Delta b_y^2 + \left(\frac{\partial x_c}{\partial d_y}\right)^2 \Delta d_y^2 + \left(\frac{\partial x_c}{\partial \cot(\theta_y)}\right)^2 \Delta \cot(\theta_y)^2 \\ \Delta y_c^2 = \left(\frac{\partial y_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial y_c}{\partial y_u}\right)^2 \Delta y_u^2 + \left(\frac{\partial y_c}{\partial f}\right)^2 \Delta f^2 + \left(\frac{\partial y_c}{\partial b_y}\right)^2 \Delta b_y^2 + \left(\frac{\partial y_c}{\partial d_y}\right)^2 \Delta d_y^2 + \left(\frac{\partial y_c}{\partial \cot(\theta_y)}\right)^2 \Delta \cot(\theta_y)^2 \quad (7.7) \\ \Delta z_c^2 = \left(\frac{\partial z_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial z_c}{\partial y_u}\right)^2 \Delta y_u^2 + \left(\frac{\partial z_c}{\partial f}\right)^2 \Delta f^2 + \left(\frac{\partial z_c}{\partial b_y}\right)^2 \Delta b_y^2 + \left(\frac{\partial z_c}{\partial d_y}\right)^2 \Delta d_y^2 + \left(\frac{\partial z_c}{\partial \cot(\theta_y)}\right)^2 \Delta \cot(\theta_y)^2 \end{cases}$$

, onde Δx_u , Δy_u , Δf , Δb_y , Δd_y e $\Delta \cot(\theta_y)$ representam os valores de erro na determinação de cada uma dessas variáveis e parâmetros.

A maioria desses parâmetros é, entretanto, obtida através dos procedimentos de calibração, que utilizam imagens capturadas de uma placa de calibração e dos lasers. Os únicos valores de erro que não estão relacionados diretamente com a calibração são os valores de Δx_u e Δy_u , que dependem diretamente das características da câmera.

Considerando-se então uma calibração ideal dos parâmetros da câmera e do sistema, a Equação (7.7) pode ser simplificada como:

$$\begin{cases} \Delta x_c^2 = \left(\frac{\partial x_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial x_c}{\partial y_u}\right)^2 \Delta y_u^2 \\ \Delta y_c^2 = \left(\frac{\partial y_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial y_c}{\partial y_u}\right)^2 \Delta y_u^2, \\ \Delta z_c^2 = \left(\frac{\partial z_c}{\partial x_u}\right)^2 \Delta x_u^2 + \left(\frac{\partial z_c}{\partial y_u}\right)^2 \Delta y_u^2 \end{cases} \quad (7.8)$$

onde as derivadas são dadas por:

$$\begin{aligned} \frac{\partial x_c}{\partial x_u} &= \frac{b_y + d_y \cot(\theta_y)}{f \cot(\theta_y) + y_u} \\ \frac{\partial y_c}{\partial x_u} &= 0 \\ \frac{\partial z_c}{\partial x_u} &= 0 \end{aligned} \quad (7.9)$$

$$\begin{aligned} \frac{\partial x_c}{\partial y_u} &= -\frac{x_u (b_y + d_y \cot(\theta_y))}{(f \cot(\theta_y) + y_u)^2} \\ \frac{\partial y_c}{\partial y_u} &= \frac{b_y + d_y \cot(\theta_y)}{f \cot(\theta_y) + y_u} - \frac{y_u (b_y + d_y \cot(\theta_y))}{(f \cot(\theta_y) + y_u)^2} \\ \frac{\partial z_c}{\partial y_u} &= -\frac{f (b_y + d_y \cot(\theta_y))}{(f \cot(\theta_y) + y_u)^2} \end{aligned} \quad (7.10)$$

Substituindo-se o valor de z_c conforme a Equação (3.9), as equações de erro do sistema podem ser reescritas como:

$$\begin{cases} \Delta x_c = \sqrt{\left(\frac{z}{f}\right)^2 \Delta x_u^2 + \left(-\frac{z^2}{f^2} \frac{x_u}{(b_y + d_y \cot(\theta_y))}\right)^2} \Delta y_u^2 \\ \Delta y_c = \sqrt{\left(\frac{z}{f} - \frac{z^2}{f^2} \frac{y_u}{(b_y + d_y \cot(\theta_y))}\right)^2} \Delta y_u^2 \\ \Delta z_c = \sqrt{\left(\frac{z^2}{f} \frac{1}{(b_y + d_y \cot(\theta_y))}\right)^2} \Delta y_u^2 \end{cases} \quad (7.11)$$

Observa-se através dessas equações que a precisão com que as coordenadas tridimensionais de um ponto podem ser recuperadas depende não somente da distância que este ponto se encontra da câmera, mas também da posição desse ponto na imagem formada pela câmera e, devido ao desalinhamento d_y , também da inclinação que o laser móvel assume durante a digitalização desse ponto.

Entretanto, de forma a simplificar essas expressões e possibilitar uma análise mais direta da influência de cada um dos parâmetros sobre a precisão do sistema é possível analisar somente a situação mais crítica de angulação do laser e de posicionamento do ponto na imagem. Mesmo que a situação mais crítica não venha a ocorrer ela fornece um limite superior para o erro esperado.

A primeira simplificação que pode ser realizada na Equação (7.11) refere-se aos valores que as coordenadas bidimensionais (x_u, y_u) de um ponto na imagem podem assumir. Através da Equação (7.11) observa-se que o erro do sistema cresce a medida que essas variáveis também crescem. Entretanto, essas coordenadas estão limitadas pelo tamanho da pastilha do sensor CCD e, considerando-se que o centro da imagem é o centro da pastilha tem-se que:

$$\begin{cases} \max x_u = \frac{\text{Largura do Sensor CCD}}{2} \\ \max y_u = \frac{\text{Altura do Sensor CCD}}{2} \end{cases} \quad (7.12)$$

Já para o ângulo de inclinação do laser móvel, observa-se através da Equação (7.11) que esse ângulo influencia o erro final do sistema alterando a distância base b_y' , definida pela Equação (3.8). Quanto menor essa distância base maior é o erro final observado.

De forma a determinar o valor crítico da inclinação do laser móvel é necessário definir-se inicialmente uma faixa de varredura para esse laser. Considerando-se um faixa de 90° , obtem-se uma varredura suficiente para alcançar grande parte do campo de visão de câmeras e lentes comuns (42.5°) para diferentes valores da distância base b_y , conforme ilustrado na Figura 7.8.

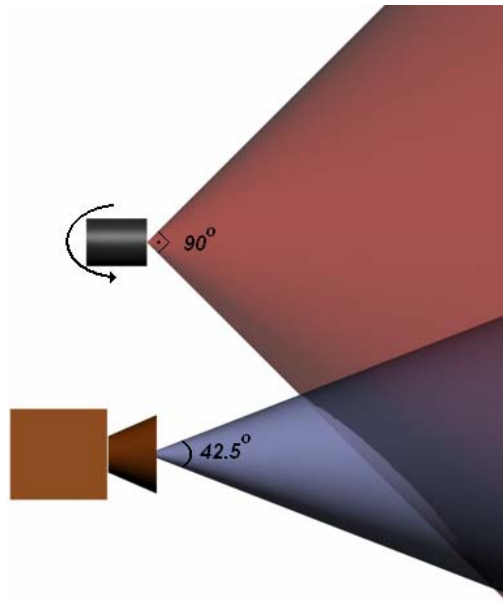


Figura 7.8. Definição da Amplitude de Varredura para Análise de Precisão.

Definida a amplitude do movimento a ser realizado para a varredura do campo de visão da câmera, determinam-se os valores da inclinação do laser que geram essa amplitude conforme ilustrado na Figura 7.9.

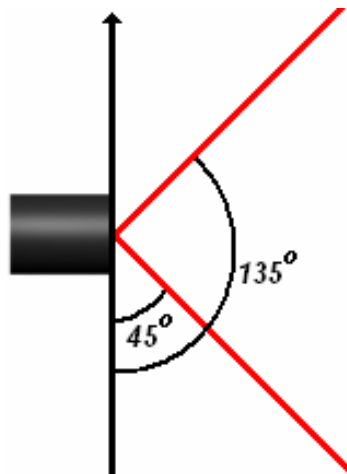


Figura 7.9. Definição das Angulações Máximas para Análise de Precisão.

Concluída essa análise sobre o movimento de varredura do laser, o próximo passo é determinar o valor crítico dessa angulação dentro da faixa especificada, ou seja:

$$\min b_y' = \min(b_y + d_y \cot(\theta_y)) \Big|_{\theta_y=45}^{135} \quad (7.13)$$

Logo, considerando que o desalinhamento do laser é menor que a distância de base e que ambos os valores são positivos:

$$b_y > d_y \geq 0 \Rightarrow \min b_y' = b_y - d_y \quad (7.14)$$

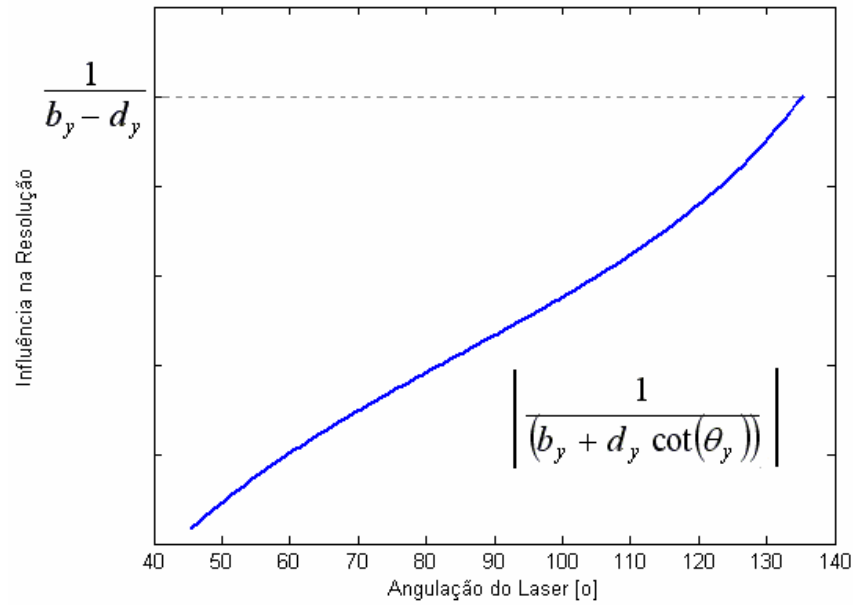


Figura 7.10. Termo da Precisão do Sistema Dependente da Angulação do Laser.

Avaliando a Equação (7.11) nos pontos críticos apresentados, tem-se:

$$\begin{cases} \Delta x_c = \sqrt{\left(\frac{z}{f}\right)^2 \Delta x_u^2 + \left(\frac{z^2}{f^2} \frac{\max x_u}{b_y - d_y}\right)^2 \Delta y_u^2} \\ \Delta y_c = \left| \frac{z}{f} + \frac{z^2}{f^2} \frac{\max y_u}{b_y - d_y} \right| \Delta y_u \\ \Delta z_c = \left| \frac{z^2}{f} \frac{1}{b_y - d_y} \right| \Delta y_u \end{cases} \quad (7.15)$$

Considerando então somente os termos de maior ordem (z^2) da Equação (7.15) obtêm-se as equações que descrevem teoricamente o comportamento do erro do sistema:

$$\begin{cases} \Delta x_c = \left| \frac{z^2}{f^2} \frac{\max x_u}{b_y - d_y} \right| \Delta y_u \\ \Delta y_c = \left| \frac{z^2}{f^2} \frac{\max y_u}{b_y - d_y} \right| \Delta y_u \\ \Delta z_c = \left| \frac{z^2}{f} \frac{1}{b_y - d_y} \right| \Delta y_u \end{cases} \quad (7.16)$$

Uma vez obtidas as equações de propagação de erros do sistema de digitalização é necessário agora determinar os valores de Δx_u e Δy_u . A partir da Equação (2.10), e desconsiderando os erros de aquisição e distorção da câmera, tem-se que:

$$\begin{cases} x_u = dd_x (X_i - C_x) \\ y_u = dd_y (Y_i - C_y) \end{cases} \quad (7.17)$$

Logo, a incerteza de medição associada a x_u e y_u será:

$$\begin{cases} \Delta x_u = dd_x \Delta X_i \\ \Delta y_u = dd_y \Delta Y_i \end{cases} \quad (7.18)$$

Considerando-se que nenhuma espécie de precisão subpixel é utilizada, tem-se que:

$$\begin{cases} \Delta X_i = 0.5\text{pixel} \\ \Delta Y_i = 0.5\text{pixel} \end{cases} \quad (7.19)$$

Logo,

$$\begin{cases} \Delta x_u = \frac{dd_x}{2} \\ \Delta y_u = \frac{dd_y}{2} \end{cases} \quad (7.20)$$

Utilizando-se então os parâmetros dos sistemas implementados em laboratório e em campo, dados pela Tabela 3.1 e pela Tabela 3.2, as curvas de incerteza do sistema em função da distância de trabalho são ilustradas na Figura 7.11.

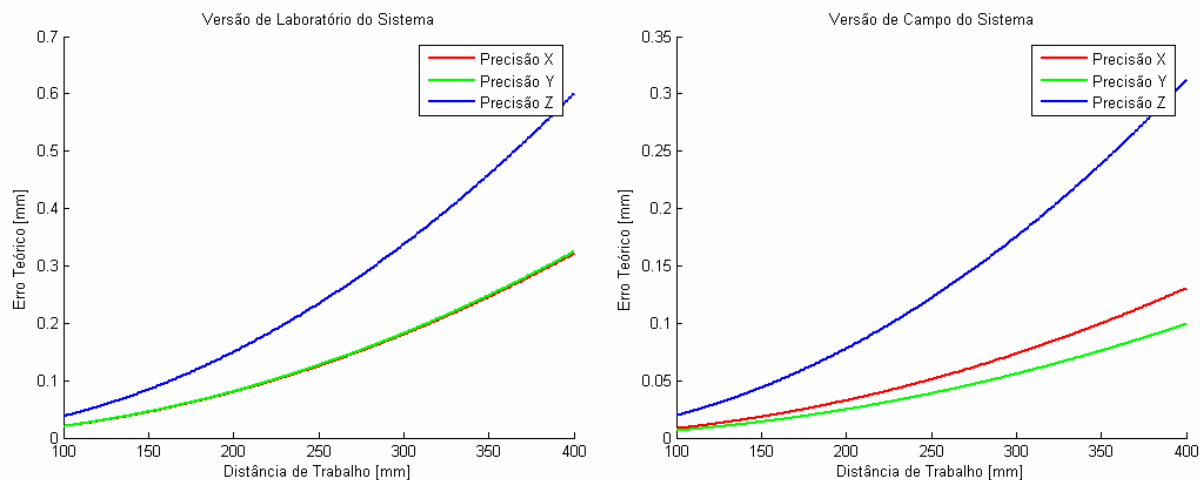


Figura 7.11. Erros Teóricos dos Sistemas de Digitalização Implementados.

Em seguida, de forma a obter-se um valor único de incerteza do sistema, combinam-se os resultados da incerteza de medição do sistema em cada uma das três dimensões de digitalização conforme a Equação (7.21) e obtêm-se os valores de incerteza teórica para a arquitetura utilizada pelo sistema VISSCAN-3D conforme ilustrado na Tabela 7.3.

$$\Delta = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \quad (7.21)$$

Tabela 7.3. Erro Máximo Teórico da Arquitetura de Digitalização.

Arquitetura VISSCAN-3D	
Δx	$\left \frac{z^2}{f^2} \frac{\max x_u}{b_y - d_y} \right \Delta y_u$
Δy	$\left \frac{z^2}{f^2} \frac{\max y_u}{b_y - d_y} \right \Delta y_u$
Δz	$\left \frac{z^2}{f^2} \frac{1}{b_y - d_y} \right \Delta y_u$
Δ	$\left \frac{z^2}{f^2} \frac{\sqrt{\max x_u^2 + \max y_u^2 + f^2}}{b_y - d_y} \right \Delta y_u$

Utilizando-se os parâmetros dos sistemas implementados em laboratório e em campo, a precisão global (Δ) de cada um dos sistemas será:

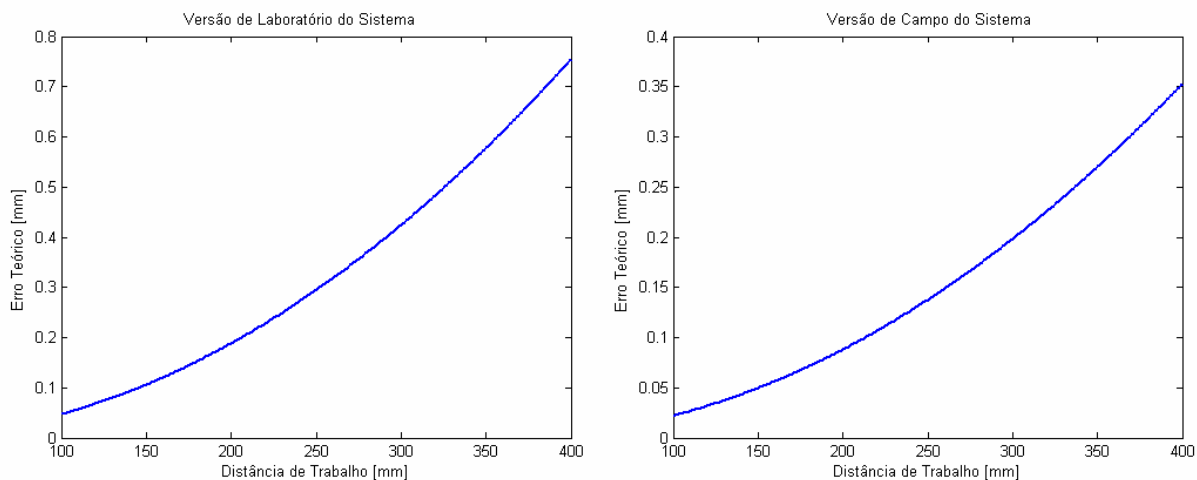


Figura 7.12. Precisão Teórica dos Sistemas de Digitalização Implementados.

O levantamento das equações teóricas para a resolução e para a precisão do sistema VISSCAN-3D, apesar de não refletirem todos os fatores envolvidos no processo de digitalização de uma superfície, são de grande valia para a avaliação do sistema e para se determinar a influência de cada um dos parâmetros do sistema.

Observando-se a Tabela 7.3, onde as equações teóricas de incerteza do sistema estão enumeradas, é possível observar a influência direta do comprimento focal (f) e da distância base do diodo laser móvel (b_y) sobre a incerteza final. Ao avaliar-se esse resultado juntamente com os elementos da triangulação (Figura 3.4), observa-se que o incremento desses parâmetros proporciona um crescimento das arestas dos triângulos envolvidos, o que influencia os fatores de escala envolvidos no processo de triangulação, favorecendo o processo e melhorando sua precisão.

Outra constatação importante que pode ser feita sobre o sistema através das equações de precisão está no caso particular em que o diodo laser móvel assume uma angulação tal que $d_y \cot(\theta_y) = -b_y$. Essa situação, conforme pode também ser observado na Equação (3.9) e na Equação (3.10), geram uma singularidade. Nessa situação ocorre um colapso dos triângulos ilustrados na Figura 3.4, não sendo possível realizar a triangulação e impossibilitando o sistema de recuperar as coordenadas tridimensionais de uma superfície. Esse colapso das equações é observado nas equações de precisão (Equação (7.11)) como uma singularidade onde o erro do sistema tende a ∞ .

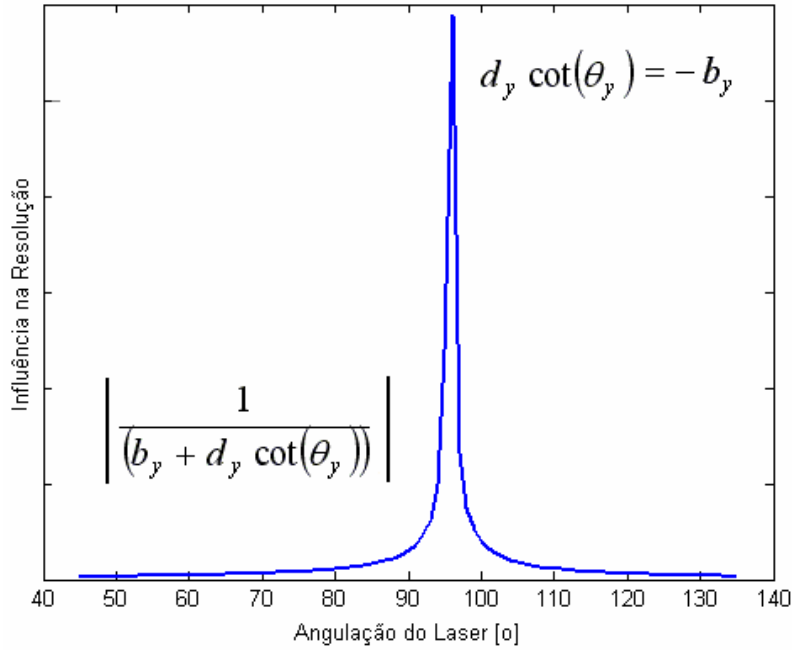


Figura 7.13. Singularidade do Sistema para $d_y \cot(\theta_y) = -b_y$.

Uma forma de evitar essa singularidade é a restrição imposta pela Equação (7.14), pois, uma vez que a angulação assumida pelo diodo móvel esteja restrita ao intervalo $[45^\circ \ 135^\circ]$, tem-se que:

$$\left| \cot(\theta_y) \right|_{45^\circ}^{135^\circ} \leq 1 \quad (7.22)$$

Logo, sendo $|b_y| > |d_y|$ tem-se que $|d_y \cot(\theta_y)| < |b_y|$ e a singularidade não irá ocorrer.

7.2 PRECISÃO EXPERIMENTAL DO SISTEMA

Para se realizar uma avaliação experimental da precisão do sistema é necessária a definição de uma metodologia para a obtenção adequada de dados. Nessa seção os procedimentos utilizados e os resultados obtidos a partir desses são apresentados e detalhados.

Para a execução dos procedimentos experimentais de avaliação da precisão do sistema VISSCAN-3D foi utilizado um bloco metrológico de aço produzido pela empresa ausJENA que possui um comprimento padrão de 50 mm e pode ser observado na Figura 7.14.

O bloco metrológico é posicionado a uma certa distância do sistema de digitalização e é digitalizado cinco vezes, mantendo suas bordas padronizadas na posição vertical e tendo sua orientação alterada em cada escaneamento conforme pode ser observado na Figura 7.15. Em seguida o bloco é posicionado a uma distância maior do sistema de digitalização e todo o procedimento é repetido.

O bloco padrão foi posicionado em cinco diferentes distâncias do sistema de digitalização, totalizando 25 digitalizações.



Figura 7.14. Bloco Metrológico.

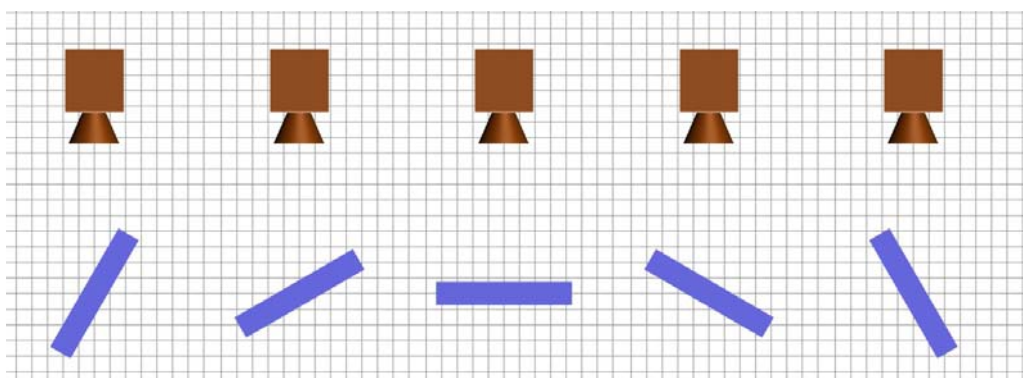


Figura 7.15. Diferentes Orientações para Digitalização do Bloco Metrológico.

A orientação exata do bloco em cada uma das digitalizações não é uma informação necessária. As diferentes orientações são utilizadas para se obter uma grande massa de dados para cada uma das diferentes distâncias entre o sistema de digitalização e o bloco metrológico. Na Figura 7.16 é possível observar uma dessas digitalizações sendo realizada.

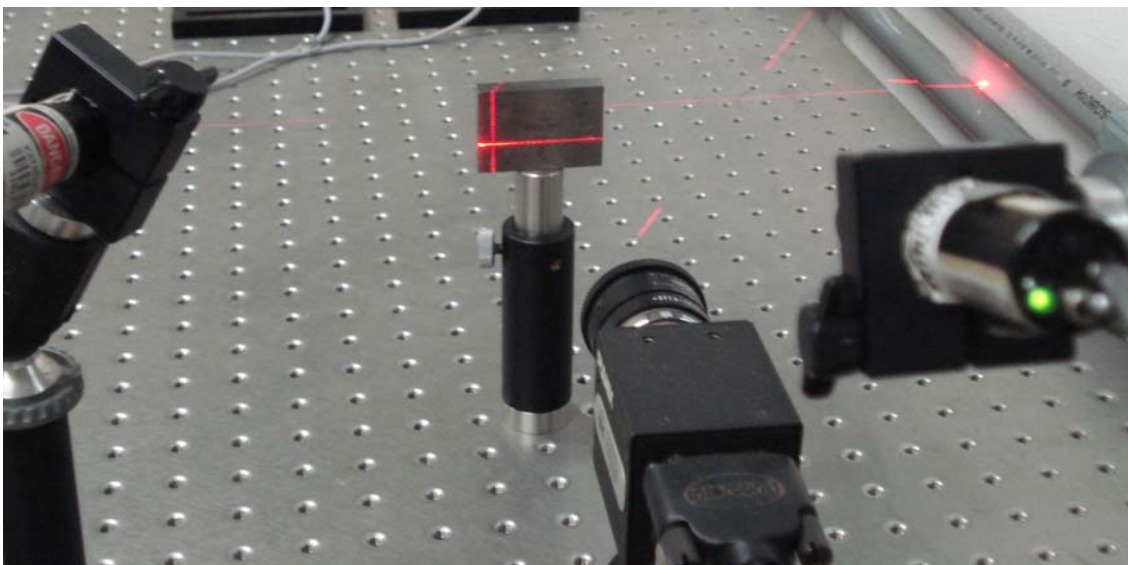


Figura 7.16. Digitalização do Bloco Metrológico.

Uma vez de posse da massa de dados obtida a partir das digitalizações realiza-se a avaliação da precisão do sistema utilizando-se duas estratégias diferentes. O sistema é inicialmente avaliado quanto à correta digitalização da distância entre as duas bordas do bloco metrológico e em seguida é avaliado quanto a planicidade obtida.

O valor padronizado do bloco metrológico utilizado é a distância entre as duas bordas verticais, conforme ilustrado no esquemático da Figura 7.17. Logo, dados dois pontos quaisquer P_1 e P_2 , cada um pertencente a uma das bordas do bloco, e considerando que essas bordas são paralelas ao eixo-Y da câmera, a distância padrão digitalizada é obtida através Equação (7.23)

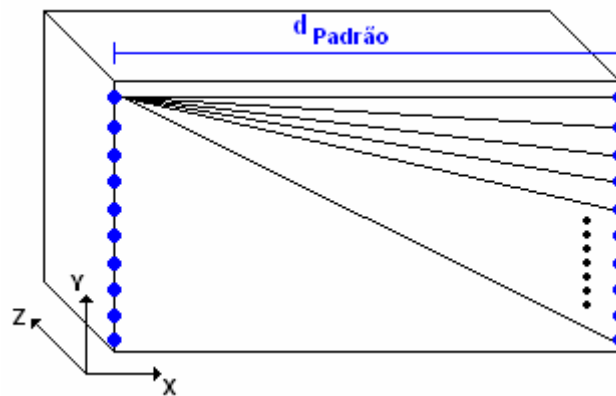


Figura 7.17. Esquemático de Digitalização do Bloco Metrológico e suas Bordas.

$$\begin{cases} P_1 = [x_1 & y_1 & z_1]^T \\ P_2 = [x_2 & y_2 & z_2]^T \end{cases} \Rightarrow d_{\text{medição}} = \sqrt{(x_1 - x_2)^2 + (z_1 - z_2)^2} \quad (7.23)$$

Calcula-se então a distância padrão obtida para cada par de pontos digitalizados, de forma que a partir de N linhas digitalizadas obtêm-se N^2 medições, conforme ilustrado na Figura 7.17.

Cada um desses valores é então comparado com a distância padrão do bloco metrológico e os valores de média e desvio padrão do erro obtido são calculados conforme a Equação (7.24) (Bussab e Morettin, 1987).

$$E_i = |d_{\text{medição}} - d_{\text{padrão}}| \Rightarrow \begin{cases} \bar{E} = \frac{\sum_{i=1}^m E_i}{m} \\ E_{\text{max}} = \max(E_i) \\ \sigma = \sqrt{\frac{\sum_{i=1}^m (E_i - \bar{E})^2}{m}} \end{cases} \quad (7.24)$$

, onde m é o número de medições realizadas.

Essa avaliação é realizada separadamente para cada diferente distância entre o bloco metrológico e o sistema de digitalização e para cada diferente orientação do bloco, de forma que é possível observar a influência da orientação do bloco sobre a incerteza calculada.

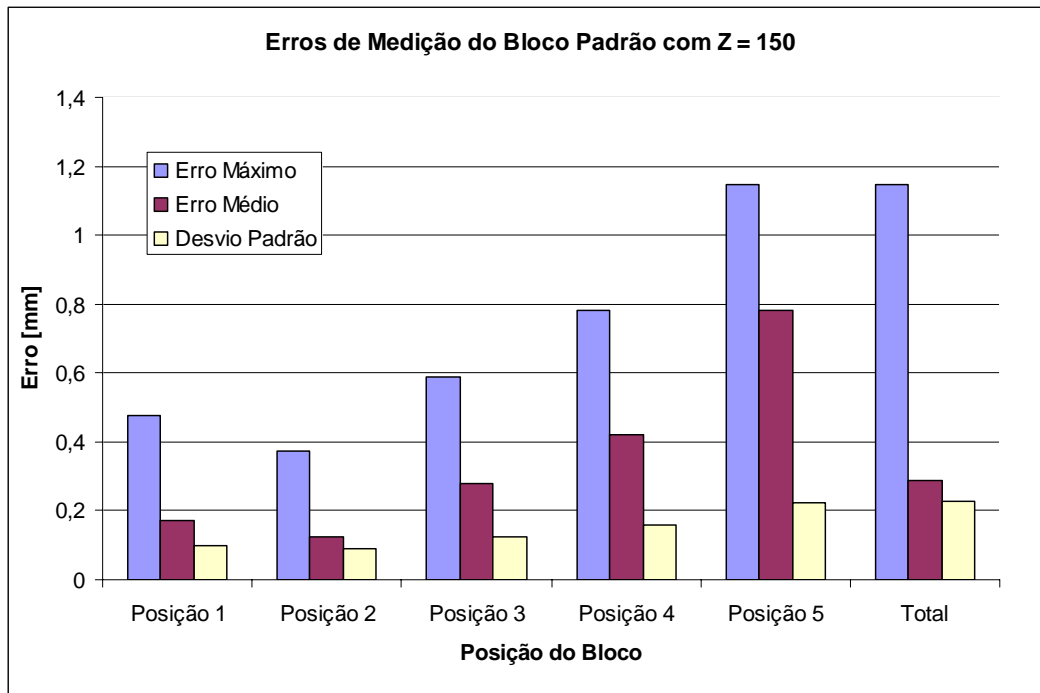


Figura 7.18. Erros de Medição do Bloco Padrão com $Z=150$ para a Versão de Laboratório.

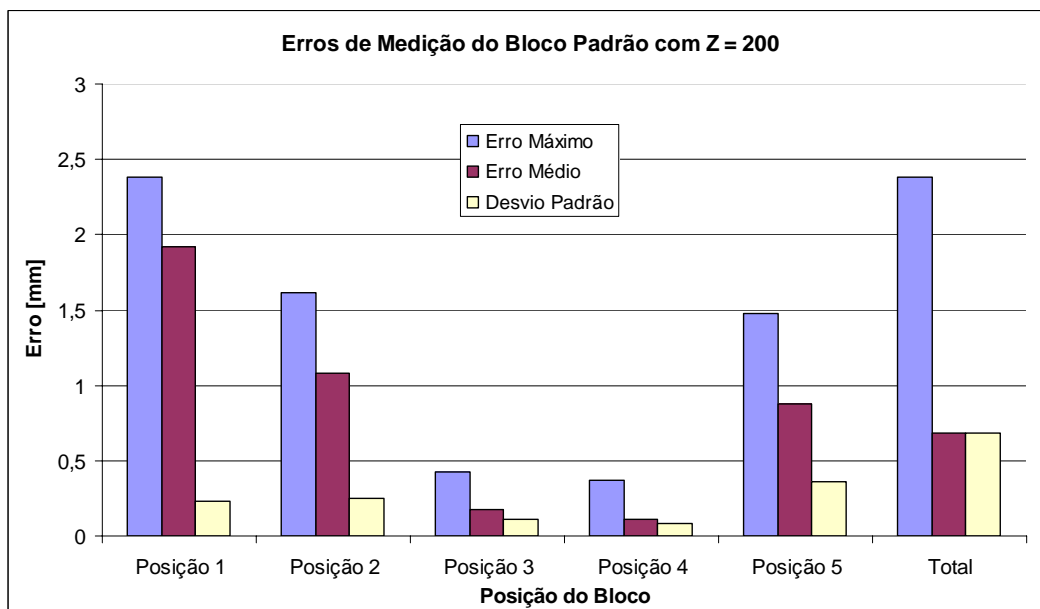


Figura 7.19. Erros de Medição do Bloco Padrão com $Z=200$ para a Versão de Laboratório.

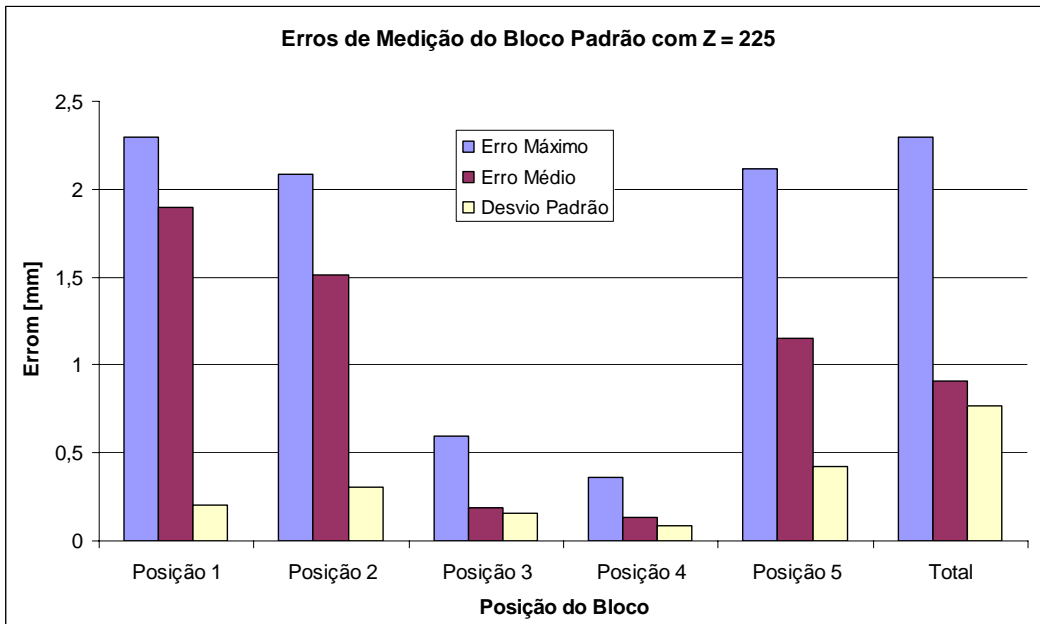


Figura 7.20. Erros de Medição do Bloco Padrão com Z=225 para a Versão de Laboratório.

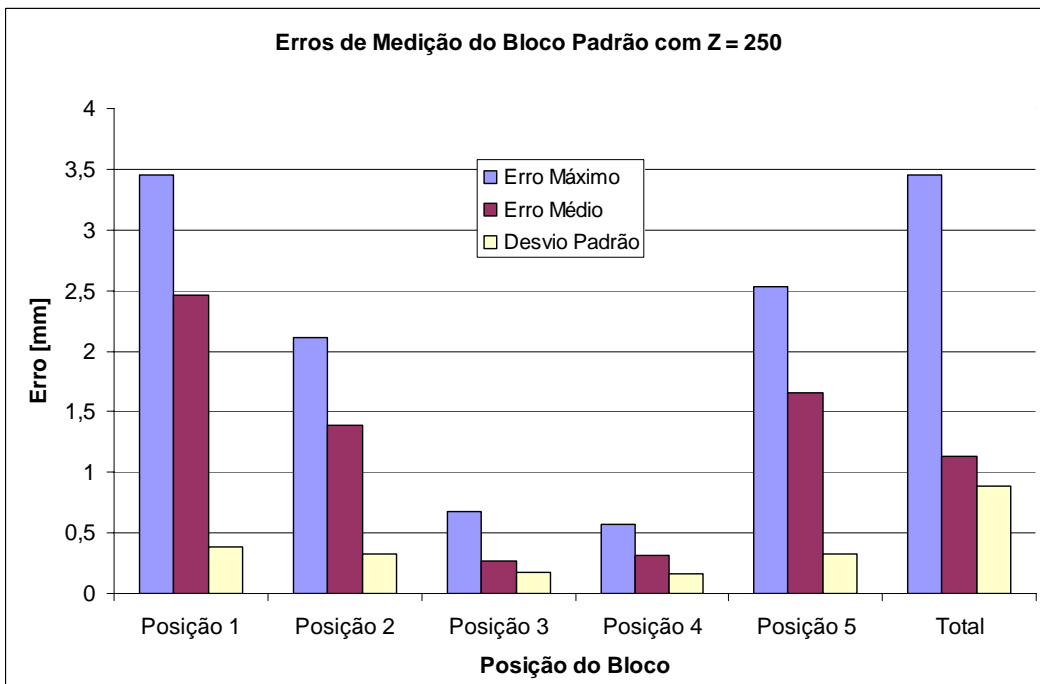


Figura 7.21. Erros de Medição do Bloco Padrão com Z=250 para a Versão de Laboratório.

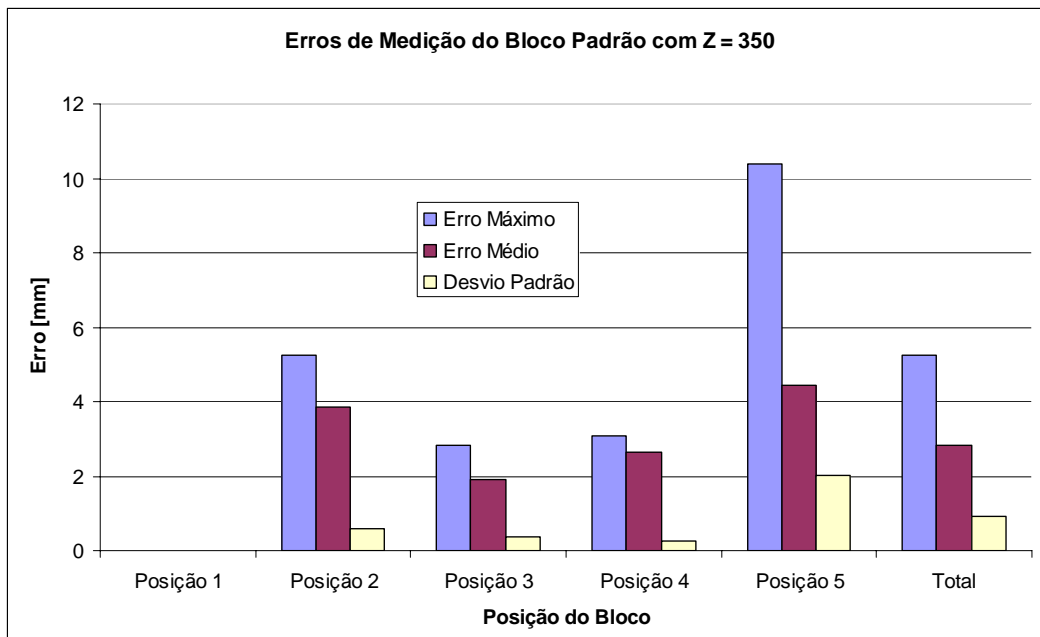


Figura 7.22. Erros de Medição do Bloco Padrão com Z=350 para a Versão de Laboratório.

Para a Posição 1 do bloco a uma distância de 350mm da câmera (Figura 7.22), devido às condições de iluminação do experimento não foi possível realizar nenhuma medição.

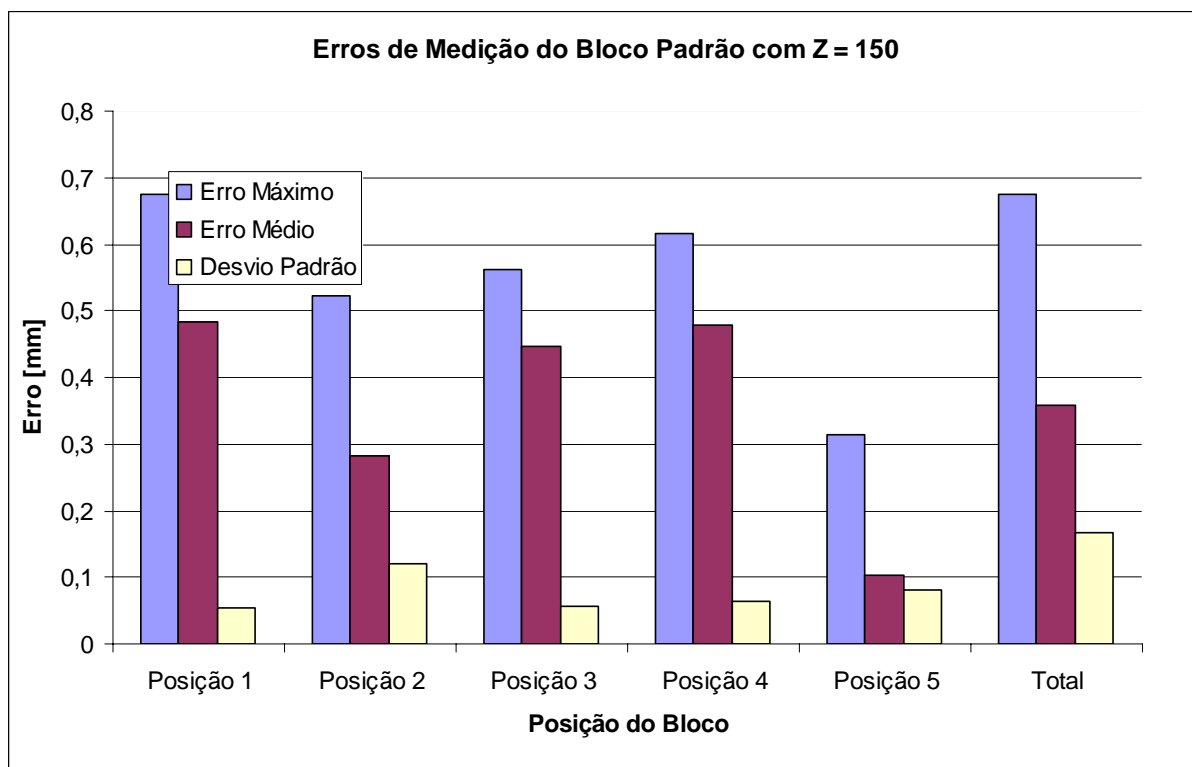


Figura 7.23. Erros de Medição do Bloco Padrão com Z=150 para a Versão de Campo.

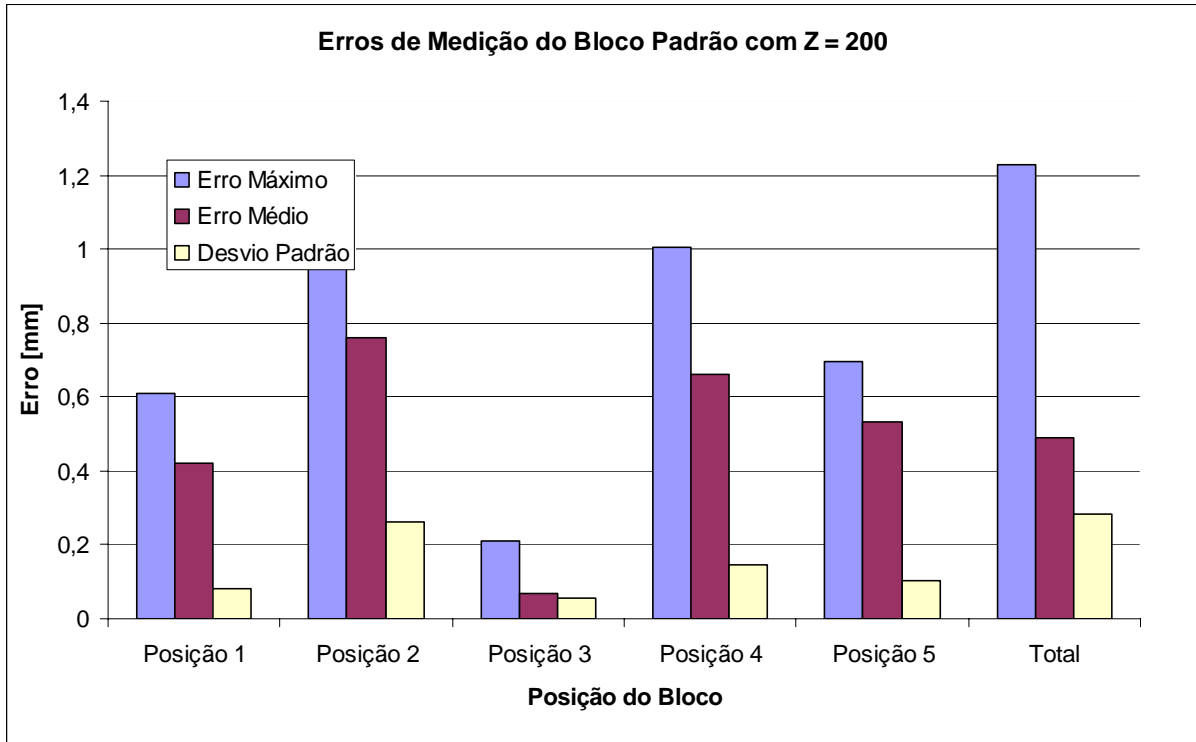


Figura 7.24. Erros de Medição do Bloco Padrão com Z=200 para a Versão de Campo.

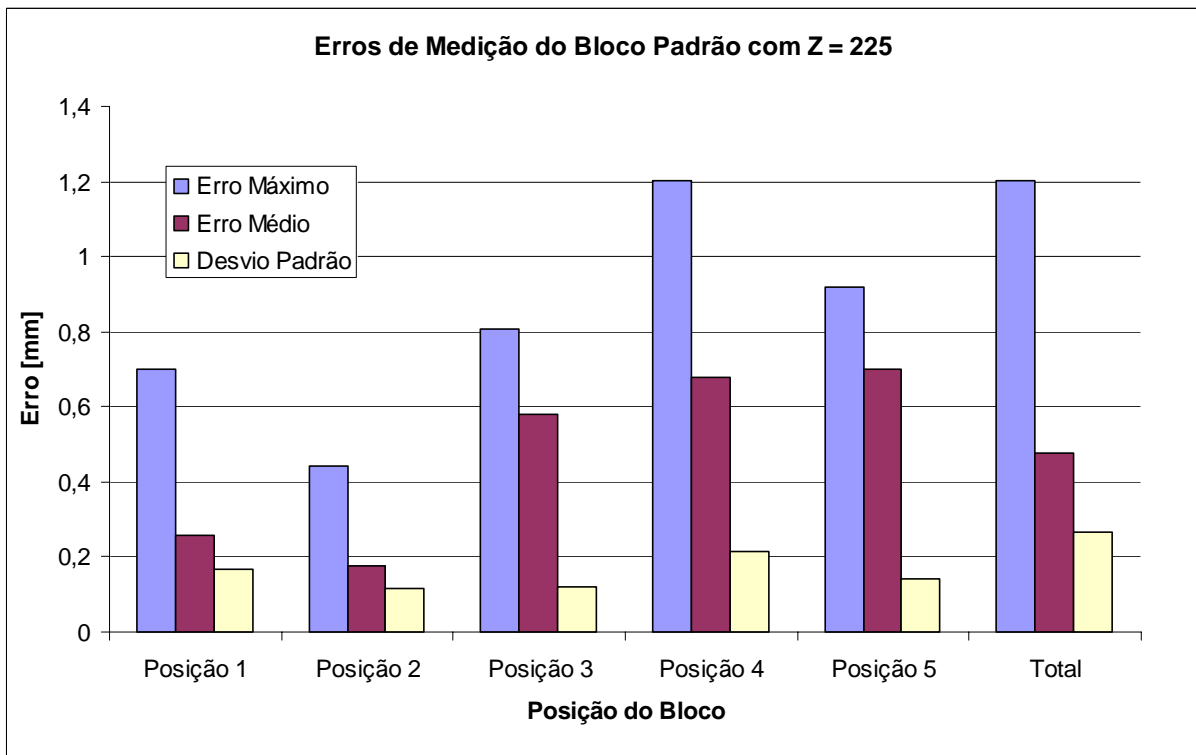


Figura 7.25. Erros de Medição do Bloco Padrão com Z=225 para a Versão de Campo.

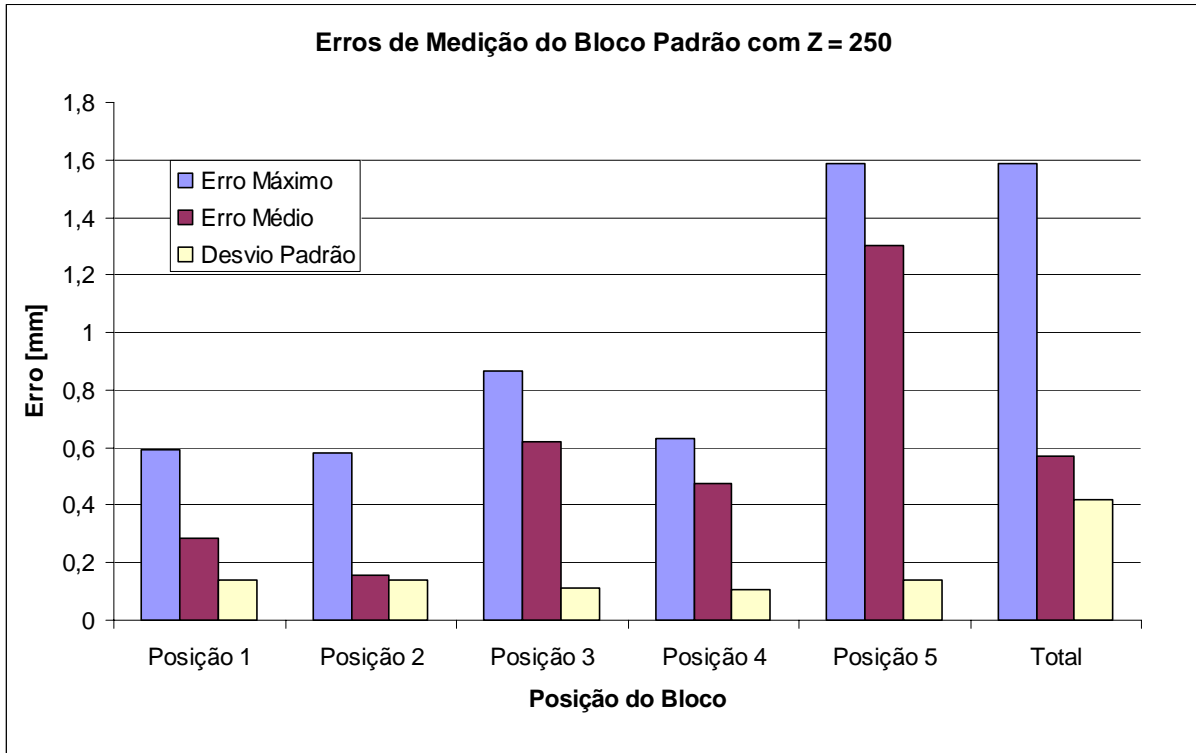


Figura 7.26. Erros de Medição do Bloco Padrão com Z=250 para a Versão de Campo.

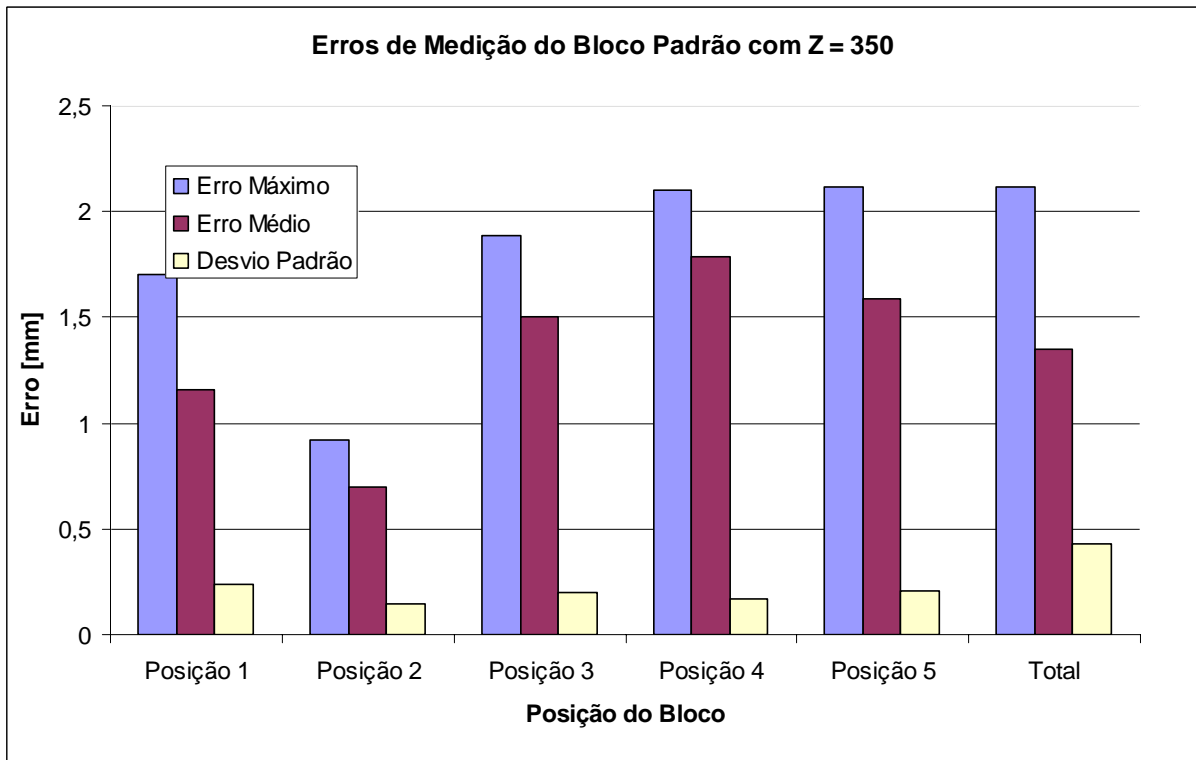


Figura 7.27. Erros de Medição do Bloco Padrão com Z=350 para a Versão de Campo.

Já na Figura 7.28 e na Figura 7.29, esses dados são ilustrados de forma mais simplificada de forma a possibilitar a observação da influência da distância sobre a incerteza de medição do sistema.

Barras de Erro de Medição

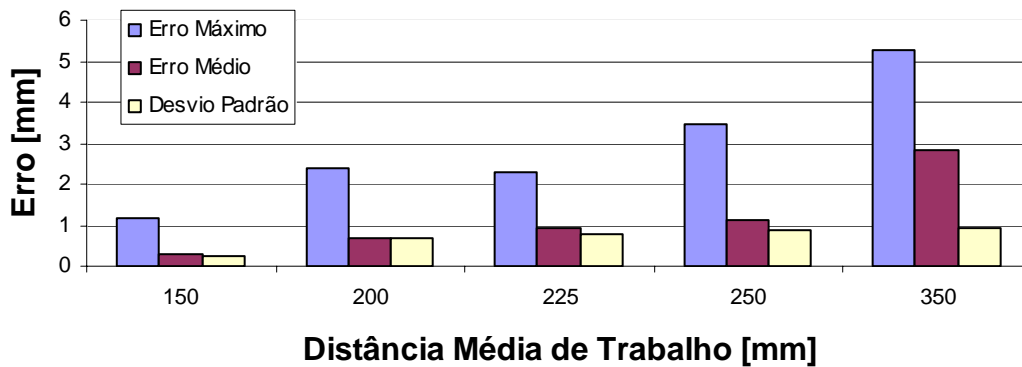


Figura 7.28. Erros Obtidos com a Distância de Bordas para a Versão de Laboratório.

Barras de Erro de Medição

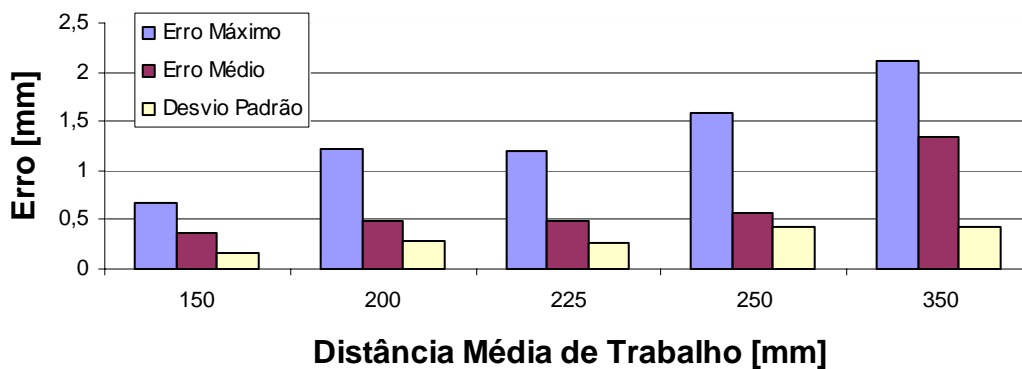


Figura 7.29. Erros Obtidos com a Distância de Bordas para a Versão de Campo.

A avaliação seguinte utiliza as técnicas de processamento apresentadas no Apêndice E para cálculo do plano médio que representa a nuvem de pontos digitalizada a partir do bloco metrológico e em seguida calcula como erro a distância entre cada um dos pontos digitalizados e esse plano conforme a Equação (7.25) e realiza o tratamento estatístico desses valores de erro conforme a Equação (7.27). É importante observar que o plano médio de cada digitalização é diferente e é recalculado, ou seja, para cada uma das 25 digitalizações realizadas é calculado um plano médio.

$$E_i = \frac{|Ax_i + By_i - z_i + C|}{\sqrt{A^2 + B^2 + 1}} \quad (7.25)$$

, onde o plano médio é dado por:

$$P_{\text{médio}} : Ax + By - Z + C = 0 \quad (7.26)$$

$$\left\{ \begin{array}{l} \bar{E} = \frac{\sum_{i=1}^m E_i}{m} \\ E_{\text{max}} = \max(E_i) \\ \sigma = \sqrt{\frac{\sum_{i=1}^m (E_i - \bar{E})^2}{m}} \end{array} \right. \quad (7.27)$$

Os valores obtidos nessa etapa de avaliação da precisão do sistema estão ilustrados na Figura 7.30 e na Figura 7.31.

Erro de Planicidade do Bloco Padrão

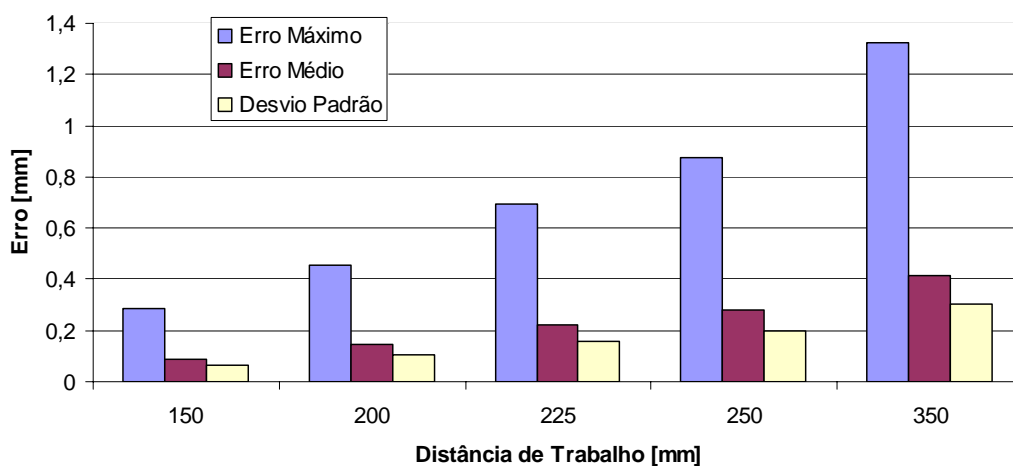


Figura 7.30. Erros Obtidos com a Planicidade do Bloco para a Versão de Laboratório.

Erro de Planicidade do Bloco Padrão

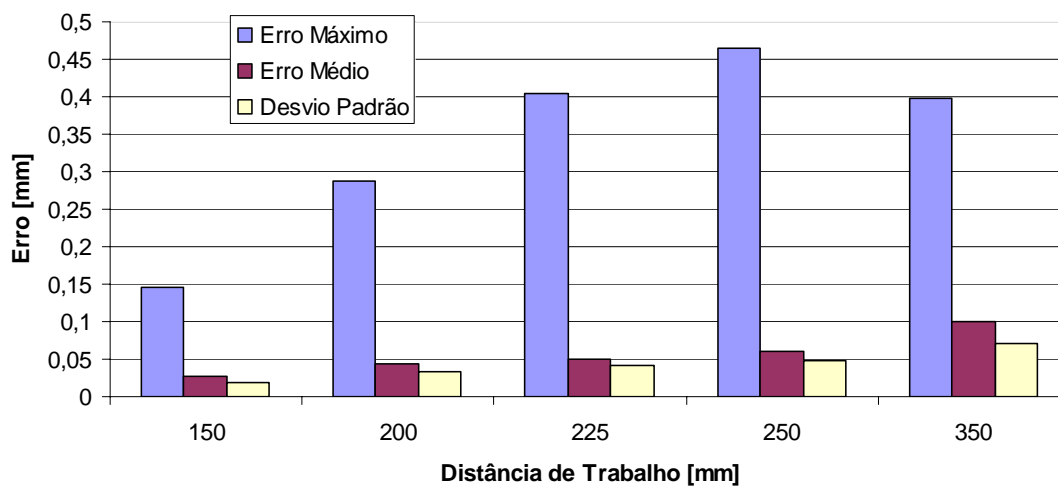


Figura 7.31. Erros Obtidos com a Planicidade do Bloco para a Versão de Campo.

Utilizando-se então um mesmo gráfico para a visualização de ambas as estratégias de cálculo do erro do sistema é possível realizar uma análise comparativa entre os resultados obtidos.

Erro Médio x Distância de Trabalho

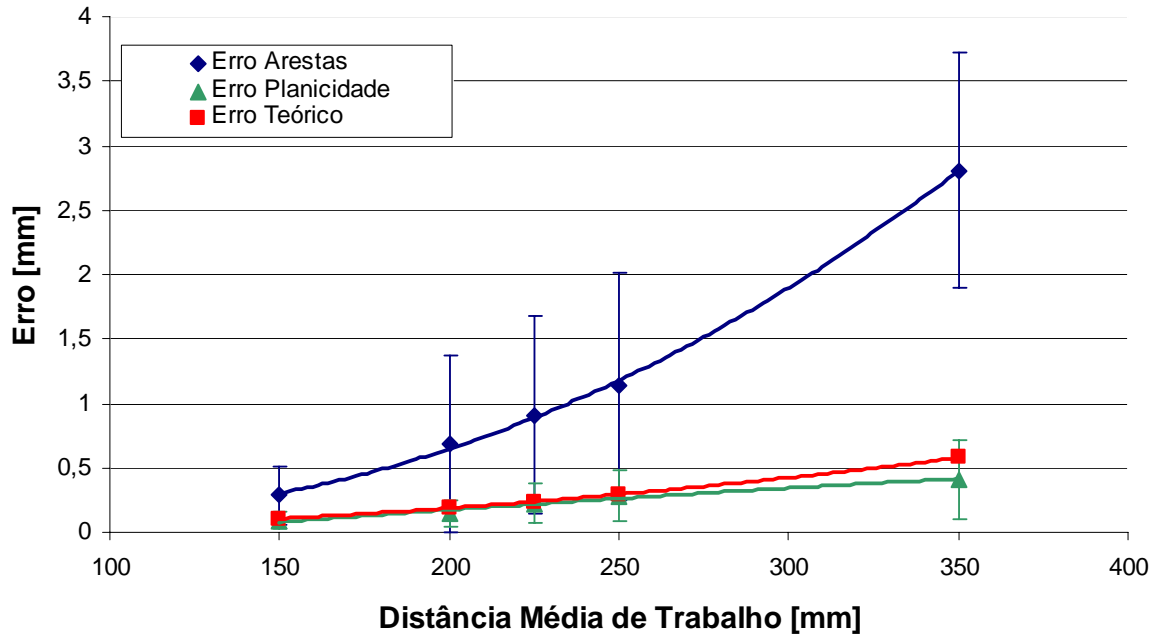


Figura 7.32. Precisão da Versão de Laboratório do Sistema VISSCAN-3D.

Erro Médio x Distância de Trabalho

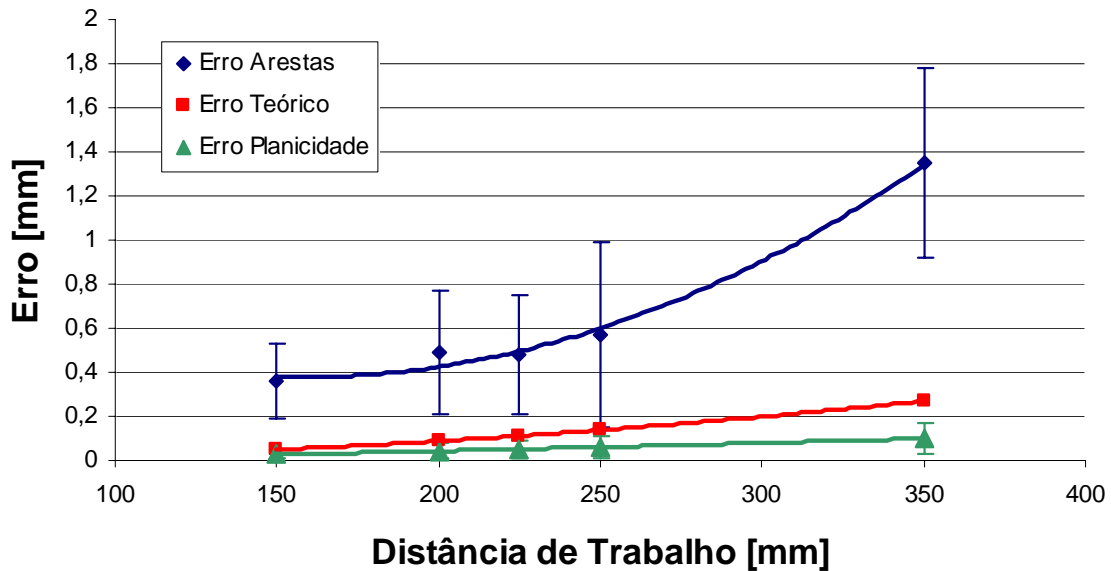


Figura 7.33. Precisão da Versão de Campo do Sistema VISSCAN-3D.

Observa-se que a precisão obtida através da medição da aresta do bloco metrológico é significativamente inferior à precisão obtida através do cálculo da planicidade do bloco. Um dos fatores que pode explicar esse resultado é o fato da medição da aresta ser mais suscetível aos efeitos de iluminação e reflexos. Esse efeito também pode ser observado na Figura 7.18 a Figura 7.27, onde é possível notar a variação do erro calculado em decorrência da variação da inclinação do bloco em relação à câmera. Outro ponto referente à iluminação que pode gerar erros é o fato das arestas padrões do bloco metrológico serem espelhadas (Figura 7.34). Esse espelhamento não somente gera reflexos que podem comprometer a precisão do sistema de digitalização, mas também altera consideravelmente a intensidade com que o feixe laser pode ser observado sobre o bloco.

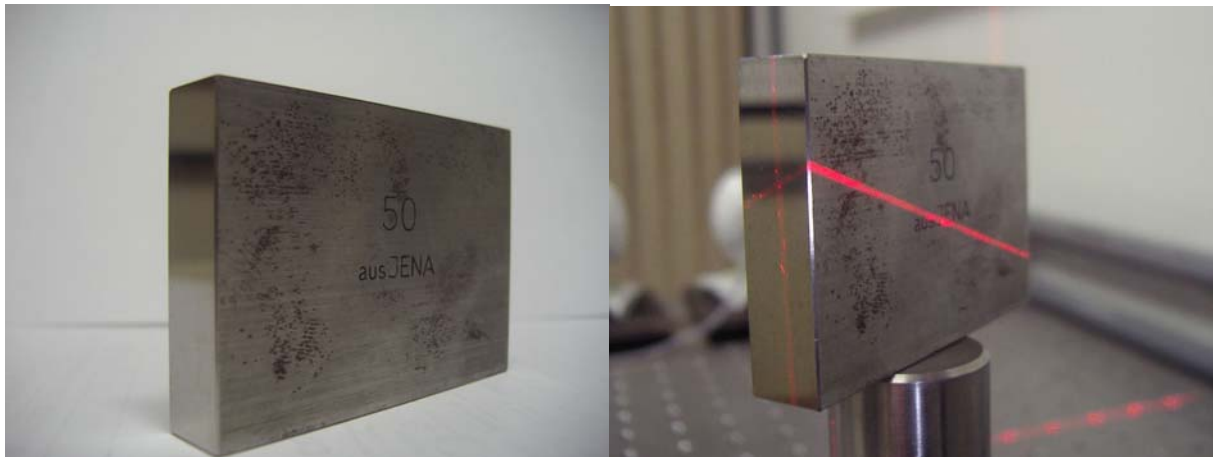


Figura 7.34. Arestas Espelhadas do Bloco Metrológico.

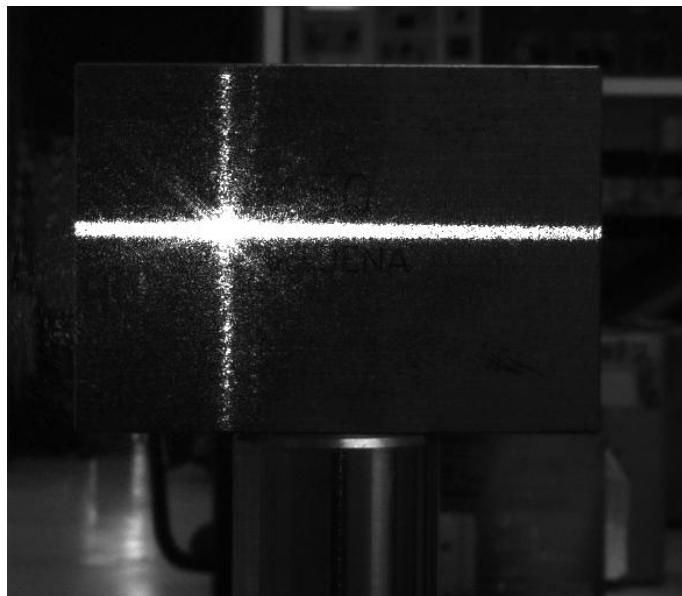


Figura 7.35. Efeito de Reflexão Observado nos Ensaio Experimentais.

Um outro fator que também influencia os resultados obtidos é o tamanho das amostras de dados colhidas. Quanto maior o tamanho de uma amostra, menos suscetível a pontos espúrios essa amostra será (Bussab e Morettin, 1987). Enquanto para o cálculo da distância entre as arestas têm-se cerca de

100 medições por digitalização, para o cálculo da planicidade do bloco têm-se cerca de 1500 medições.

Observando as curvas experimentais de precisão do sistema juntamente com seus desvios padrões e com a curva teórica na Figura 7.32 e na Figura 7.33, é possível observar que o erro obtido através da planicidade do bloco padrão está muito próximo do erro teórico calculado, enquanto que o erro calculado através da distância entre as arestas do bloco padrão cresce muito mais rapidamente a medida que a distância de trabalho do sistema aumenta.

Esses resultados ilustram a principal desvantagem de sistema de digitalização baseados em visão computacional, sua grande dependência à iluminação e à qualidade das imagens obtidas para processamento. As curvas de erro ilustram essa dependência de forma significativa, onde o erro calculado através da distância entre as arestas do bloco padrão, muito mais sensível aos efeitos de iluminação e reflexão é cerca de 700% superior ao valor esperado e ao calculado através da análise de planicidade do bloco. Existem técnicas de análise baseadas na polarização da luz laser (Clark et.al., 1997) que buscam corrigir esses problemas e aumentar a robustez e a precisão de sistemas de digitalização baseados em triangulação ativa, entretanto a solução mais utilizada continua sendo o uso de pós para polvilhar a superfície a ser digitalizada (Sansoni e Docchio, 2004).

7.3 SENSOREAMENTO DA ANGULAÇÃO DO DIODO LASER

Uma das principais características do sistema VISSCAN-3D é a substituição do sensor de medição angular do diodo laser móvel pelo uso de técnicas de visão computacional e triangulação para realizar essa medição da rotação do diodo.

O sistema substitui o uso de um sensor de posicionamento angular de alto custo pelo uso de um segundo diodo laser que, apesar de elevar o custo computacional do processo de digitalização, reduz o custo financeiro total do sistema e melhora a incerteza de medição dessa angulação.

De forma a avaliar o desempenho e a precisão dessas técnicas é apresentado aqui um estudo teórico das equações utilizadas para o cálculo dessa angulação a partir do ponto de cruzamento das duas linhas lasers (Equação (3.11)) e uma série de experimentos, onde os resultados obtidos através dessas técnicas foram comparados com os resultados fornecidos por uma etapa rotatória de precisão.

7.3.1 Avaliação Teórica

As equações que descrevem o cálculo da angulação do diodo laser foram apresentadas detalhadamente na Seção 3.1 (Equação (3.11)) e, observando-se detalhadamente essas equações, nota-se que a única variável presente é o ponto de cruzamento das linhas lasers (x_{int}, y_{int}). Todos os outros termos das equações são parâmetros construtivos fixos do sistema. Logo, a precisão com que é possível obter-se a

angulação do diodo laser depende da precisão com que esse ponto de cruzamento é determinado e da precisão com que cada um dos parâmetros do sistema é determinado através da calibração.

Entretanto, deseja-se determinar a incerteza com que o ângulo de inclinação do laser pode ser determinado e não a incerteza com que a cotangente desse ângulo pode ser determinada, de forma que, a partir da Equação (3.11):

$$\theta_y = \operatorname{arccot} \left(\frac{\frac{b_y}{b_x} (f \cot(\theta_x) + x_{int}) - y_{int}}{f - \frac{d_y}{fb_x} (f \cot(\theta_x) + x_{int})} \right) \quad (7.28)$$

Derivando essa equação em relação a cada um dos parâmetros e variáveis do sistema é possível obter a equação de propagação de erros apresentada na Equação (7.29).

$$\begin{aligned} \Delta\theta_y^2 = & \left(\frac{\partial\theta_y}{\partial x_{int}} \right)^2 \Delta x_{int}^2 + \left(\frac{\partial\theta_y}{\partial y_{int}} \right)^2 \Delta y_{int}^2 + \left(\frac{\partial\theta_y}{\partial f} \right)^2 \Delta f^2 + \left(\frac{\partial\theta_y}{\partial b_y} \right)^2 \Delta b_y^2 + \dots \\ & \dots + \left(\frac{\partial\theta_y}{\partial d_y} \right)^2 \Delta d_y^2 + \left(\frac{\partial\theta_y}{\partial b_x} \right)^2 \Delta b_x^2 + \left(\frac{\partial\theta_y}{\partial \cot(\theta_x)} \right)^2 \Delta \cot(\theta_x)^2 \end{aligned} \quad (7.29)$$

Considerando a situação ideal, onde os parâmetros calibrados não apresentam erros, é possível simplificar a Equação (7.29) como:

$$\Delta\theta_y^2 = \left(\frac{\partial\theta_y}{\partial x_{int}} \right)^2 \Delta x_{int}^2 + \left(\frac{\partial\theta_y}{\partial y_{int}} \right)^2 \Delta y_{int}^2, \quad (7.30)$$

onde:

$$\begin{aligned} \frac{\partial\theta_y}{\partial x_{int}} = & \frac{b_y}{fb_x \left(1 - \frac{d_y (fctg_x - x_{int})}{fb_x} \right)} + \frac{d_y \left(\frac{b_y (fctg_x - x_{int})}{b_x} + y_{int} \right)}{f^2 b_x \left(1 - \frac{d_y (fctg_x - x_{int})}{fb_x} \right)^2} \left(1 + \frac{\left(\frac{b_y (fctg_x - x_{int})}{b_x} + y_{int} \right)^2}{f^2 \left(1 - \frac{d_y (fctg_x - x_{int})}{fb_x} \right)^2} \right)^{-1} \\ \frac{\partial\theta_y}{\partial y_{int}} = & \left(f \left(1 - \frac{d_y (fctg_x - x_{int})}{fb_x} \right) \left(1 + \frac{\left(\frac{b_y (fctg_x - x_{int})}{b_x} + y_{int} \right)^2}{f^2 \left(1 - \frac{d_y (fctg_x - x_{int})}{fb_x} \right)^2} \right) \right)^{-1} \end{aligned} \quad (7.31)$$

Logo, simplificando a Equação (7.30) de acordo com a Equação (3.9) e a Equação (3.10). tem-se:

$$\Delta\theta_y^2 = \left(\frac{b_y + d_y \cot(\theta_y)}{fb_x} \frac{z}{z - d_y} \frac{1}{1 + \cot(\theta_y)^2} \right)^2 \Delta x_{\text{int}}^2 + \left(\frac{1}{f} \frac{z}{z - d_y} \frac{1}{1 + \cot(\theta_y)^2} \right)^2 \Delta y_{\text{int}}^2 \quad (7.32)$$

Considerando então que o valor de d_y é desprezível em relação à distância z entre o ponto de interseção e a câmera, realiza-se a seguinte simplificação:

$$d_y \ll z \rightarrow \Delta\theta_y^2 \approx \left(\frac{b_y + d_y \cot(\theta_y)}{fb_x} \frac{1}{1 + \cot(\theta_y)^2} \right)^2 \Delta x_{\text{int}}^2 + \left(\frac{1}{f} \frac{1}{1 + \cot(\theta_y)^2} \right)^2 \Delta y_{\text{int}}^2 \quad (7.33)$$

Através da Equação (7.33) observa-se que a precisão com que a inclinação do diodo laser móvel pode ser determinada não varia diretamente com a profundidade z do ponto de interseção dos lasers, ou seja, diferentemente do que poderia ser presumido, o cálculo da angulação do laser móvel, apesar de depender da triangulação, não tem sua precisão afetada diretamente pela distância de trabalho do sistema. Essa característica desse modo de sensoriamento é uma qualidade importante para torná-lo competitivo frente aos sensores de posição angular que tem entre suas principais vantagens a independência em relação ao sistema de visão.

Avaliando então a Equação (7.33), na situação mais crítica, conforme realizado na Seção 7.1.2, tem-se:

$$\Delta\theta_y = \sqrt{\left(\frac{|b_y + d_y|}{fb_x} \right)^2 \Delta x_{\text{int}}^2 + \left(\frac{1}{f} \right)^2 \Delta y_{\text{int}}^2} \quad (7.34)$$

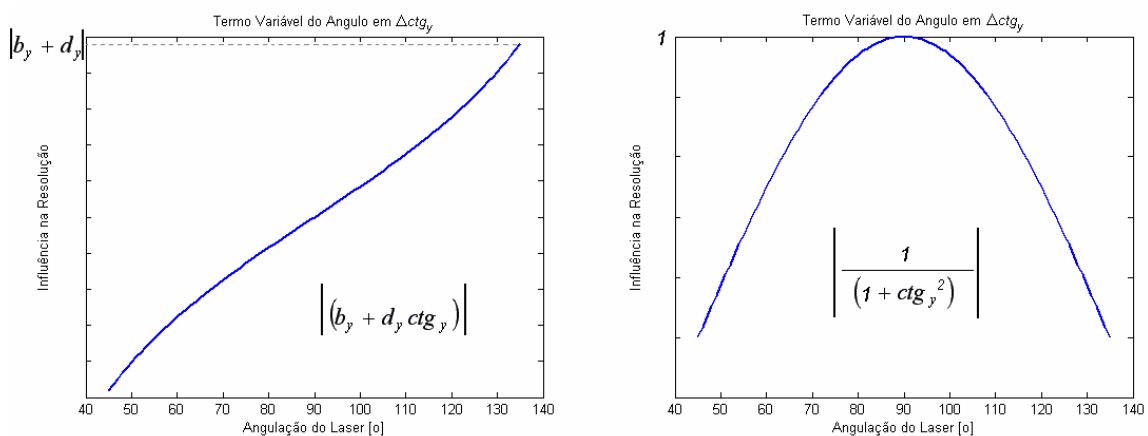


Figura 7.36. Precisão Teórica dos Sistemas de Digitalização Implementados.

Utilizando os valores de parâmetros da versão de laboratório e de campo do sistema VISSCAN-3D (Tabela 3.1 e Tabela 3.2), obtém-se a incerteza de medição com que a angulação do laser é calculada.

Tabela 7.4. Erro Teórico do Sensoreamento da Angulação do Laser.

	$\Delta\theta_y [^\circ]$
<i>Versão de Laboratório</i>	0.0739
<i>Versão de Campo</i>	0.0807

A redução da precisão teórica do sensoreamento da angulação laser observada na versão de campo do sistema VISSCAN-3D ocorre devido a redução da distância base b_x entre o laser fixo e a câmera.

7.3.2 Avaliação Experimental

Para uma avaliação mais precisa das técnicas de cálculo da angulação do diodo laser, foram realizadas uma série de testes experimentais. Utilizando-se um estágio rotatório de precisão fornecido pela Edmund Optics que possui uma resolução de 0.02° e uma incerteza de 0.23° que pode ser observado na Figura 7.37, o diodo laser foi posicionado em diferentes angulações e monitorado por ambos os sistemas.

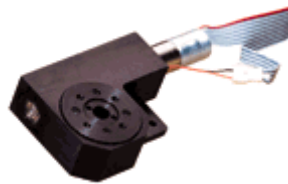


Figura 7.37. Estágio Rotatório para Avaliação do Cálculo da Angulação do Laser.

Para a determinação da angulação do laser utilizando o sistema VISSCAN-3D é necessário que o plano de luz laser gerado pelo diodo seja projetado sobre uma superfície. Logo, para a realização desses experimentos uma superfície plana foi posicionada em três diferentes distâncias conhecidas em relação ao sistema de digitalização conforme ilustrado na Figura 7.38 e em seguida o diodo laser foi rotacionado com passos de 0.2° , varrendo o plano dez vezes para cada uma das distâncias de trabalho.

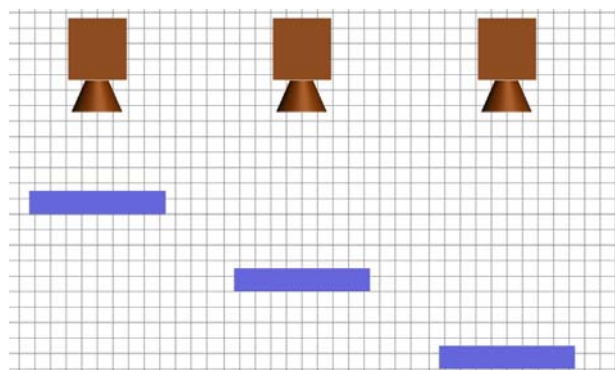


Figura 7.38. Posições do Plano para Cálculo da Angulação.

O sistema VISSCAN-3D, entretanto, monitora a cotangente da angulação do laser ($\cot(\theta_y)$), enquanto a etapa rotatória monitora diretamente a angulação (θ_{Laser}) através do uso de *encoders*, logo, para realizar uma comparação efetiva, as diferenças observadas entre as duas medições são comparadas conforme a Equação (7.35) (Bussab e Morettin, 1987).

$$E_i = \left| \cot^{-1} \left({}^i \cot(\theta_y) \right) - \theta_{Laser} \right| \Rightarrow \begin{cases} \bar{E} = \frac{\sum_{i=1}^m E_i}{m} \\ E_{\max} = \max(E_i) \\ \sigma = \sqrt{\frac{\sum_{i=1}^m (E_i - \bar{E})^2}{m}} \end{cases} \quad (7.35)$$

, onde m é o número de medições realizadas.

Essa avaliação é realizada separadamente para cada diferente distância entre o plano no qual o laser é projetado e o sistema de digitalização e os resultados obtidos podem ser observados na Figura 7.39.

Reta de Calibração do Sistema

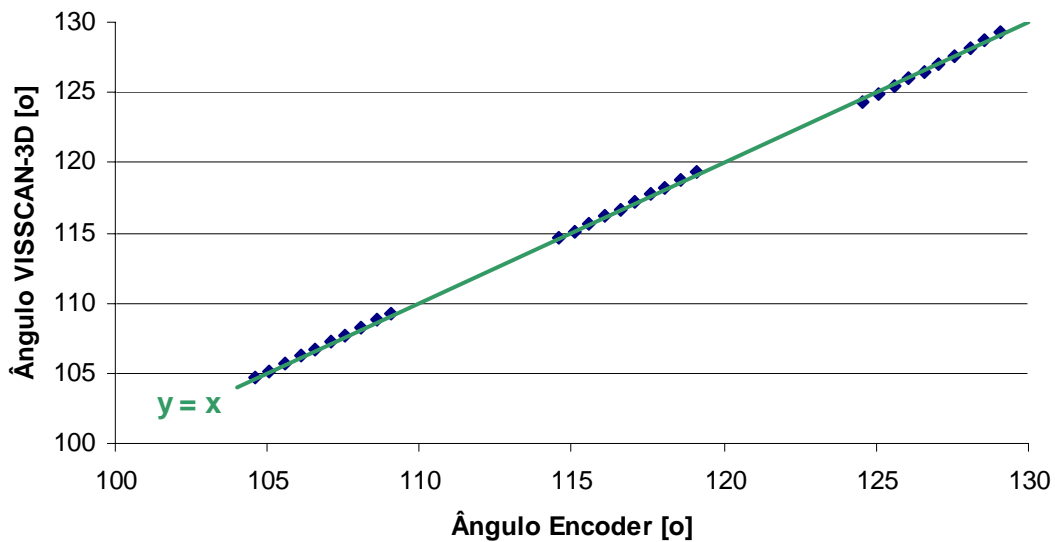
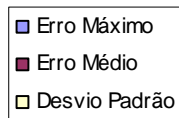


Figura 7.39. Diferenças entre as Medições de Angulação do Laser.

Considerando-se as medições fornecidas pela etapa rotatória como o valor padrão para comparação, traçam-se as curvas de precisão do sistema de monitoramento da angulação do laser do sistema VISSCAN-3D em função da distância de trabalho.



Barras de Diferença de Ângulo

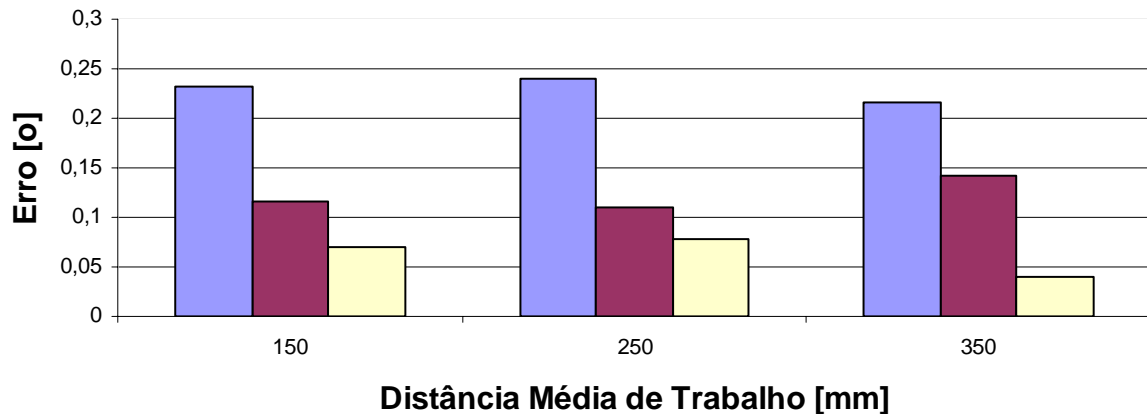


Figura 7.40. Barras de Precisão Experimental do Cálculo da Angulação do Laser.

Diferença de Ângulos x Distância de Trabalho

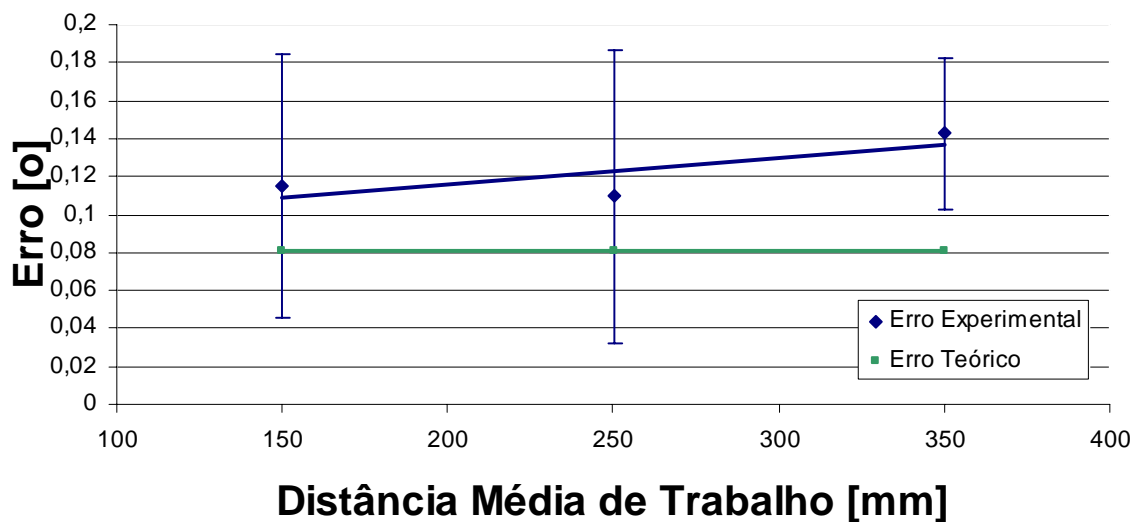


Figura 7.41. Precisão Experimental e Teórica do Cálculo da Angulação do Laser.

Os dados de precisão obtidos através da comparação entre a etapa rotatória de precisão e o sistema de visão apesar de apresentarem uma análise qualitativa do sistema de sensoriamento, não constituem uma análise quantitativa válida, pois a precisão esperada para o sistema de visão (Tabela 7.4) é superior à precisão informada pelo fabricante da etapa rotatória.

Para se realizar uma análise quantitativa válida do sistema de sensoriamento seria necessário que o instrumento de medição utilizado para comparação tivesse uma precisão cerca de 10 vezes melhor do que a esperada para o sistema de visão (Alvez, 1996).

Observando-se a avaliação de precisão de uma forma diferente, é possível considerar os procedimentos experimentais como procedimentos de aferição do *encoder* e da precisão indicada pelo fabricante. A partir dos erros de inclinação indicados na Figura 7.40 é possível observar que o máximo erro obtido está faixa de 0.23° , o que é exatamente a faixa de erro indicada pelo fabricante da etapa de rotação.

8 RESULTADOS OBTIDOS

Nesse capítulo são apresentadas várias digitalizações realizadas através do sistema VISSCAN-3D. São apresentadas digitalizações de diferentes tipos de superfícies com as mais variadas geometrias, além de algumas digitalizações mais complexas feitas em várias etapas com a posterior combinação desses dados em um único sólido.

Inicialmente são apresentadas algumas geometrias básicas e simples e gradativamente são apresentadas digitalizações mais complexas até a conclusão do capítulo, apresentando uma digitalização completa de um objeto realizada através de várias vistas e sua combinação.

A primeira digitalização apresentada aqui é a de uma superfície plana que foi posicionada em cinco diferentes orientações conhecidas e suas orientações calculadas através dos dados do sistema VISSCAN-3D.

Para o posicionamento inicial do bloco com uma inclinação de 0° , o bloco é posicionado manualmente em uma posição aproximadamente perpendicular à câmera e é digitalizado. O valor de inclinação calculado é então considerado como a inclinação zero e todas as outras medições de inclinação são tomadas em referência a essa orientação inicial. Os resultados obtidos podem ser observados na Figura 8.3, na Figura 8.4 e na Tabela 8.1.

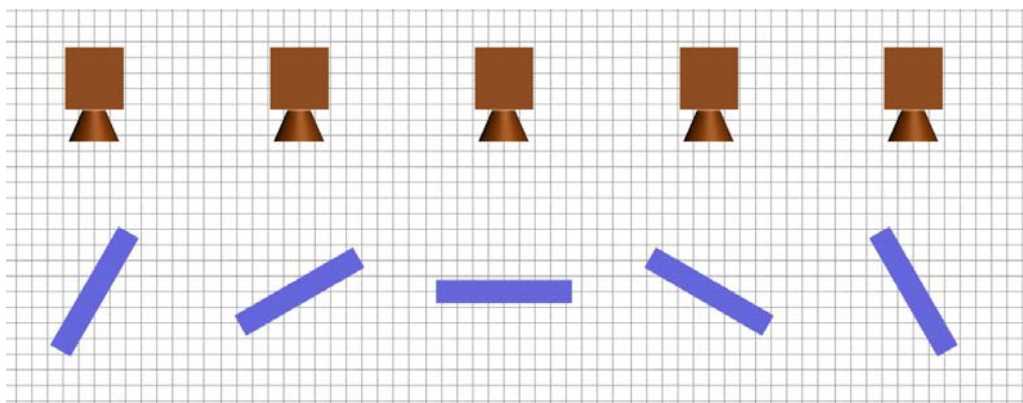


Figura 8.1. Esquemático da Digitalização.

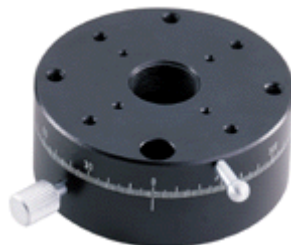


Figura 8.2. Estágio Rotacional Graduado Manual.

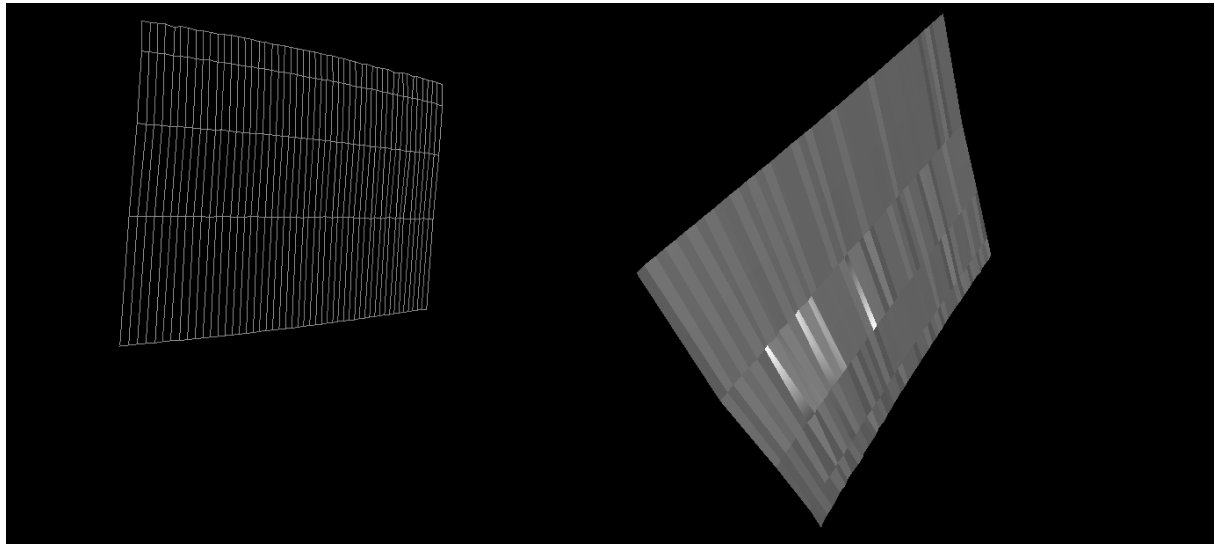


Figura 8.3. Digitalização de uma Superfície Plana.

Tabela 8.1. Orientações Calculadas.

<i>Orientação Etapa Rotacional</i>	<i>Orientação VISSCAN-3D</i>
-20°	$-20,52^{\circ}$
-10°	$-10,37^{\circ}$
0°	0°
10°	$10,34^{\circ}$
20°	$20,67^{\circ}$

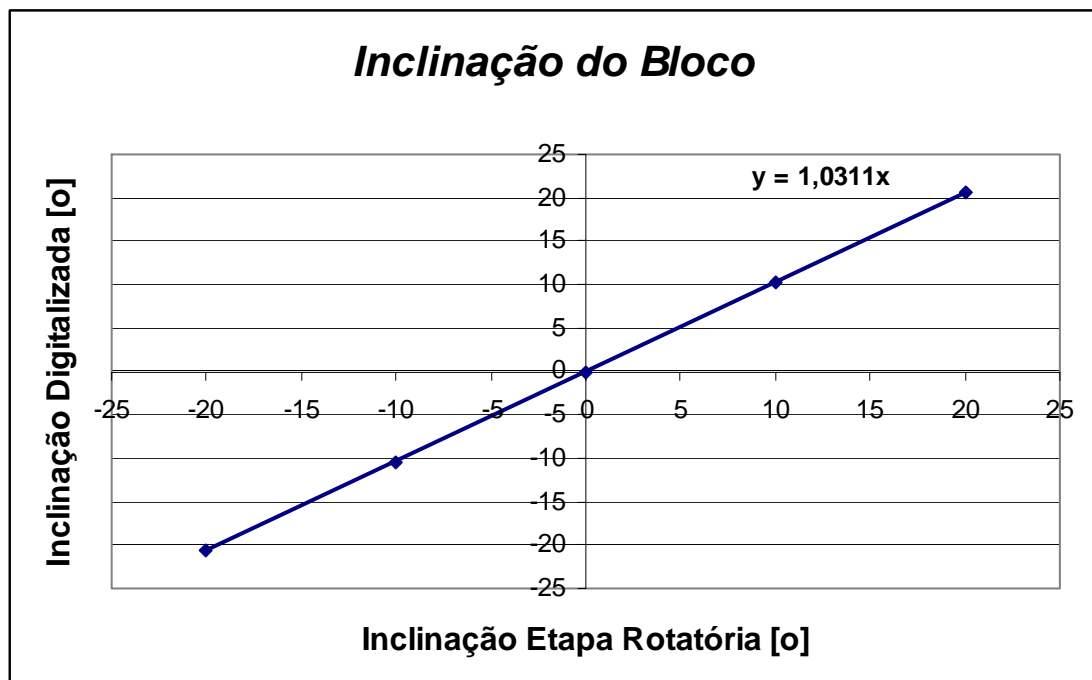


Figura 8.4. Orientações Calculadas.

A diferença entre o valor de inclinação esperado e o calculado deve-se não somente ao erro de medição do sistema, mas também ao erro de posicionamento do bloco, uma vez que o bloco é rotacionado manualmente com o auxílio de uma etapa rotacional graduada com 1° de resolução.

A segunda digitalização apresentada aqui consiste na digitalização de uma esfera. Como o sistema VISSCAN-3D é um scanner de superfície, ele não realiza a digitalização completa da esfera, sendo obtida somente uma calota esférica, conforme observado na Figura 8.5.

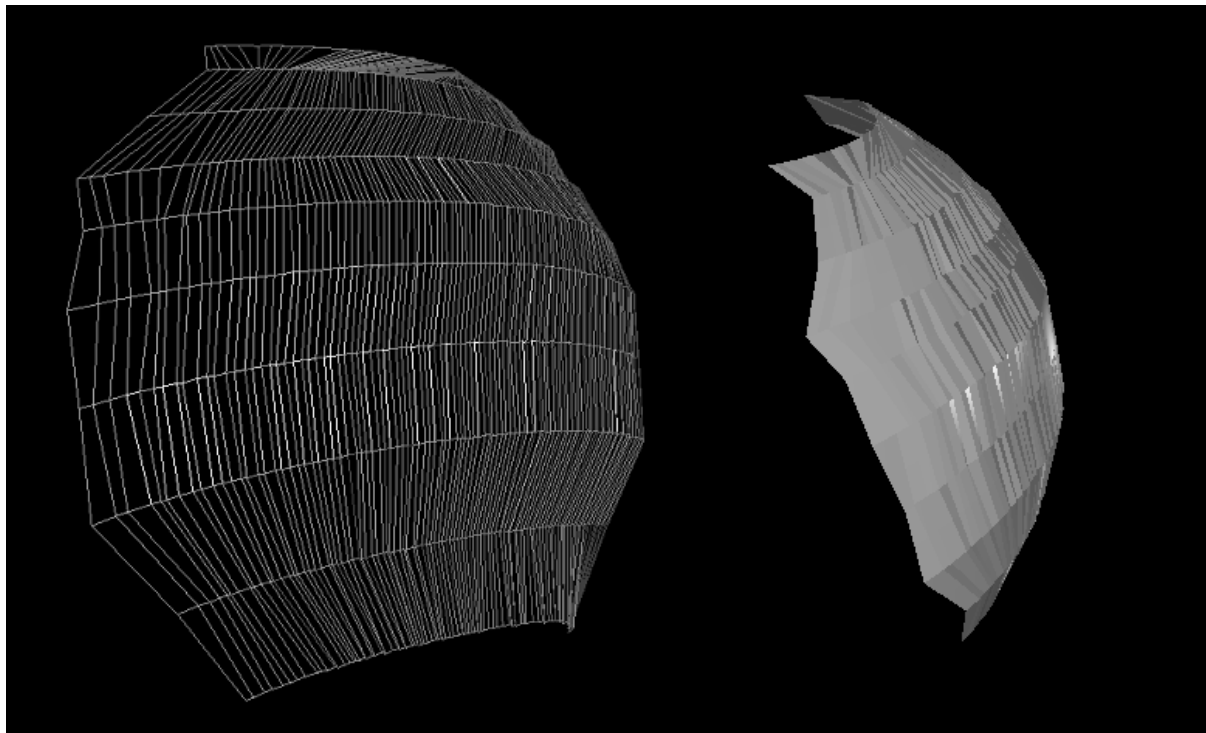


Figura 8.5. Digitalização de uma Esfera.

Na nuvem de pontos obtida na Figura 8.5 tem-se um grande espaçamento entre as linhas de digitalização. Quanto menor esse espaçamento, maior o número de linhas que podem ser digitalizadas em uma superfície e maior o número de detalhes dessa superfície que poderão ser observados. Esse espaçamento é limitado pelo sistema de movimentação do laser e sua resolução.

A terceira digitalização é a digitalização de uma peça mecânica plana, com uma série de furos passantes e reentrâncias em seu contorno.

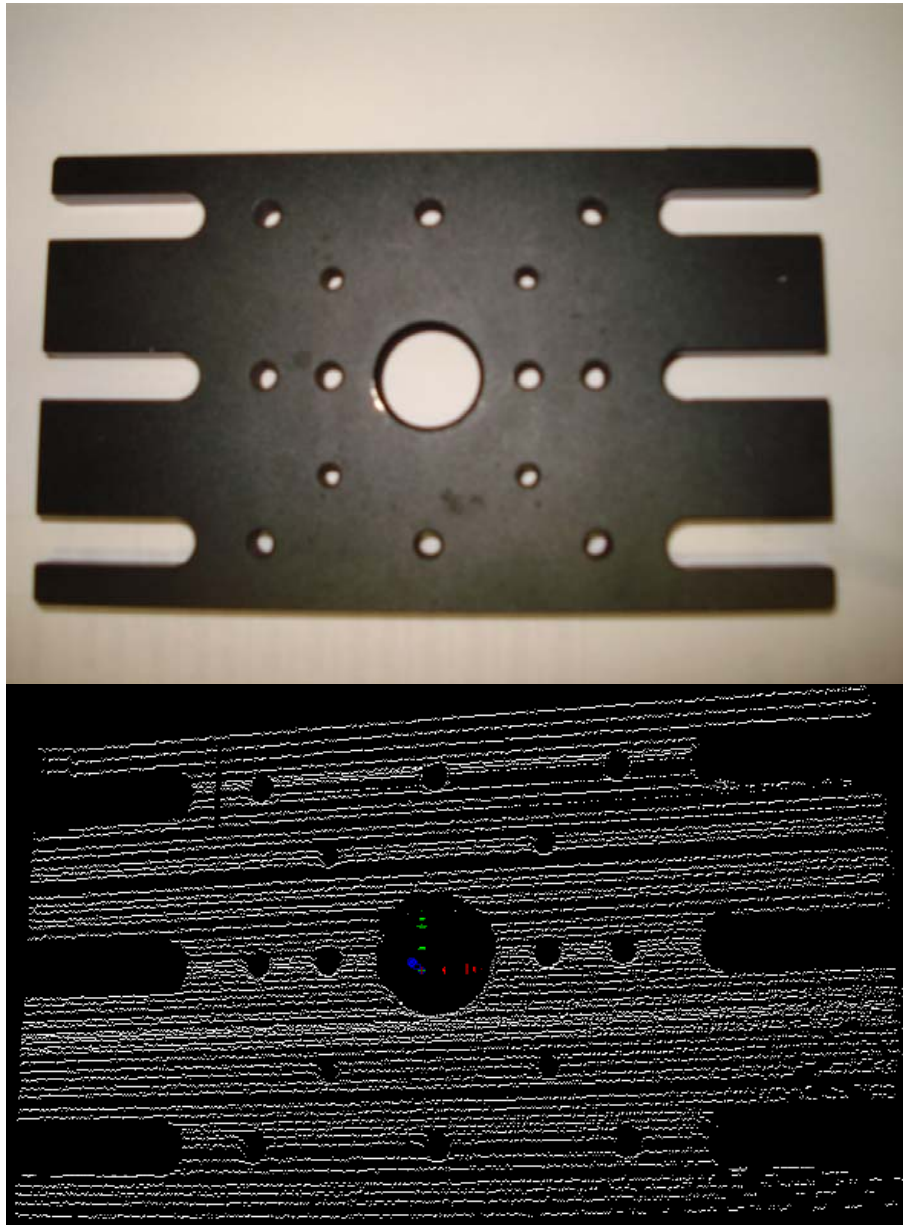


Figura 8.6. Peça Mecânica e sua Digitalização.

Na nuvem de pontos obtida na Figura 8.6 observa-se algumas falhas na digitalização da superfície, principalmente na parte inferior da peça. Uma das razões da ocorrência desse tipo de falhas é a aplicação de um limiar muito elevado para a extração da linha laser da imagem, o que faz com que parte da linha seja perdida durante essa operação.

A escolha da estratégia de processamento de imagens para a extração da linha laser é uma operação crítica e, dependendo da natureza da superfície que se deseja digitalizar, é necessário adequar essa estratégia às características dessa superfície.

A quarta digitalização apresentada é a digitalização de um cordão de solda. Essa superfície, sua digitalização e seu processamento utilizando as funções de detecção por gradiente (Apêndice E) são apresentadas na Figura 8.7 e na Figura 8.8.

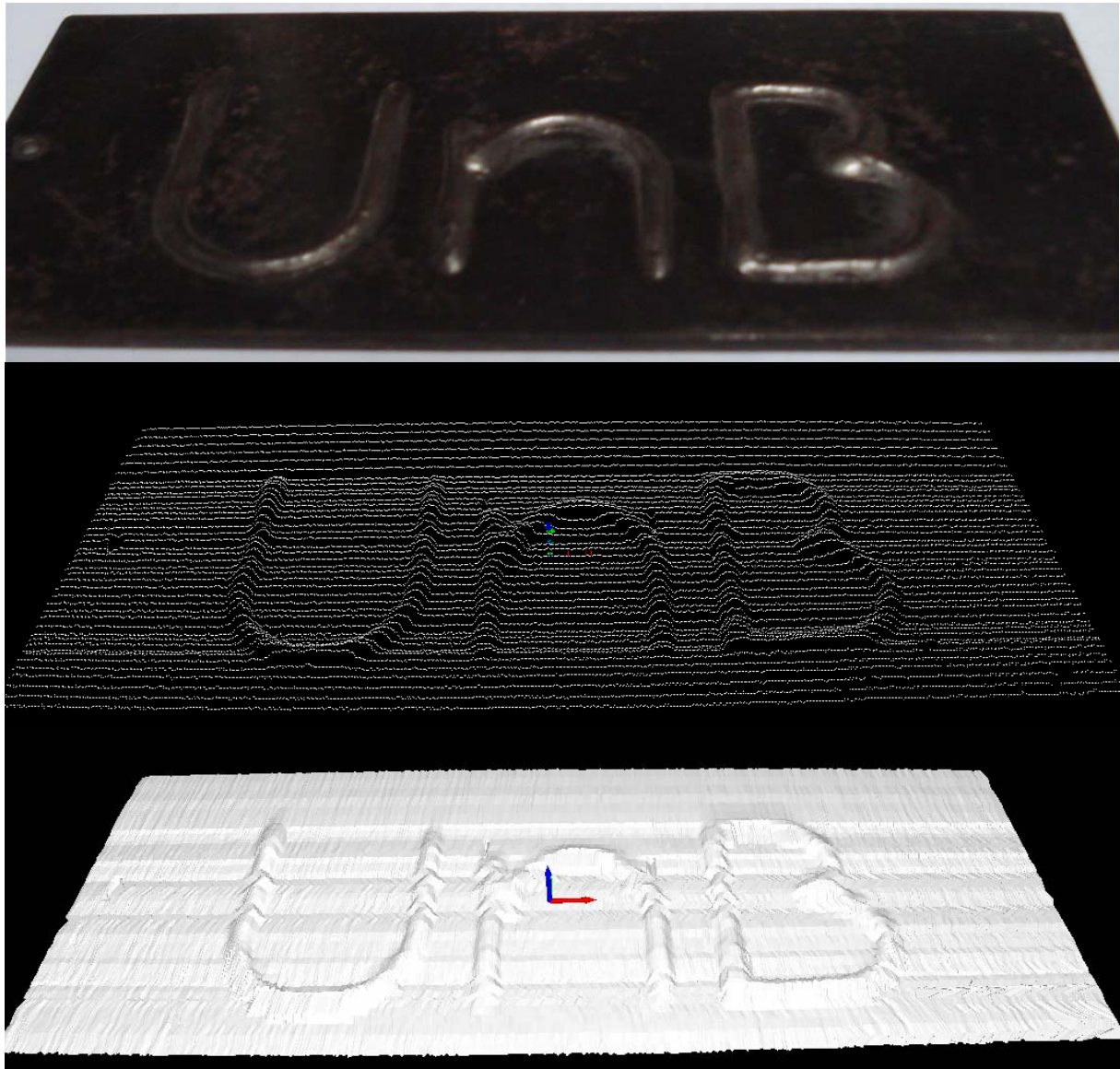


Figura 8.7. Cordão de Solda e sua Digitalização.

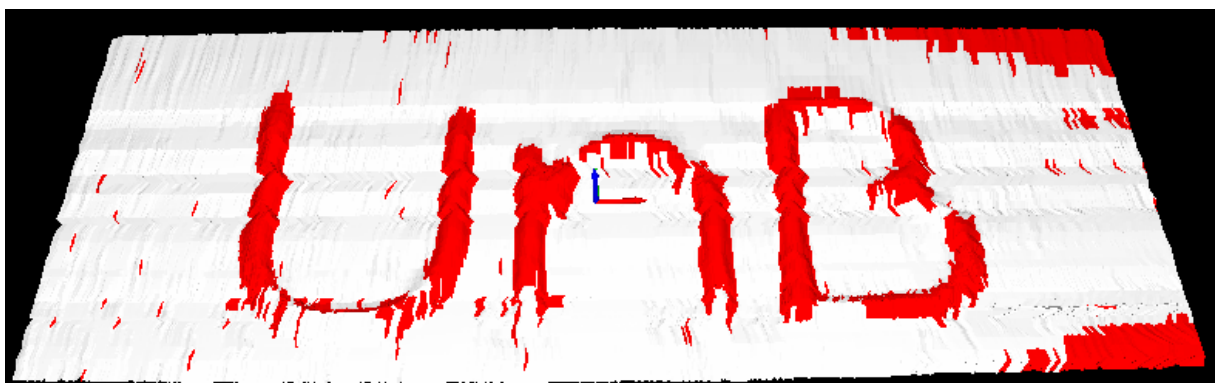


Figura 8.8. Processamento do Cordão de Solda.

Através da análise da nuvem de pontos utilizando a função de detecção por gradiente, é possível destacar o cordão de solda sobre a placa e observar que a placa encontra-se empenada.

A quinta e a sexta digitalizações apresentadas são a digitalização de uma calculadora e de um aparelho celular, conforme pode ser observado na Figura 8.9 e na Figura 8.10 respectivamente.

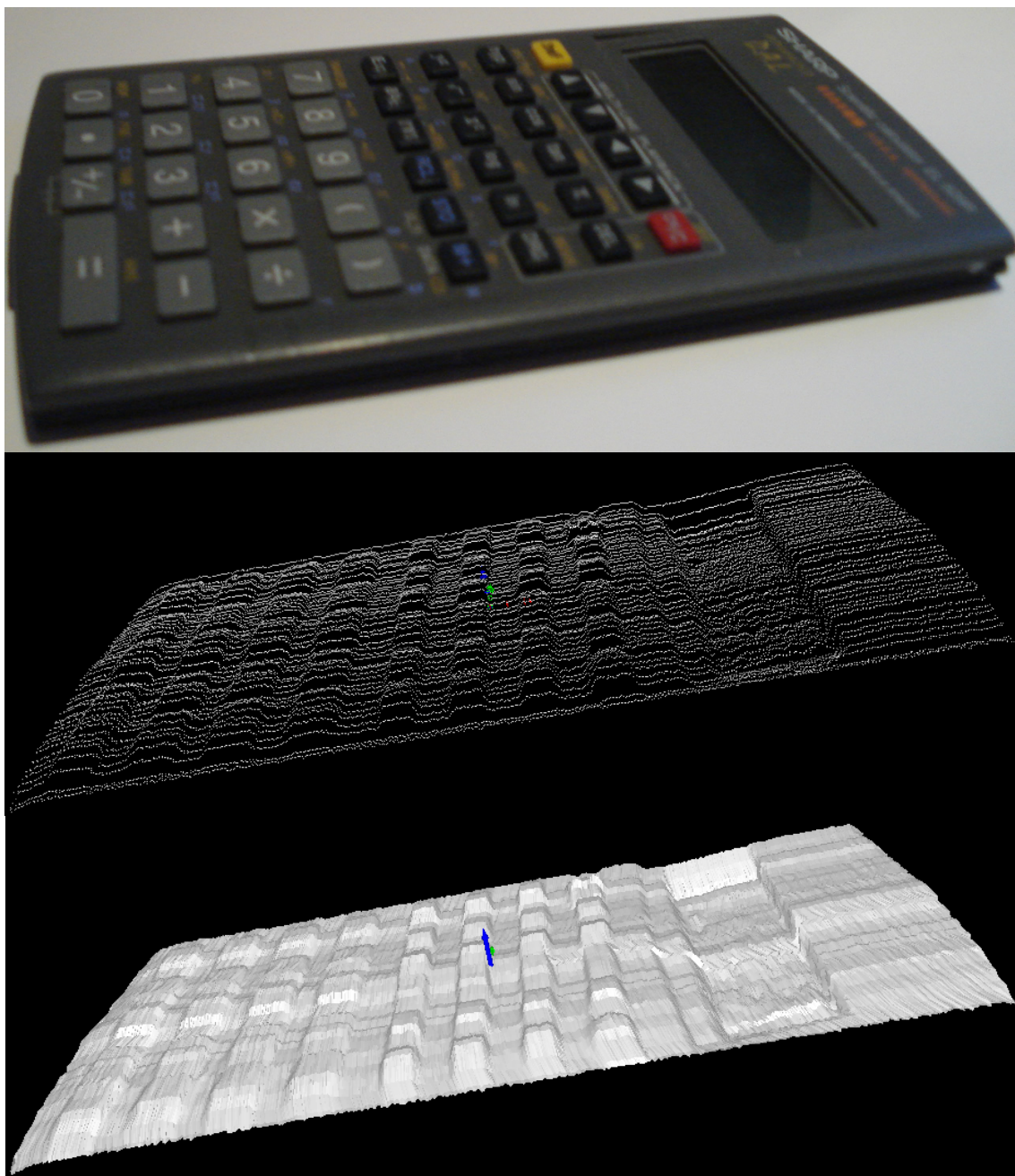


Figura 8.9. Digitalização de uma Calculadora.

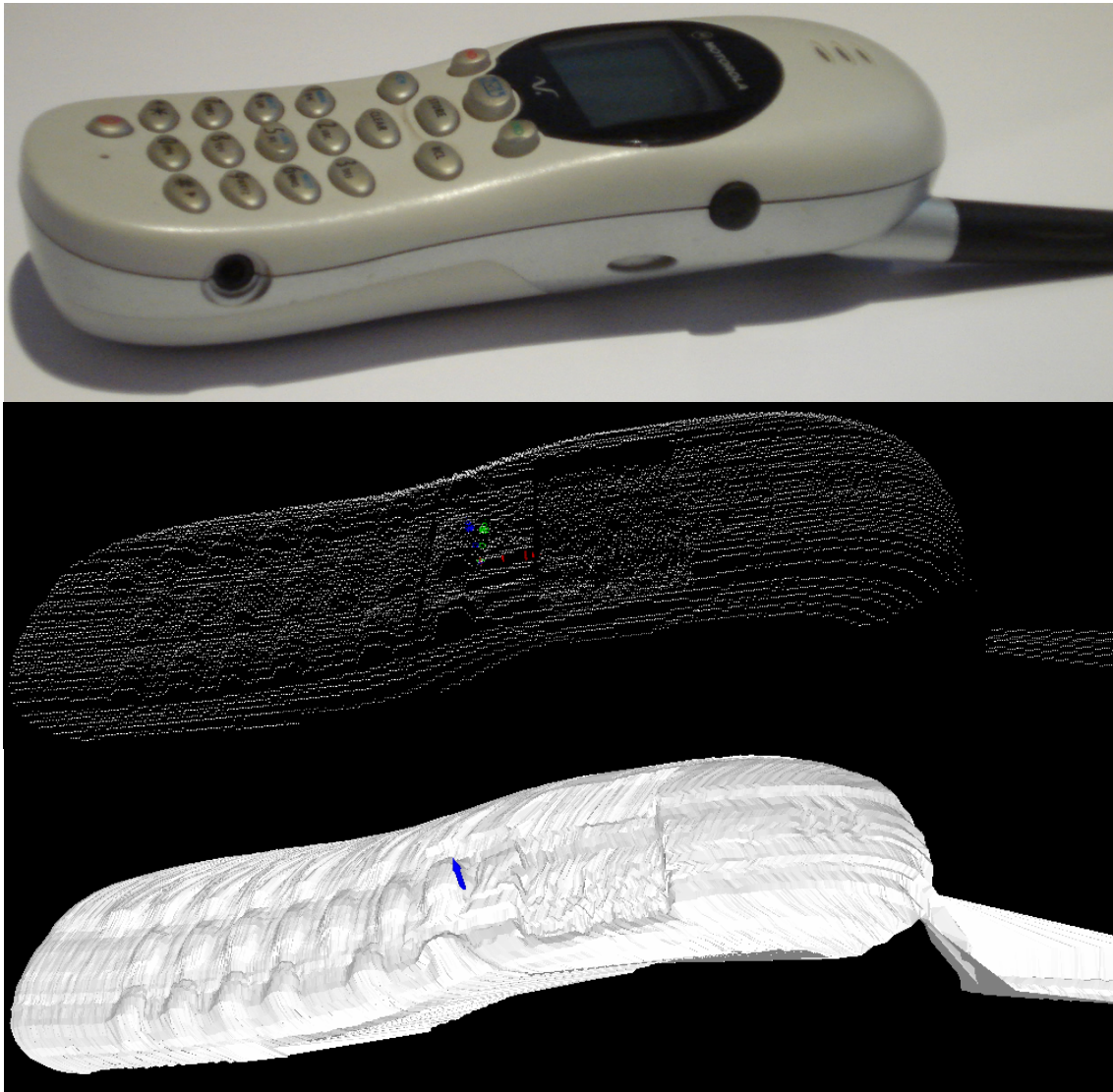


Figura 8.10. Digitalização de um Aparelho Celular.

A sétima e a oitava digitalização é a de um pequeno molde de plástico em forma de um crustáceo estilizado e de um mouse de computador.

Na digitalização do molde plástico, assim como observado na digitalização da antena do aparelho celular, fica clara a inadequação do algoritmo de renderização dos pontos quando as superfícies não são convexas.

O algoritmo de renderização, apresentado na Seção 4.3.2, supõem que as superfícies digitalizadas são sempre contínuas, de forma que, caso haja uma reentrância, um rasgo ou um furo nessa superfície, essa característica, apesar de visível na nuvem de pontos e digitalizada corretamente, não é renderizada corretamente.

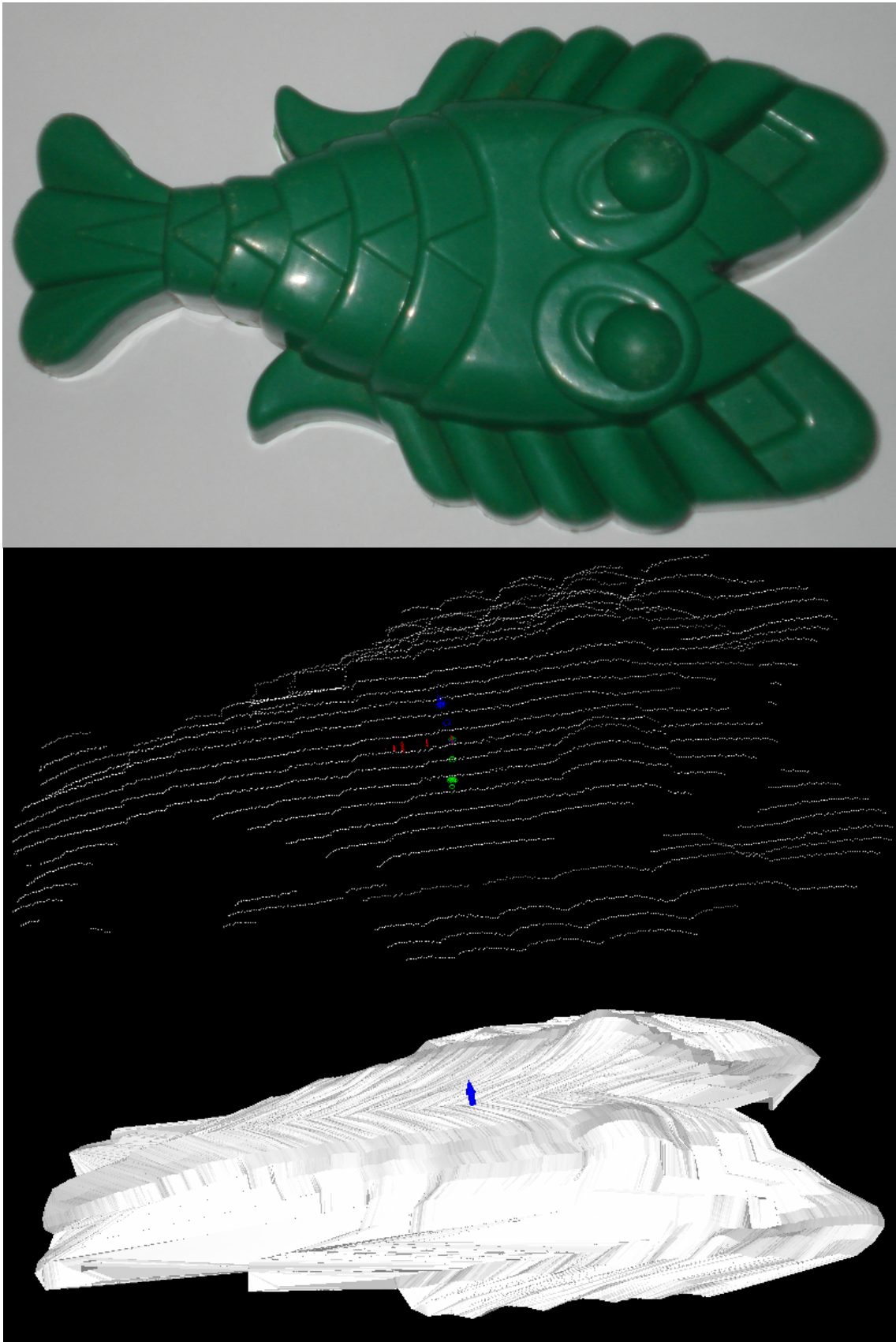


Figura 8.11. Digitalização de um Molde.

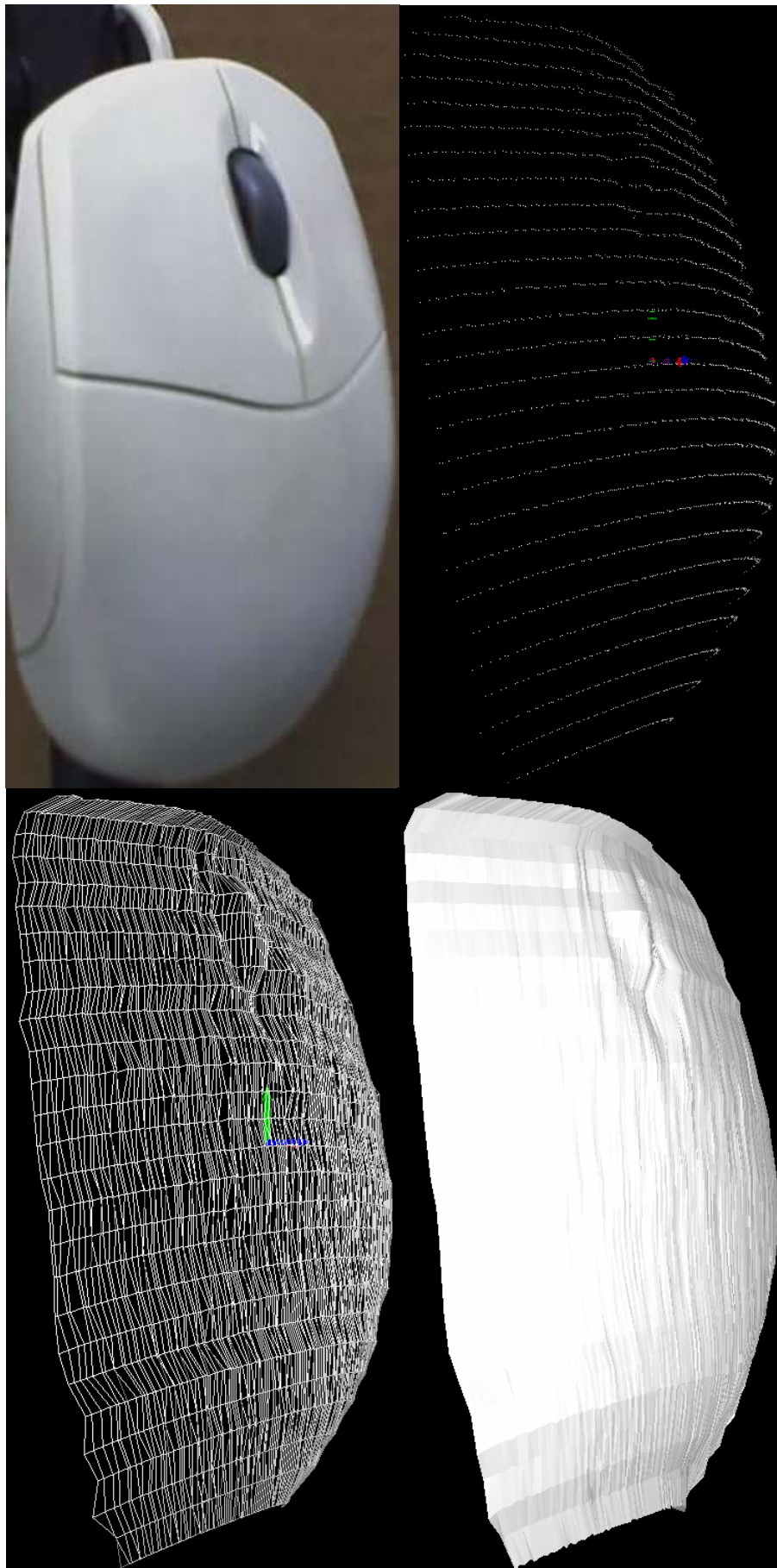


Figura 8.12. Digitalização de um Mouse de Computador.

A nona digitalização apresentada é a digitalização de um vaso.

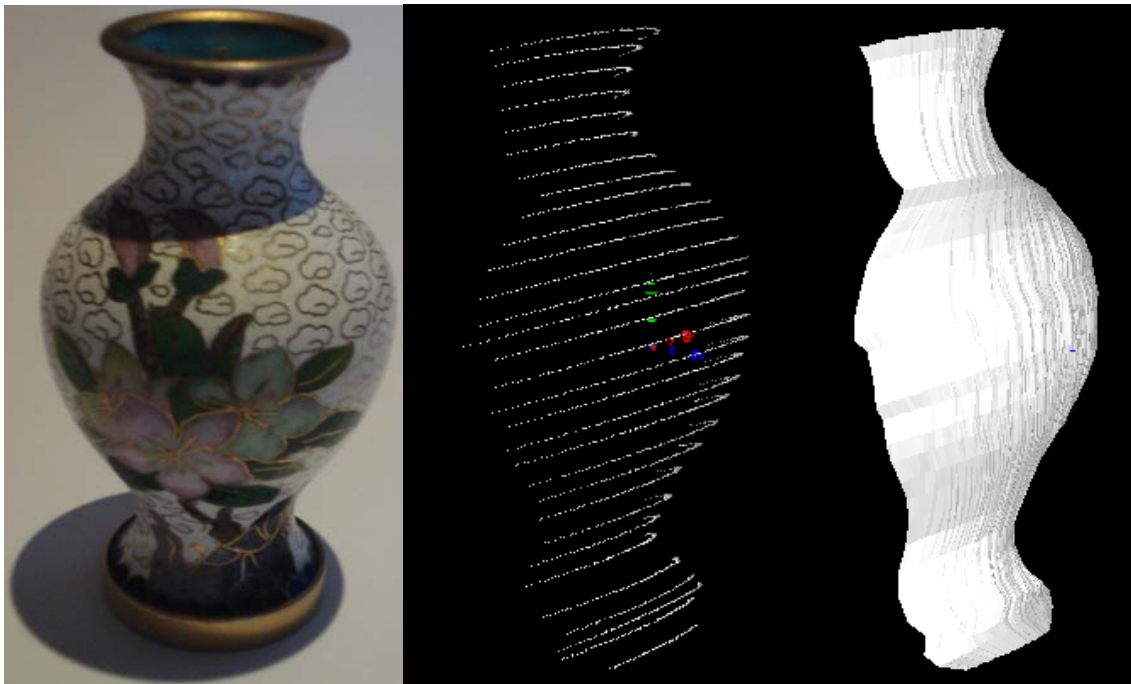


Figura 8.13. Digitalização de um Vaso.

A décima digitalização é a de uma fonte de geração de corrente contínua. Nessa digitalização é apresentada não somente a da superfície do painel frontal desse equipamento, mas também o uso de uma das imagens capturadas da superfície para a atribuição de uma textura a essa superfície. Ou seja, com o uso de câmeras CCD coloridas é possível também a digitalização de uma superfície mantendo suas propriedades de cor.

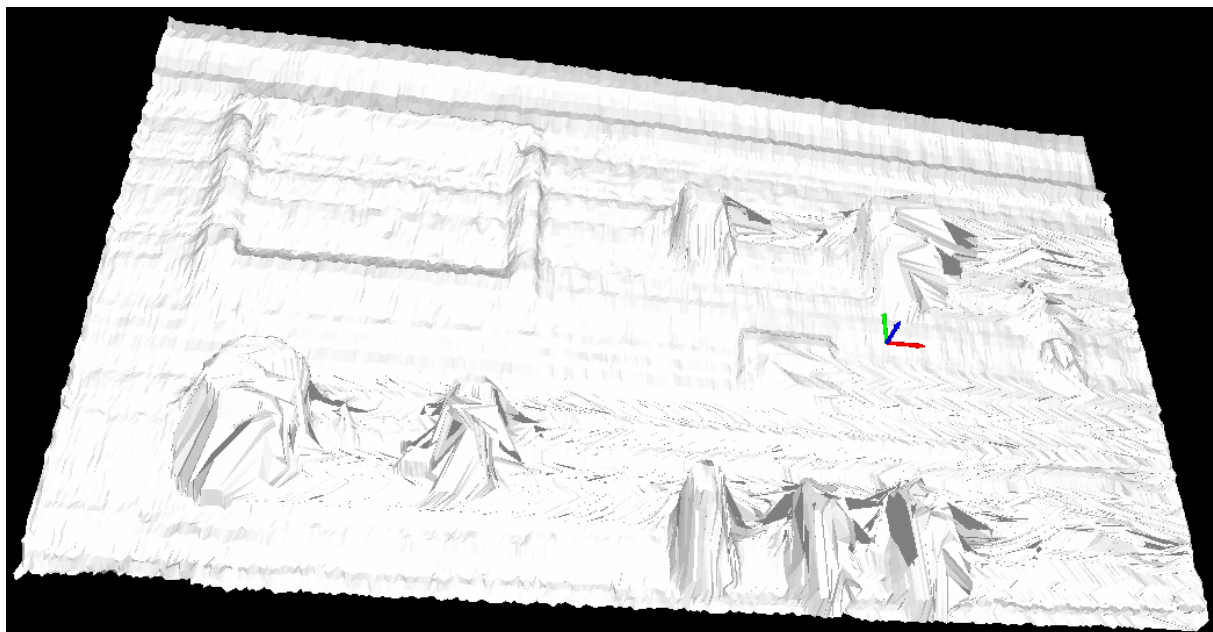


Figura 8.14. Fonte de Geração CC.

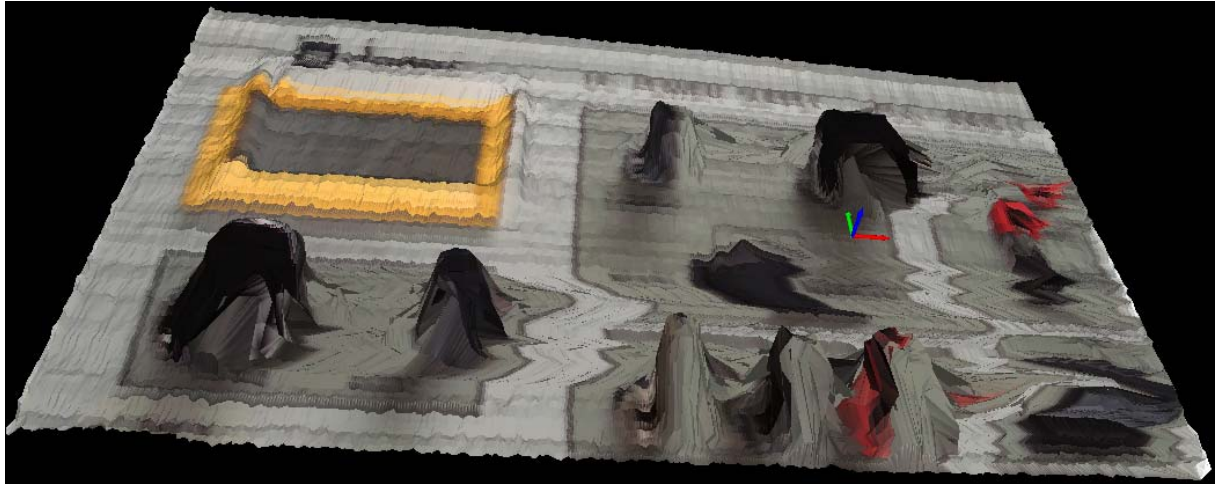


Figura 8.15. Digitalização e Aplicação de Textura.

A décima primeira digitalização apresentada aqui é a de uma chave. Essa digitalização é realizada em duas etapas. Primeiramente é digitalizada uma das faces da chave e, em seguida, com a rotação dessa chave em 180° é realizada uma segunda digitalização. Em seguida, os dados obtidos em ambas as digitalizações são combinados e a chave é reconstruída integralmente.



Figura 8.16. As Duas Faces de uma Chave.

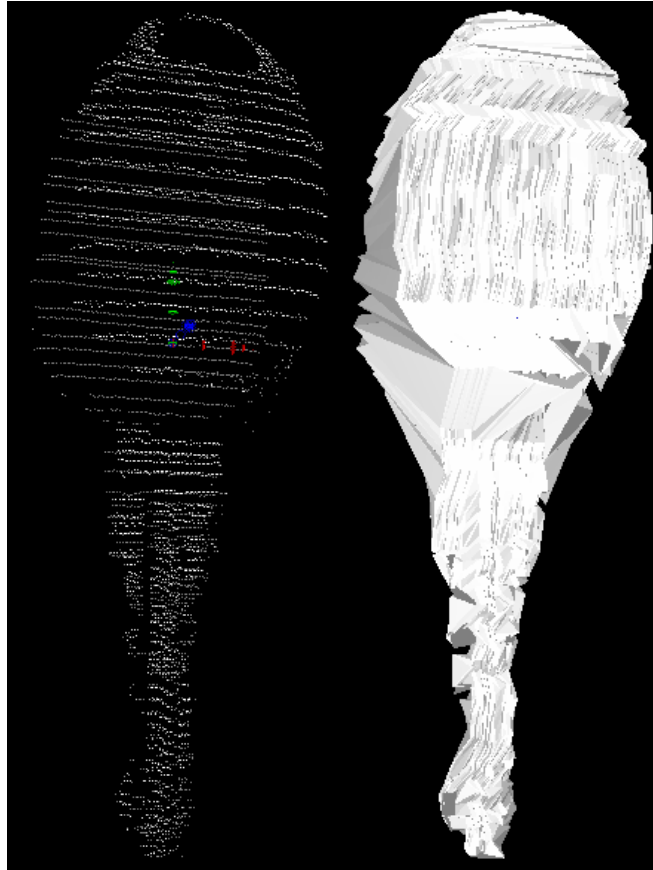
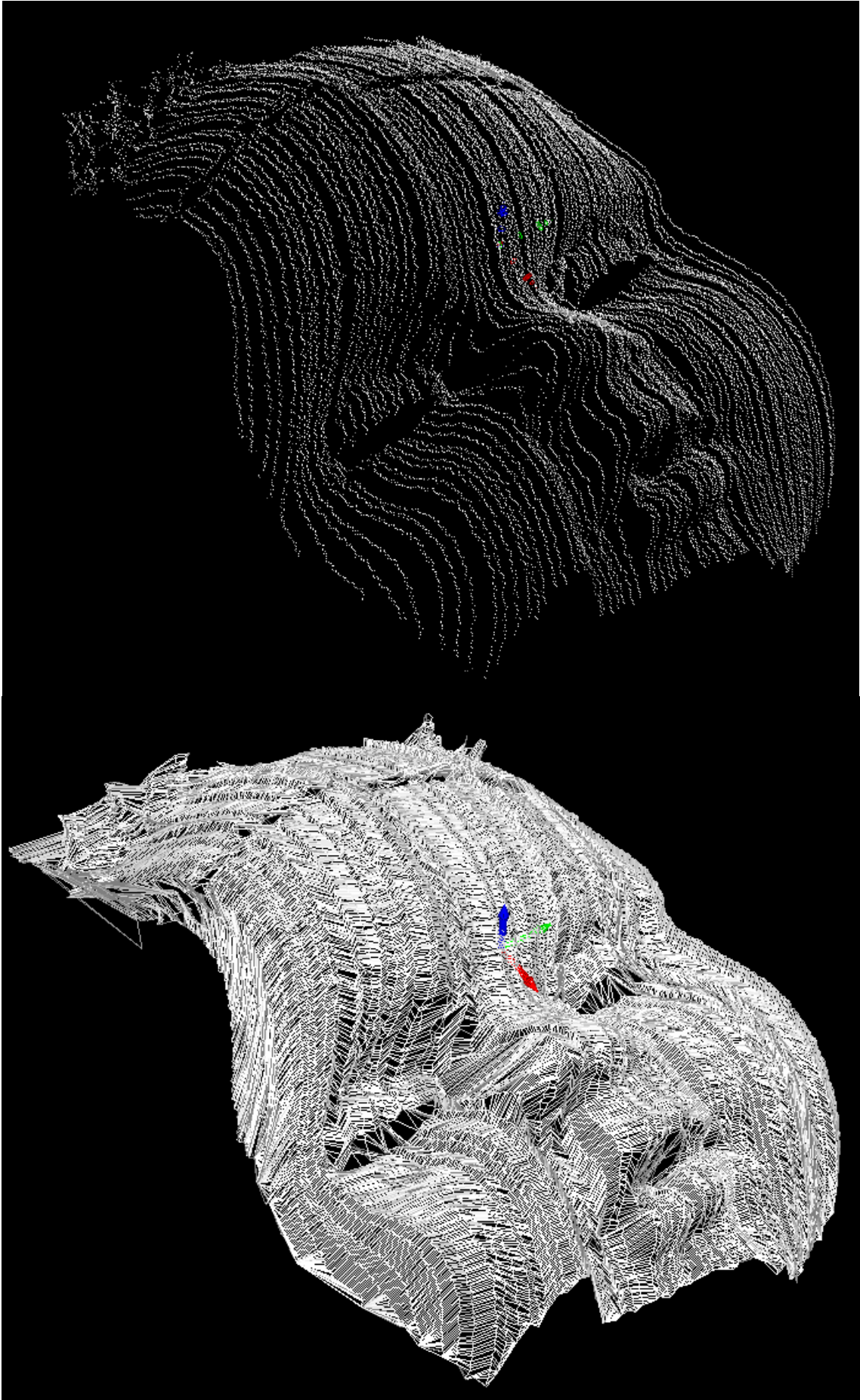


Figura 8.17. Digitalização 3D de uma Chave.

A última digitalização apresentada nesse capítulo é a digitalização da face de uma boneca, onde, mais uma vez, ficam evidentes as limitações do algoritmo de renderização.





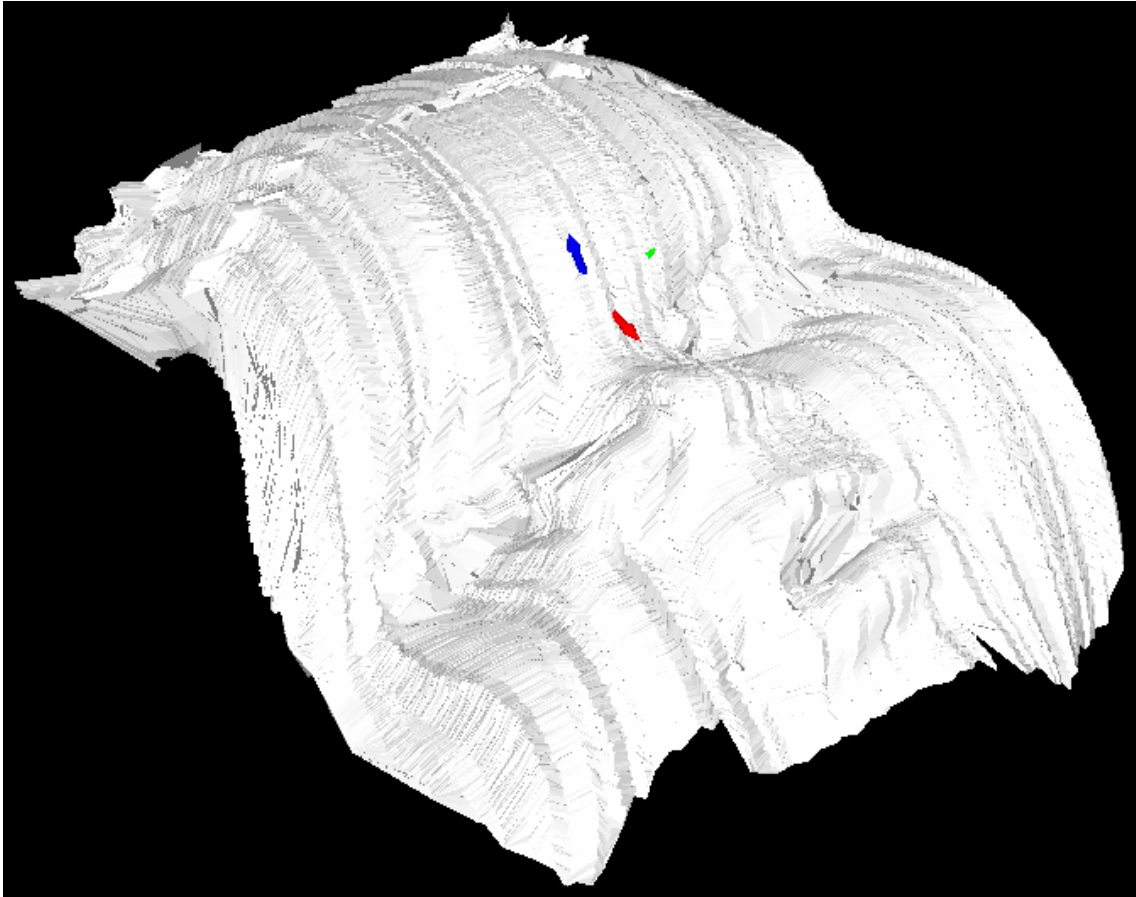


Figura 8.18. Digitalização da Face de uma Boneca.

A digitalização dessa face ilustra o grande potencial do sistema quanto à sua capacidade de digitalização de superfícies complexas, entretanto, antes do uso do sistema em pessoas é necessário realizar-se considerações quanto ao tempo necessário para a varredura completa da face e sua digitalização.

9 CONCLUSÕES

O principal objetivo desse trabalho foi o desenvolvimento de um sensor de digitalização tridimensional de superfícies que apresente uma grande flexibilidade, baixo custo e um nível de precisão que possibilite seu uso em aplicações metrológicas.

Para a realização desse desenvolvimento foi proposto o uso de um sistema de digitalização baseado na varredura laser de superfícies e no uso de técnicas de visão computacional para evitar a necessidade de um sensor de posicionamento angular sobre o diodo que realiza a varredura.

A não utilização de um sensor angular de posição do laser que realiza a varredura e o sistema de auto-calibração desenvolvido são as duas principais características do digitalizador implementado. Enquanto a remoção do sensor reduz os custos do sistema, a auto-calibração torna desnecessário o uso de outros sistemas de medição para realizar a calibração do sistema.

O sistema proposto é apresentado detalhadamente através da descrição de sua arquitetura, seus princípios de triangulação, seus componentes e suas limitações. E seu desempenho é avaliado através de estudos teóricos, simulados e experimentais de precisão.

Com a simulação computacional do sistema foi possível avaliar o processo de digitalização e, através da avaliação teórica das equações que regem esse processo foi possível caracterizar a incerteza de medição do sistema em função de seus parâmetros construtivos. Dessa forma criou-se a possibilidade de projetar o sistema de acordo com o nível de incerteza desejado e, com esse controle, torna-se viável o uso do sistema em aplicações metrológicas.

Já com os ensaios experimentais observou-se a validade da avaliação teórica realizada além do efeito gerado por características como a iluminação ambiente e as propriedades reflexivas das superfícies digitalizadas.

Também através de ensaios teóricos e experimentais foi avaliado o desempenho do sistema de sensoriamento da angulação através da interseção dos lasers e, uma vez que o ponto de interseção dos lasers seja obtido corretamente, o sistema apresentou um nível de precisão muito superior ao oferecido normalmente por *encoders* e um custo reduzido.

Concluindo, o sistema desenvolvido não apenas atendeu aos objetivos propostos, mas também possibilitou o desenvolvimento de ferramentas de processamento de imagens, de processamento de nuvens de pontos e de calibração e simulação de câmeras, sendo essas ferramentas de grande valia para o desenvolvimento de futuros projetos na área de visão computacional.

10 TRABALHOS FUTUROS

O desenvolvimento de um sistema de digitalização tridimensional de superfícies abre as portas para o desenvolvimento de várias frentes de pesquisa nas mais diversas áreas do conhecimento. Não só é possível a realização de pesquisas para melhoria do sistema de digitalização, mas esse pode ser utilizado como ferramenta em pesquisas de computação gráfica, robótica, navegação e muitas outras.

10.1 DESENVOLVIMENTOS FUTUROS DO SISTEMA VISSCAN-3D

Baseando-se nos resultados obtidos durante o desenvolvimento do sistema de digitalização são propostas aqui idéias de novas pesquisas para o contínuo desenvolvimento do sistema de digitalização implementado.

- A consideração de efeitos de polarização da luz laser para evitar efeitos de reflexos, observados principalmente na digitalização de superfícies metálicas;
- Estudo de diferentes modelos de câmeras para a determinação do modelo que fornece melhor precisão ao sistema de digitalização;
- Estudo sobre o uso de fontes de iluminação externas para melhorar a qualidade das imagens capturadas;
- Avaliação do uso de câmeras coloridas para a reconstrução de superfícies mantendo sua coloração original;
- Estudo de técnicas para contornar problemas de oclusão;
- Adaptação do código fonte do sistema para uso de representações de ponto flutuante mais precisas (*double, long double, ...*);
- Ampliação das linguagens de *scripts* para processamento de imagens e nuvens de pontos e estudos sobre a geração automática desses *scripts* a partir das condições externas observadas pelo sistema de digitalização;

10.2 NOVAS LINHAS DE PESQUISA

A possibilidade de se digitalizar um objeto qualquer e obter uma nuvem de pontos tridimensional que representa esse objeto torna possível o desenvolvimento de várias frentes de pesquisa. Aqui são listadas algumas dessas novas possibilidades e são sugeridas novas frentes de pesquisa e desenvolvimento.

- A digitalização de várias vistas de um objeto visando a digitalização tridimensional completa desse objeto;

- Estudo de diferentes técnicas de interpolação e renderização de superfícies;
- Identificação e localização de objetos baseado em suas características tridimensionais para manipulação desses objetos por sistema robóticos;
- Reconhecimento facial de pessoas baseado em características tridimensionais de faces;
- Detecção de falhas em superfícies;
- Detecção de chanfros para a determinação automática de caminhos de soldagem de robôs;
- Avaliação da geometria tridimensional de cordões de solda para controle de qualidade do cordão e avaliação da quantidade de material depositado;

REFERÊNCIAS BIBLIOGRÁFICAS

- Alves, A.S., 1996, “Metrologia Geométrica”, Editora Fundação Calouste Gulbenkian.
- Baltsavias, E.P., 1999, “A Comparison Between Photogrammetry and Laser Scanning”, *Journal of Photogrammetry & Remote Sensing*, Vol. 54, pp. 83-94.
- Ballard, D.H., Brown, C.M., 1982, “Computer Vision”, Editora Prentice-Hall.
- Barata, J., Santos, J., 1999, “Dominando 110% 3D Studio Max 2.5”, Editora Brasport.
- Barhak, J., Fischer, A., 2001, “Parameterization for Reconstruction of 3D Freeform Objects from Laser-Scanned Data based on a PDE Method”, *The Visual Computer*, Vol. 17, pp. 353-369.
- Bösemann, W., 2005, “Advances in Photogrammetric Measurement Solutions”, *Computers in Industry*, Vol. 56, No. 8, pp. 886-893.
- Boyce, W.E., DiPrima, R.C., 2002, “Equações Diferenciais Elementares e Problemas de Contorno”, Editora LTC.
- Bussab, W.O., Morettin, P.A., 1987, “Estatística Básica”, Editora Atual.
- Cantzler, H., Fisher, R.B., 2001, “Comparison of HK and SC Curvature Description Methods”, *Third International Conference on 3-D Digital Imaging and Modeling*, pp. 285-291.
- Clark, J., Trucco, E., Wolff, L.B., 1997, “Using Light Polarization in Laser Scanning”, *Image and Vision Computing*, Vol. 15, pp. 107-117.
- Davis, H.M., Khotanzad, A., Flamig, D.P., Harms, S.E., 1995, “Curvature Measurement of 3D Objects: Evaluation and Comparison of Three Methods”, *International Conference on Image Processing*, Vol. 2, pp. 2627.
- Dennis J.E., Schnabel, R.B., 1983, “Numerical Methods for Unconstrained Optimization and Nonlinear Equations”, Editora Prentice Hall.
- Dimas, E., Briassoulis, D., 1999, “3D Geometric Modelling based on NURBS: A Review”, *Advances in Engineering Software*, Vol. 30, No. 9, pp. 741-751.
- Dinechin, F., Villard, G., 2006, “High Precision Numerical Accuracy in Physics Research”, *Nuclear Instruments & Methods in Physics Research*, Vol. 559, No. 1, pp 207-210.
- Edmund Optics, 2004, “Optics and Optical Instruments Catalog”.
- Edmund Optics, 2007, “Master Source Book”.
- Edwards, C.H., Penney, D.E., 1998, “Introdução a Álgebra Linear”, Editora LTC.
- Gonzales, R.C., Woods, R.E., 2002, “Digital Image Processing”, Editora Prentice Hall.
- Hallert, B., 1960, “Photogrammetry: Basic principles and General Survey”, Editora McGraw-Hill.

- Hsuehl, W., Antonsson, E.K., 1997, "Automatic High-Resolution Optoelectronic Photogrammetric 3D Surface Geometry Acquisition System", *Machine Vision and Applications*, Vol. 10, pp. 98-113.
- Jacoby, S.L.S, Kowalik, J.S., Pizzo, J.T., 1972, "Iterative Methods for Nonlinear Optimization Problems", Editora Prentice Hall.
- Klette, R., Schlüns, K., Koschan, A., 1996, "Computer Vision – Three Dimensional Data from Images", Ed. Springer.
- Kreyszig, E., 2006, "Advanced Engineering Mathematics", Editora John Wiley & Sons.
- Lampton, M., 1997, "Damping-Undamping Strategies for the Levenberg-Marquardt Non-Linear Least-Squares Method", *Computers in Physics*, Vol. 11, No 1, pp. 110-115.
- Lee, K.H., Woo, H., Suk, T., 2001, "Data Reduction Methods for Reverse Engineering", *International Journal of Advanced Manufacturing Technology*, Vol. 17, pp. 735-743.
- Lenz, R.K., Tsai, R. Y., 1987, "Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3D Machine Vision Metrology", *Proceeding of the IEEE International Conference on Robotics and Automation*, Raleigh, Vol. NC, pp. 68-75.
- Milne, W.E., 1962, "Numerical Solution of Differential Equations", Editora John Wiley & Sons.
- Motta, J.M.S.T, 1999, "Optimized Robot Calibration Using a Vision-Based Measurement System with a Single Camera", Doctor Thesis, School of Industrial and Manufacturing Science, Cranfield University, UK.
- Motta, J.M.S.T, Carvalho, G.C., McMaster, R.S., 2001, "Robot Calibration using a 3D Vision-based Measurement System with a Single Camera", *Robotics and Computer Integrated Manufacturing*, Vol. 17, No. 6, pp. 487-497.
- Motta, J.M.S.T, McMaster, R.S., 2002, "Experimental Validation of a 3-D Vision-Based Measurement System Applied to Robot Calibration", *Journal of the Brazilian Society of Mechanical Sciences*, Vol. 24, No. 3, pp. 220-225.
- Neider, J., Tom, D., Mason, W., 1993, "OpenGL Programming Guide: The Official Guide to Learning OpenGL", Editora Addison-Wesley.
- OpenGL Architecture Review Board, 1992, "OpenGL Reference Manual: The Official Reference Document for OpenGL", Editora Addison-Wesley.
- Optical Society of America, 1978, "Handbook of Optics", Editora McGraw-Hill.
- Parker, J.R., 1997, "Algorithms for Image Processing and Computer Vision", Ed. John Wiley & Sons.
- Petkovic, D., 1989, "The Need for Accuracy Verification of Machine Vision Algorithms and Systems", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 439-440.

- Petrov, M., Talapov, A., Robertson, T., Lebedev, A., Zhilyaev, A., Polonskiy, L., 1998, "Optical 3D Digitizers: Bringing Life to the Virtual World", IEEE Computer Graphics and Applications, Vol. 18, No. 3, pp. 28-37.
- Press, W.H., Teukolsky, S.A., Vetterling, S.A., Flannery, B.P., 1992, "Numerical Recipes in C: The Art of Scientific Computing", Ed. Cambridge University Press.
- Ruggiero, M.A.G., Lopes, V.L.R., 1996, "Cálculo Numérico – Aspectos Teóricos e Computacionais", Editora Makron Books.
- Sansoni, G., Docchio, F., 2004, "Three-Dimensional Optical Measurements and Reverse Engineering for Automotive Applications", Robotics and Computer-Integrated Manufacturing, Vol. 20, pp. 359-367.
- Sufroniou, M., Spaletta, G., 2005, "Precisie Numerical Computation ", The Journal of Logic and Algebraic Programming, Vol. 64, No. 1, pp. 113- 134.
- Sun,W.,Cooperstock, J.R., 2005, "Requirements for Camera Calibration: Must Accuracy come wit a High Price?", IEEE Workshop on Applications of Computer Vision, Vol. 1, pp. 356-361.
- Swokowski, 1995, "Cálculo com Geometria Analítica", Vol. 2, Editora Makron Books.
- Tognolaa, G., Parazzinia, M., Sveltob, C., Ravazzania, P., Grandoria, F., 2003, "A Fast and Reliable System for 3D Surface Acquisition and Reconstruction", Image and Vision Computing, Vol. 21, pp. 295-305.
- Tomasi, C., Kanade, T., 1991, "Shape and Motion from Image Streams: A Factorization Method – Part 3 – Detection and Tracking of Point Features", Technical Report CMU-CS-91-132, Carnegie Mellon University, Pittsburgh.
- Tsai, R. Y., 1987, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the Shelf TV Cameras and Lenses", IEEE International Journal of Robotics and Automation, Vol. RA-3, No. 4, pp. 323-344.
- Trucco, E., Verri, A., 1998, "Introductory Techniques for 3D Computer Vision", Ed. Prentice Hall.
- Zhuang, H., Roth, Z.S., 1996 "Camera-Aided Robot Calibration", Ed. CRC Press.

APÊNDICES

	Pág.	
Apêndice A	Diferenciação Numérica	131
Apêndice B	Solução de Sistemas de Equações	135
Apêndice C	Formatos de Arquivos 3D	139
Apêndice D	Funções para Processamento de Imagens	149
Apêndice E	Funções para Processamento de Nuvens de Pontos	170
Apêndice F	O Processo de Calibração do Sistema Passo a Passo	181

APÊNDICE A: Diferenciação Numérica

Existem muitos problemas importantes dentro da engenharia onde métodos analíticos não podem ser aplicados ou sua aplicação apresentaria níveis elevados de complexidade (Boyce e DiPrima, 2002). Nesses casos é necessário o uso de abordagens alternativas para a solução desses problemas e uma dessas abordagens é a utilização de métodos numéricos.

A diferenciação numérica é a computação dos valores das derivadas de uma função f a partir dos valores numéricos dessa função e é amplamente descrita na literatura (Kreyszig, 2006, Boyce e DiPrima, 2002, Trucco e Verri, 1998, Ruggiero e Lopes, 1996, Press *et. al.*, 1992, Milne, 1962).

Diferentemente da integração numérica, a diferenciação é uma operação instável, que pode vir a gerar erros numéricos que invalidem os resultados obtidos. Press *et. al.* (1992) e Milne (1962) descrevem várias técnicas sofisticadas que buscam aprimorar a precisão dessa operação e controlar o erro gerado. Esse trabalho apresenta uma introdução à essa área do cálculo numérico, focando nas necessidades da derivação de imagens e nuvens de pontos.

O conceito básico por trás da diferenciação numérica está na aproximação do limite matemático que define o conceito de derivada:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (\text{A.1})$$

Considerando uma função real $f(x)$ diferenciável, da qual são conhecidas N amostras x_1, x_2, \dots, x_N , é possível expandir essa função escrevendo-a como uma série de Taylor conforme ilustrado na Equação (A.2) e na Equação (A.3) (Kreyszig, 2006, Trucco e Verri, 1998, Swokowski, 1995).

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + O(h^3) \quad (\text{A.2})$$

$$f(x-h) = f(x) - hf'(x) + \frac{1}{2}h^2 f''(x) + O(h^3) \quad (\text{A.3})$$

Combinando-se essas duas equações obtém-se uma aproximação inicial para primeira derivada da função $f(x)$ em um ponto x_i .

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2), \quad (\text{A.4})$$

onde $O(h^2)$ significa que o erro de truncamento devido a simplificação do polinômio infinito de Taylor tende a zero a medida que h^2 tende a zero.

De maneira similar, a partir da Equação (A.2) e da Equação (A.3) obtem-se uma aproximação para a derivada de segunda ordem da função $f(x)$:

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h) \quad (\text{A.5})$$

A Equação (A.4) e a Equação (A.5) ilustram uma das formas mais simples de diferenciação numérica de uma função e são conhecidas como aproximações de diferença centrais. Elas utilizam um intervalo centrado no ponto onde deseja-se calcular a derivada da função, entretanto também é possível utilizar-se intervalos não simétricos, gerando aproximações como as ilustradas na Equação (4.6) e na Equação (A.7). Essas aproximações entretanto apresentam um erro de truncamento maior que o apresentado pela aproximação central.

$$f'(x) = \frac{f(x+h) - f(x)}{h} + O(h) \quad (\text{A.6})$$

$$f'(x) = \frac{f(x) - f(x-h)}{h} + O(h) \quad (\text{A.7})$$

Utilizando esse mesmo desenvolvimento é possível obter-se aproximações que apresentem erros de truncamento ainda menores. Para isso basta realizar-se a expansão da série de Taylor da função $f(x)$ até um termo de ordem superior (Trucco e Verri, 1998).

Seguindo essa estratégia obtem-se as aproximações centrais utilizadas nesse trabalho para o cálculo das derivadas tanto das imagens quanto das nuvens de pontos.

$$f'(x) = \frac{-f(x+2h) + 8f(x+h) - f(x-h) + f(x-2h)}{12h} + O(h^4) \quad (\text{A.8})$$

$$f''(x) = \frac{-f(x+2h) + 16f(x+h) - 30f(x) + 16f(x-h) - f(x-2h)}{12h^2} + O(h^3) \quad (\text{A.9})$$

Quando se trabalha com imagens, é possível simplificar essas relações. Uma vez que as imagens utilizadas são digitais, as amostras x_1, x_2, \dots, x_N são pixels da imagem. Ou seja, o intervalo de amostras é uniforme e igualmente espaçado de um pixel. Ou seja:

$$x_{i+1} = x_i + 1 \quad (\text{A.10})$$

Logo, as aproximações centrais para a diferenciação numérica de imagens pode ser simplificada como:

$$f'(x_i) = \frac{f(x_{i-2}) - 8f(x_{i-1}) + 8f(x_{i+1}) - f(x_{i+2})}{12} \quad (\text{A.11})$$

$$f''(x_i) = \frac{-f(x_{i-2}) + 16f(x_{i-1}) - 30f(x_i) + 16f(x_{i+1}) - f(x_{i+2})}{12} \quad (\text{A.12})$$

Extrapolando os conceitos apresentados acima utilizando funções monovariáveis para funções bivariáveis, tem-se as equações utilizadas para a derivação de uma nuvem de pontos $F(x, y)$ e para a derivação de uma imagem $I(x, y)$:

$$\frac{\partial F(x_i, y_i)}{\partial x} = \frac{F(x_{i-2}, y_i) - 8F(x_{i-1}, y_i) + 8F(x_{i+1}, y_i) - F(x_{i+2}, y_i)}{12(x_{i+2} - x_{i-2})} \quad (\text{A.13})$$

$$\frac{\partial F(x_i, y_i)}{\partial y} = \frac{F(x_i, y_{i-2}) - 8F(x_i, y_{i-1}) + 8F(x_i, y_{i+1}) - F(x_i, y_{i+2})}{12(y_{i+2} - y_{i-2})} \quad (\text{A.14})$$

$$\frac{\partial^2 F(x_i, y_i)}{\partial x^2} = \frac{-F(x_{i-2}, y_i) + 16F(x_{i-1}, y_i) - 30F(x_i, y_i) + 16F(x_{i+1}, y_i) - F(x_{i+2}, y_i)}{12(x_{i+2} - x_{i-2})} \quad (\text{A.15})$$

$$\frac{\partial^2 F(x_i, y_i)}{\partial y^2} = \frac{-F(x_i, y_{i-2}) + 16F(x_i, y_{i-1}) - 30F(x_i, y_i) + 16F(x_i, y_{i+1}) - F(x_i, y_{i+2})}{12(y_{i+2} - y_{i-2})} \quad (\text{A.16})$$

$$\frac{\partial I(x_i, y_i)}{\partial x} = \frac{I(x_{i-2}, y_i) - 8I(x_{i-1}, y_i) + 8I(x_{i+1}, y_i) - I(x_{i+2}, y_i)}{12} \quad (\text{A.17})$$

$$\frac{\partial I(x_i, y_i)}{\partial y} = \frac{I(x_i, y_{i-2}) - 8I(x_i, y_{i-1}) + 8I(x_i, y_{i+1}) - I(x_i, y_{i+2})}{12} \quad (\text{A.18})$$

$$\frac{\partial^2 I(x_i, y_i)}{\partial x^2} = \frac{-I(x_{i-2}, y_i) + 16I(x_{i-1}, y_i) - 30I(x_i, y_i) + 16I(x_{i+1}, y_i) - I(x_{i+2}, y_i)}{12} \quad (\text{A.19})$$

$$\frac{\partial^2 I(x_i, y_i)}{\partial y^2} = \frac{-I(x_i, y_{i-2}) + 16I(x_i, y_{i-1}) - 30I(x_i, y_i) + 16I(x_i, y_{i+1}) - I(x_i, y_{i+2})}{12} \quad (\text{A.20})$$

Uma característica peculiar da diferenciação de imagens digitais utilizando essas equações está na possibilidade de se realizar essa operação através da convolução das imagens com uma máscara (Trucco e Verri, 1998, Parker, 1997). Para o caso particular da Equação (A.17), da Equação (A.18), da Equação (A.19) e da Equação (A.20) as máscaras que implementam essas operações estão ilustradas na Figura A.1 e na Figura A.2. Para realizar o cálculo das derivadas da imagem em relação a x ou y basta convolucionar essas máscaras com as linhas ou colunas da imagem.

-1	8	-8	1
----	---	----	---

Figura A.1. Máscara para Cálculo da Primeira Derivada de uma Imagem.

-1	16	-30	16	-1
----	----	-----	----	----

Figura A.2. Máscara para Cálculo da Segunda Derivada de uma Imagem.

APÊNDICE B: Solução de Sistemas de Equações

A solução de sistemas de equações não-lineares é um problema presente em diversas áreas de conhecimento, inclusive na estimação não-linear e na calibração de câmeras. O ajuste dos parâmetros de modelagem a partir de dados experimentais obtidos da placa de calibração envolve a solução de sistemas lineares e não-lineares sobredeterminados.

Existem diversos métodos para a resolução desse tipo de sistemas e um desses métodos é a minimização da soma dos quadrados, ou simplesmente mínimos quadrados. Jacoby et.al. (1972) e Dennis e Schnabel (1983) apresentam uma discussão mais detalhada sobre os diferentes métodos e algoritmos de resolução.

B.1 Decomposição de Valores Singulares (SVD)

Dado um sistema linear sobre-determinado de equações na forma $Ax = b$, existe uma série de métodos para solucionar-se esse sistema. Um desses métodos é o método dos mínimos quadrados. Entretanto, o método dos mínimos quadrados, dependendo da estratégia de resolução escolhida, torna-se muito suscetível a erros em consequência de singularidades que venham a ocorrer no sistema. O método de Decomposição de Valor Singular (SVD) soluciona o problema de mínimos quadrados evitando essas singularidades oferecendo uma solução mais confiável e mais robusta. O SVD baseia-se no seguinte teorema da álgebra linear:

“Qualquer matriz A , $m \times n$, onde o número de linha m é maior ou igual ao número de colunas n , pode ser escrita como o produto de uma matriz U , $m \times n$, coluna-ortogonal, uma matriz W diagonal $n \times n$ onde os valores singulares são expressos e a transposta de uma matriz ortogonal V , $n \times n$.”(Press et.al., 1992)

De forma simplificada, o teorema diz que uma matriz A , $m \times n$ pode ser escrita conforme a Equação (B.1).

$$\begin{pmatrix} A \end{pmatrix}_{m \times n} = \begin{pmatrix} U \end{pmatrix}_{m \times n} \begin{pmatrix} w_1 & & \\ & \dots & \\ & & w_n \end{pmatrix}_{n \times n} \begin{pmatrix} V^T \end{pmatrix}_{n \times n} \quad (\text{B.1})$$

Uma vez decomposta a matriz A , a solução do sistema é dada pela Equação (B.2).

$$\begin{pmatrix} x \end{pmatrix}_{m \times 1} = \begin{pmatrix} V \end{pmatrix}_{n \times n} \begin{pmatrix} 1/w_1 & & \\ & \dots & \\ & & 1/w_n \end{pmatrix}_{n \times n} \begin{pmatrix} U^T \end{pmatrix}_{n \times m} \begin{pmatrix} b \end{pmatrix}_{m \times 1} \quad (\text{B.2})$$

A decomposição de valores singulares é uma fatorização matricial de alto custo computacional, mas de grande poder. (Dennis, Schnabel, 1983)

O algoritmo SVD é uma alternativa confiável para a solução de sistemas lineares sobre-determinados, entretanto não tem um comportamento previsível quando utilizado na solução de sistemas não lineares, muitas vezes tornando-se instável.

B.2 Método de Levenberg-Marquardt

Um dos algoritmos mais populares para a resolução de sistemas de equações não-lineares através da minimização da soma dos quadrados é o algoritmo de Levenberg-Marquardt. Ele é um método que se aplica a uma grande variedade de problemas não-lineares devido a sua capacidade de adaptação (Lampton, 1997).

O algoritmo de Levenberg-Marquardt é um método iterativo de solução obtido a partir de alterações introduzidas no método de Gauss-Newton de forma a contornar problemas de divergência que esse algoritmo apresenta (Jacoby *et.al.*, 1972).

Esse algoritmo baseia-se na minimização não-linear da soma dos quadrados, o que implica que a função a ser minimizada apresenta a seguinte forma:

$$f(x) = \frac{1}{2} \sum (r_j(x))^2 \quad (\text{B.3})$$

, onde $x = [x_1 \ x_2 \ \dots \ x_n]$ é o vetor de parâmetros a ser otimizado e cada r_j é uma função de x conhecida como resíduo.

Representado $f(x)$ através de um vetor de resíduos tem-se:

$$r(x) = [r_1, r_2, \dots, r_m] \Rightarrow f(x) = \frac{1}{2} \|r(x)\|^2 \quad (\text{B.4})$$

Logo, as derivadas da função $f(x)$ podem ser obtidas utilizando-se a matriz Jacobiana de $r(x)$ conforme as equações ilustradas abaixo:

$$\mathbf{J} = \frac{\partial r_j(x)}{\partial x_i} \Rightarrow \begin{cases} \nabla f(x) = \sum r_j(x) \nabla r_j(x) = \mathbf{J}(x)^T r(x) \\ \nabla^2 f(x) = \mathbf{J}(x)^T \mathbf{J}(x) + \sum r_j(x) \nabla^2 r_j(x) \end{cases} \quad (\text{B.5})$$

Entretanto, uma das propriedades dos problemas de mínimos quadrados é que, supondo-se que os resíduos sejam suficientemente pequenos, o Hessiano($\nabla^2 f(x)$) da função f pode ser obtido diretamente do Jacobiano:

$$\nabla^2 f(x) = \mathbf{J}(x)^T \mathbf{J}(x) \quad (\text{B.6})$$

Uma das técnicas de minimização de funções mais simples e intuitivas é a técnica do gradiente descendente. Ela consiste simplesmente na atualização do vetor de parâmetros x utilizando-se o gradiente da função e um fator de escala λ conforme a Equação (B.7).

$${}^{i+1}x = {}^i x - \lambda \nabla f(x) \quad (\text{B.7})$$

A técnica de gradiente descendente, entretanto, apesar de simples, apresenta vários problemas de convergência. Uma das formas de melhorar o problema de convergência na minimização de uma função é utilizar não somente o gradiente da função $f(x)$, mas também informações sobre sua curvatura, ou seja, utilizar a segunda derivada dessa função. Uma das formas de realizar isso é utilizando o método de Newton para solucionar-se a equação $\nabla f(x) = 0$. Para tanto, expande-se o gradiente da função em torno do estado ${}^i x$ atual do vetor de parâmetros considerando-se somente os termos de primeira e segunda ordem.

$$\nabla f({}^{i+1}x) = \nabla f({}^i x) + ({}^{i+1}x - {}^i x)^T \nabla^2 f({}^i x) + \dots \quad (\text{B.8})$$

Obtem-se então as equações que definem o método de minimização de Gauss-Newton.

$${}^{i+1}x = {}^i x - (\nabla^2 f({}^i x))^{-1} \nabla f({}^i x) \quad (\text{B.9})$$

Os métodos de Gauss-Newton e o método do gradiente descendente são métodos complementares em relação a suas vantagens e desvantagens, de forma que Levenberg (1944) e, posteriormente, Marquardt (1963), baseados nessa observação propuseram uma união desses dois métodos utilizando-se a seguinte regra de atualização para o vetor de parâmetros x :

$${}^{i+1}x = {}^i x - (H + \lambda \text{diag}(H))^{-1} \nabla f({}^i x) \quad (\text{B.10})$$

, onde H é o Hessiano avaliado em ${}^i x$ e λ é o fator de amortecimento do algoritmo.

O fator de amortecimento é um termo crítico para a convergência do método de Levenberg-Marquardt e é o ponto chave da adaptabilidade do método (Lampton, 1997). Quanto maior esse fator, mais o método se aproxima do método do gradiente descendente e quanto menor esse fator, mais o método se aproxima do método de Gauss-Newton.

Através da atualização desse fator em cada passo de iteração, o algoritmo de Levenberg-Marquardt é capaz de navegar através de modelos não-lineares complexos e ainda assim aproximar a solução de minimização com uma velocidade quase quadrática (Lampton, 1997).

Existem diferentes estratégias de atualização do fator de amortecimento, entretanto a mais amplamente utilizado é o amortecimento multiplicativo (Lampton, 1997). A estratégia de amortecimento multiplicativo define um termo de ajuste de amortecimento α e um fator de amortecimento inicial λ_0 e, a cada passo da iteração, o fator de amortecimento λ é alterado conforme a Equação (B.11).

$$\lambda_{i+1} = \begin{cases} \alpha\lambda_i, & f(i x) \geq f(i-1 x) \\ \frac{\lambda_i}{\alpha}, & f(i x) < f(i-1 x) \end{cases} \quad (\text{B.11})$$

Outro ponto importante, não apenas do algoritmo de Levenberg-Marquardt, mas de qualquer método iterativo, é a definição dos critérios de parada. Dois critérios de parada frequentemente utilizados em algoritmos de minimização de funções são a avaliação do resíduo e a avaliação do passo do algoritmo.

Na avaliação do resíduo, a função $f(x)$ é avaliada utilizando-se os parâmetros ${}^i x$ atuais e, caso esses resíduos sejam inferiores a um limite escolhido, as iterações são finalizadas e o vetor ${}^i x$ é dado como solução do problema.

Já na avaliação do passo do algoritmo, é avaliada as alterações realizadas no vetor de parâmetros ${}^i x$ conforme a Equação (B.12). Caso essas alterações sejam inferiores a um limite escolhido, as iterações são finalizadas e o vetor ${}^i x$ é dado como solução do problema.

$$\Delta x = \left\| {}^i x - {}^{i-1} x \right\|^2 \quad (\text{B.12})$$

Apesar de ser um método amplamente utilizado é importante observar que o método de Levenberg-Marquardt não é um método ótimo. Ele é um método heurístico que ao longo dos anos demonstrou-se um método prático e confiável (Lampton, 1997).

APÊNDICE C: Formatos de Arquivos 3D

O mundo real é tridimensional e a arte de representar e reproduzir este mundo de maneira convincente vem evoluindo a passos largos durante os últimos anos graças à utilização de ferramentas em informática de modelagem e visualização realista que permitem a aplicação de efeitos surpreendentes. (Barata, Santos, 1999)

E para trabalhar com esse mundo tridimensional dentro de um computador é necessário armazenar-se essa informação, e é exatamente nesse ponto que a importância dos formatos de arquivos 3D fica evidente. Esses formatos definem como a informação referente a esse mundo tridimensional será armazenada na memória do computador. E, uma vez definido esse modo de armazenamento, todas as etapas de processamento, aplicação de efeitos, manipulação e visualização desses dados deverá trabalhar com esse formato e terá acesso somente as informações definidas pelo formato.

Existe uma grande variedade de formatos de arquivos 3D. Aqui são apresentados alguns formatos consagrados e em seguida são apresentados três formatos desenvolvidos especificamente para uso junto com o sistema de digitalização VISSCAN-3D.

C.1 O Formato 3DS

O formato 3DS é um formato de organização de arquivos binários para a representação de objetos tridimensionais em computadores. Esse formato foi desenvolvido pela Autodesk Inc. e é baseado no uso de malhas triangulares para a construção de objetos no espaço. Até o momento em que esse projeto foi realizado, não existia uma documentação completa que descrevia esse formato, apesar de ser um formato de arquivos amplamente utilizado.

Um arquivo 3DS é formado por uma série de “pedaços” organizados hierarquicamente. Cada pedaço é formado por um cabeçalho que identifica e descreve a informação contida nesse pedaço e por um ponteiro para o próximo pedaço.

O formato 3DS permite a representação de uma série de efeitos para imagens tridimensionais como texturas, materiais, pontos de luz, câmeras e outros, todos organizados hierarquicamente conforme mencionado anteriormente. Na Figura C.1 é possível observar a organização hierárquica e os valores de cabeçalho para alguns elementos principais de um arquivo 3DS.

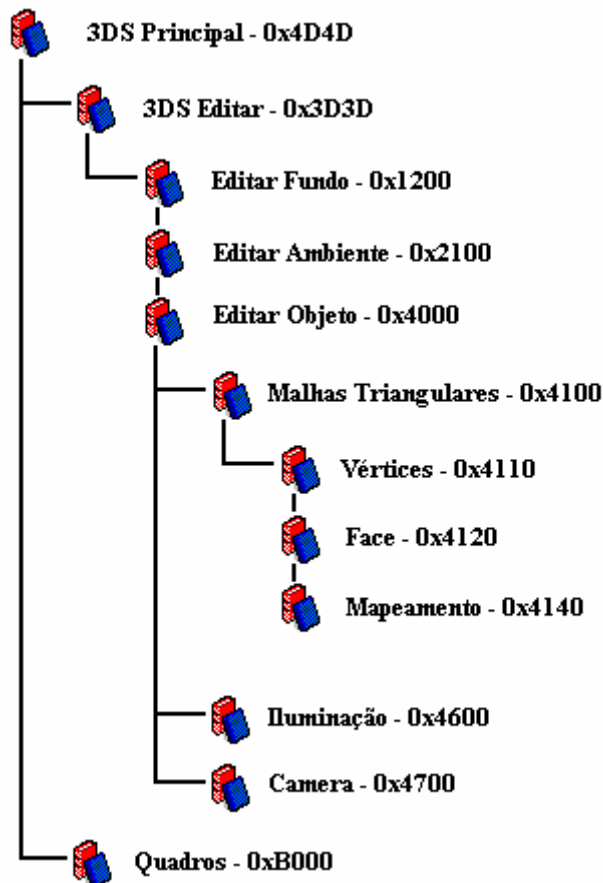


Figura C.1. Organização Hierárquica Simplificada de um Arquivo 3DS.

A organização em pedaços permite uma grande modularidade para o arquivo, facilitando sua leitura de forma que, se algum pedaço é desconhecido ou desnecessário, basta ignorá-lo. Entretanto o formato 3DS apresenta uma peculiaridade na armazenagem de informações binárias. A informação é armazenada com o byte menos significativo à frente. Por exemplo, a informação 2B 7D (2 bytes em hexadecimal) está armazenada em um arquivo 3DS como 7D 2B.. Já em um dado formado por 4 bytes, por exemplo 2B 7D 9A 6F, o arquivo conteria 7D 2B 6F 9A e assim respectivamente.

O principal pedaço de um arquivo 3DS é o Principal. Esse pedaço engloba todo o arquivo, ou seja, toda a informação contida no arquivo está dentro desse pedaço. Dentro desse pedaço destacam-se dois outros pedaços. O pedaço Quadros utilizado para a construção de animações e o pedaço Editar que contém todas as informações referentes a uma única cena estática. Dentro desse pedaço estão armazenadas informações sobre o plano de fundo da imagem, sobre a iluminação ambiente, sobre o posicionamento de câmeras para definir pontos de observação de uma cena e, principalmente, informações sobre os objetos que constituem a cena. Dentro do pedaço Editar Objeto estão armazenadas todas as malhas triangulares responsáveis pela construção do objeto tridimensional, sendo esse pedaço um dos mais importantes dentro do arquivo.

Uma observação importante acerca desse formato é o fato dele ter sido desenvolvido para uso na criação de animações computacionais, possuindo uma série de recursos adicionais ao mero armazenamento da geometria tridimensional de objetos.

C.2 O Formato DWG

O formato DWG, assim como o formato 3DS, é um formato proprietário desenvolvido pela empresa Autodesk para seu sistema de CAD (Computer Aided Drawing), o AutoCAD. Ele é utilizado para o armazenamento de dados de construção de objetos tanto tridimensionais quanto bidimensionais e é hoje um padrão mundial para desenhos técnicos.

C.3 O Formato VRML

O formato VRML (Virtual Reality Modelating Language), ou Linguagem para Modelagem de Realidade Virtual, é um padrão amplamente utilizado na Internet e não apenas descreve objetos tridimensionais, mas é utilizado para a definição completa de aplicativos de realidade virtual, ou seja, VRML não é apenas um formato de arquivos, mas uma linguagem de modelagem.

Por meio desta linguagem, escrita em modo texto, é possível criar objetos tridimensionais, definir parâmetros de cor, transparência, textura, sons, luzes, câmeras e muitos outros.

Devido a sua amplamente difundida aplicação na Internet o padrão VRML97 tornou-se um padrão ISO em 1997.

C.4 O Formato VISSCAN-3D PTO

O formato de arquivo PTO não é um arquivo que contém diretamente a informação a respeito da malha tridimensional da superfície. Esse arquivo serve como ponte entre as imagens capturadas pela câmera do digitalizador VISSCAN-3D e a malha tridimensional.

O arquivo PTO é um arquivo simples e de fácil compreensão e leitura que contém todas as informações que o sistema extrai das imagens. Ele é o resultado final de todo o processamento de imagens realizado contendo as coordenadas em pixels de todos os pontos de cruzamento dos lasers e as coordenadas também em pixels de todos os pontos válidos obtidos do laser móvel responsável pela varredura da superfície. Na Figura C.2 é possível observar detalhadamente a aparência desse arquivo.

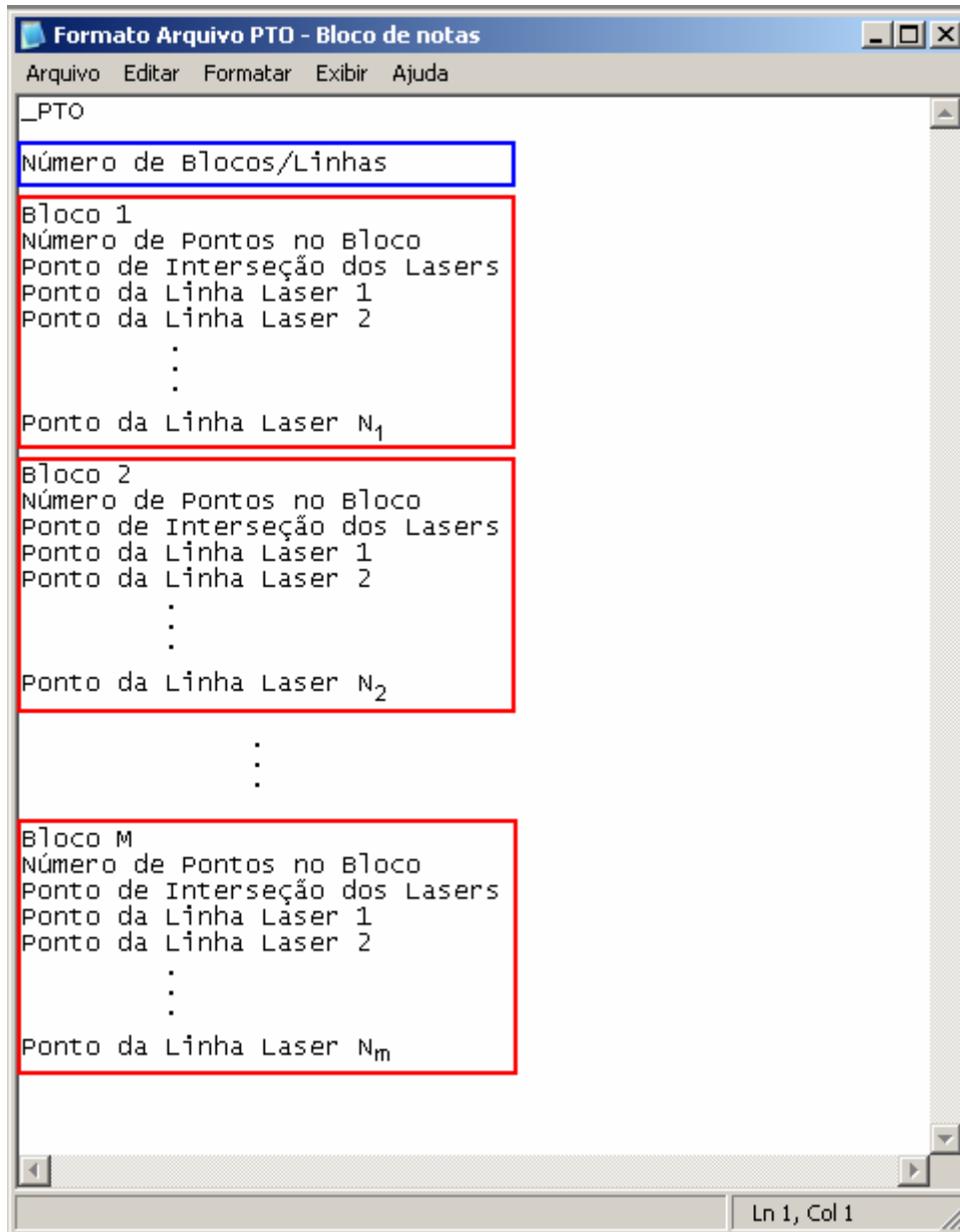


Figura C.2. Formato de Arquivo PTO.

O arquivo PTO é um arquivo de texto, o que torna sua leitura e manipulação direta fácil para um usuário do sistema.

A primeira informação contida em um arquivo PTO é um cabeçalho curto constituído somente por um título de controle. A palavra `_PTO`, que define para o sistema durante a leitura desse arquivo que o arquivo que está sendo lido apresenta o formato PTO.

A segunda informação presente no arquivo refere-se a um dos valores de resolução da malha. Esse valor define quantas linhas da superfície foram digitalizadas e define também quantos blocos de dados estão presentes no arquivo. Cada bloco de dados contém as informações referentes à digitalização de uma linha da superfície. O bloco é composto pela informação do número de pontos existentes nessa linha seguida do ponto de interseção das duas linhas lasers e da listagem de todos esses pontos.

Na Figura C.3 é possível observar um exemplo de arquivo PTO. O arquivo ilustrado nessa gravura contém dois blocos, ou seja, foram utilizadas somente duas linhas para a digitalização da superfície. Também é possível observar que o primeiro bloco do arquivo é formado por somente três pontos, enquanto o segundo bloco do arquivo é formado por quatro pontos.

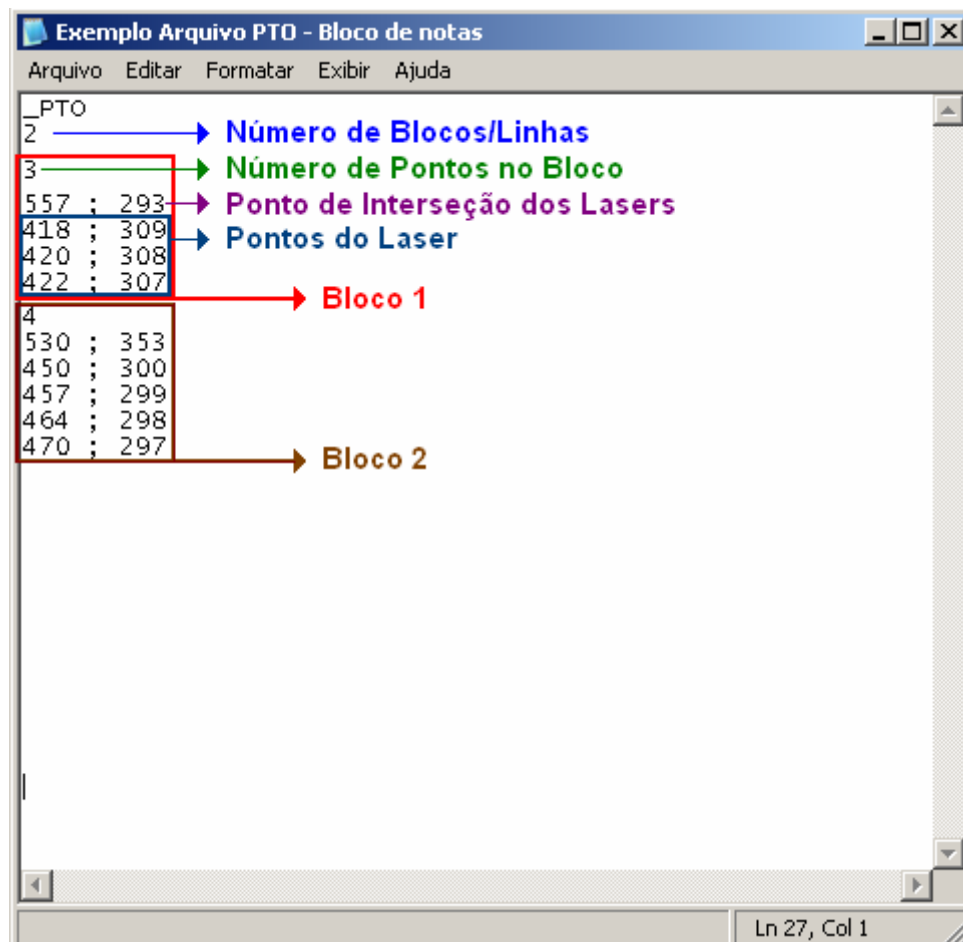


Figura C.3. Exemplo de Arquivo PTO.

Essa arquitetura de organização do arquivo em blocos, ou linhas digitalizadas, concede ao sistema a possibilidade de trabalhar com malhas não uniformes, tirando o máximo proveito de cada linha laser projetada nas superfícies digitalizadas.

C.5 O Formato VISSCAN-3D IMG

O formato de arquivo IMG foi o padrão de organização da informação tridimensional criado e adotado para a representação das malhas tridimensionais das superfícies digitalizadas. Esse arquivo é o resultado final do processo de digitalização, contendo todas as informações sobre a nuvem de pontos obtida.

O arquivo IMG, assim como o arquivo PTO, é um arquivo de texto, o que torna sua leitura e manipulação rápida e fácil para um usuário do sistema. Diferentemente do arquivo PTO que tem todos

seus dados de coordenadas expressos em pixels, o formato IMG já apresenta as informações de coordenadas em unidade S.I. (mm).

Esse formato também apresenta uma arquitetura de organização baseada em blocos, ou linhas digitalizadas, mantendo a capacidade do arquivo PTO de representar malhas não uniformes. Na Figura C.4 é possível observar detalhadamente a aparência desse arquivo.

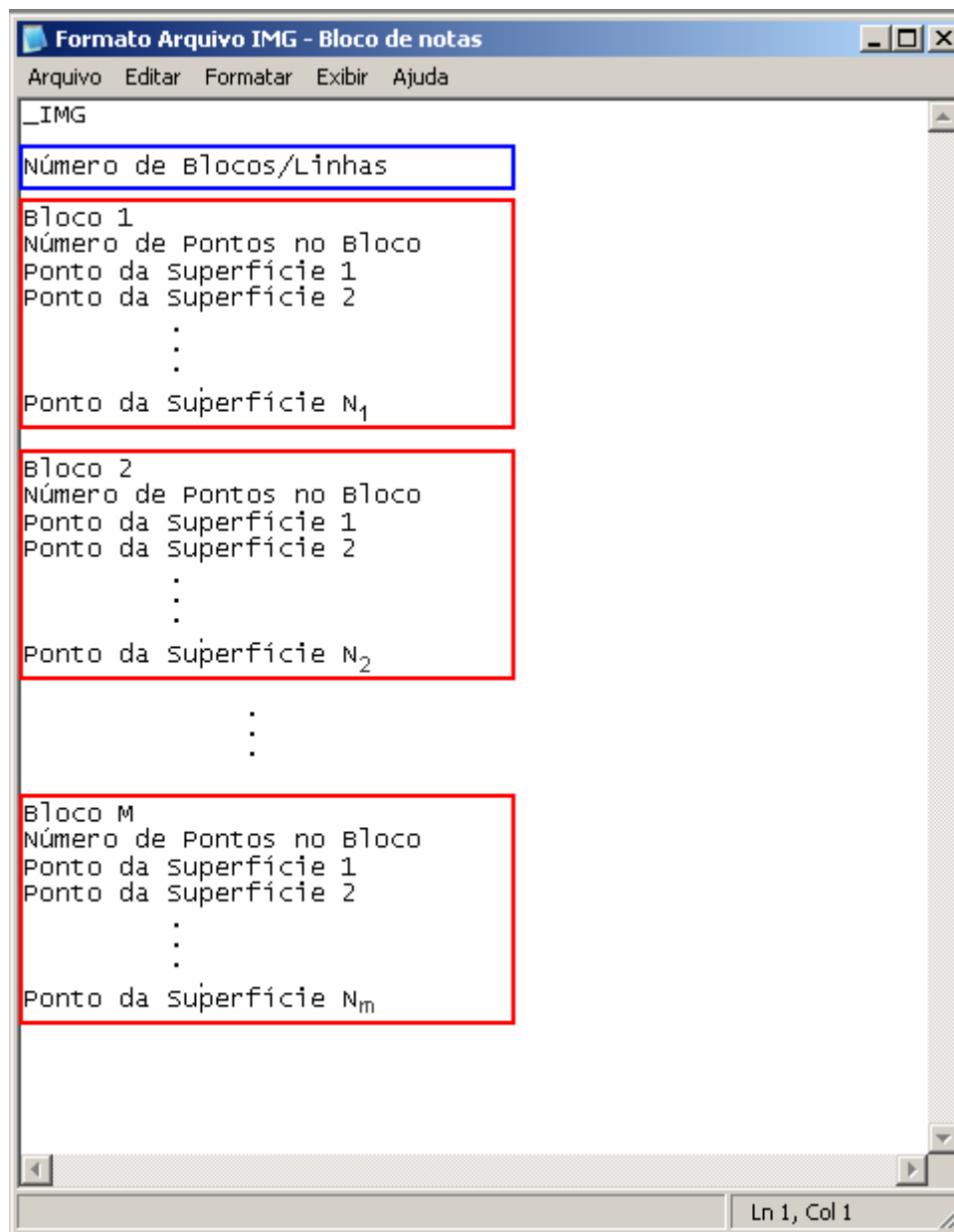


Figura C.4. Formato de Arquivo IMG.

A primeira informação contida em um arquivo IMG é também um cabeçalho curto constituído somente por um título de controle. A palavra `_IMG`, que define para o sistema durante a leitura desse arquivo que o arquivo que está sendo lido apresenta o formato IMG.

A segunda informação presente no arquivo refere-se a um dos valores de resolução da malha. Esse valor define quantas linhas da superfície foram digitalizadas e define também quantos blocos de dados

estão presentes no arquivo. Cada bloco de dados contém as informações referentes à digitalização de uma linha da superfície. O bloco é composto pela informação do número de pontos existentes nessa linha seguida da listagem de todos esses pontos.

Na Figura C.5 é possível observar um exemplo de arquivo IMG. O arquivo ilustrado nessa gravura contém dois blocos, ou seja, foram utilizadas somente duas linhas para a digitalização da superfície. Também é possível observar que o primeiro bloco do arquivo é formado por somente três pontos, enquanto o segundo bloco do arquivo é formado por quatro pontos.

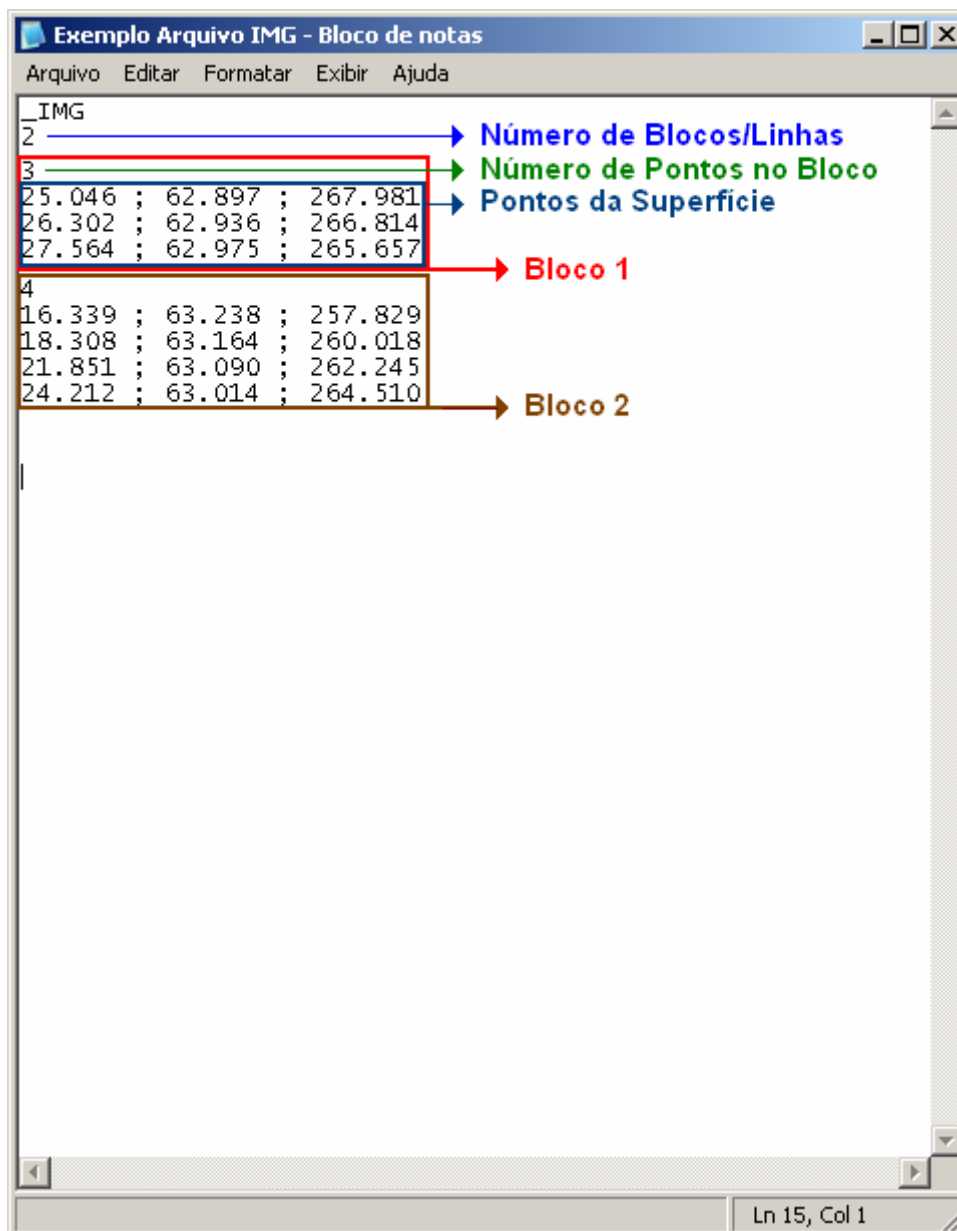


Figura C.5. Exemplo de Arquivo IMG.

C.6 O Formato VISSCAN-3D DEF

O formato de arquivo DEF é uma extensão do formato IMG e também é um padrão de organização da informação tridimensional criado e adotado para a representação das malhas tridimensionais das superfícies digitalizadas.

Esse arquivo, assim como o arquivo IMG, contém todas as informações sobre a nuvem de pontos obtida. Entretanto ele contém ainda um valor de informação associado a cada ponto da malha. A agregação dessa informação a cada ponto da malha torna o formato DEF um formato mais dispendioso do ponto de vista computacional, mas torna o formato capaz de discriminar cada um dos pontos digitalizados. Por exemplo, é possível classificar cada um dos pontos da superfície como defeituosos ou como normais.

O arquivo DEF mantém o formato de arquivo de texto, assim como os outros padrões desenvolvidos, o que torna sua leitura e manipulação rápida e fácil através de um editor de texto qualquer. Entretanto, a função principal do arquivo DEF não é somente o armazenamento da informação, ele é o padrão utilizado pelo sistema de processamento de malhas implementado e é constantemente acessado e modificado durante esse processamento. A alteração do padrão DEF para o formato de arquivo binário seria uma alteração vantajosa do ponto de vista da eficiência computacional, entretanto optou-se aqui pela manutenção da facilidade de manipulação e leitura típicas do formato de texto.

Assim como o formato IMG do qual é derivado, o arquivo DEF tem seus dados de coordenadas expressos em unidades S.I. (mm) e apresenta uma arquitetura de organização baseada em blocos, ou linhas digitalizadas. Na Figura C.6 é possível observar detalhadamente a aparência desse arquivo.

```
_DEF
Número de Blocos/Linhas
Bloco 1
Número de Pontos no Bloco
Informação 1 ; Ponto da Superfície 1
Informação 2 ; Ponto da Superfície 2
      :
      :
Informação N1 ; Ponto da Superfície N1
Bloco 2
Número de Pontos no Bloco
Informação 1 ; Ponto da Superfície 1
Informação 2 ; Ponto da Superfície 2
      :
      :
Informação N2 ; Ponto da Superfície N2
      :
      :
Bloco M
Número de Pontos no Bloco
Informação 1 ; Ponto da Superfície 1
Informação 2 ; Ponto da Superfície 2
      :
      :
Informação Nm ; Ponto da Superfície Nm
```

Figura C.6. Formato de Arquivo DEF.

A primeira informação contida em um arquivo DEF é também um cabeçalho curto constituído somente por um título de controle. A palavra `_DEF`, que define para o sistema durante a leitura desse arquivo que o arquivo que está sendo lido apresenta o formato DEF.

A segunda informação presente no arquivo refere-se a um dos valores de resolução da malha. Esse valor define quantas linhas da superfície foram digitalizadas e define também quantos blocos de dados estão presentes no arquivo. Cada bloco de dados contém as informações referentes à digitalização de uma linha da superfície. O bloco é composto pela informação do número de pontos existentes nessa linha seguida da listagem de todos esses pontos, cada um sendo precedido por um valor de informação agregado.

Na Figura C.7 é possível observar um exemplo de arquivo DEF. O arquivo ilustrado nessa gravura contém dois blocos, ou seja, foram utilizadas somente duas linhas para a digitalização da superfície. Também é possível observar que o primeiro bloco do arquivo é formado por somente três pontos, enquanto o segundo bloco do arquivo é formado por quatro pontos.

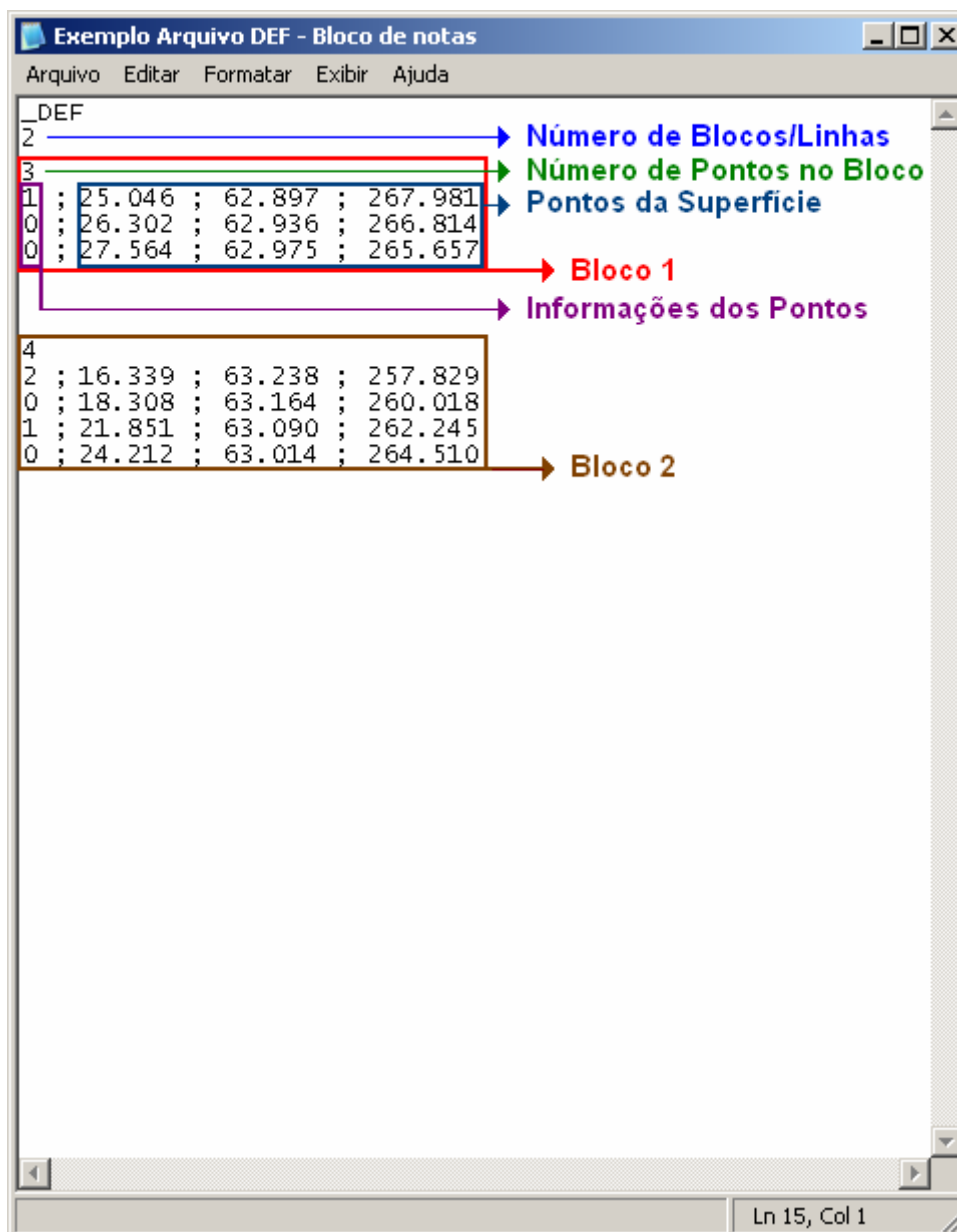


Figura C.7. Exemplo de Arquivo DEF.

D.1 Adição e Subtração

As funções de adição e subtração de imagens não apenas são operações simples, mas também são operações básicas para o processamento de imagens (Ballard e Brown, 1982). O único parâmetro dessas funções é o nome da imagem a ser adicionada ou subtraída à imagem que está sendo processada.

No caso da adição, os valores de cada um dos pixels das imagens é adicionado conforme a Equação (D.1) e, caso o valor dessa soma seja superior ao valor máximo definido pelo número de bits de cada pixel, o valor dessa soma é truncado.

$$I(i, j) = I_1(i, j) + I_2(i, j) \quad (\text{D.1})$$

Na Figura D.1 é possível observar duas imagens e o resultado de sua adição.

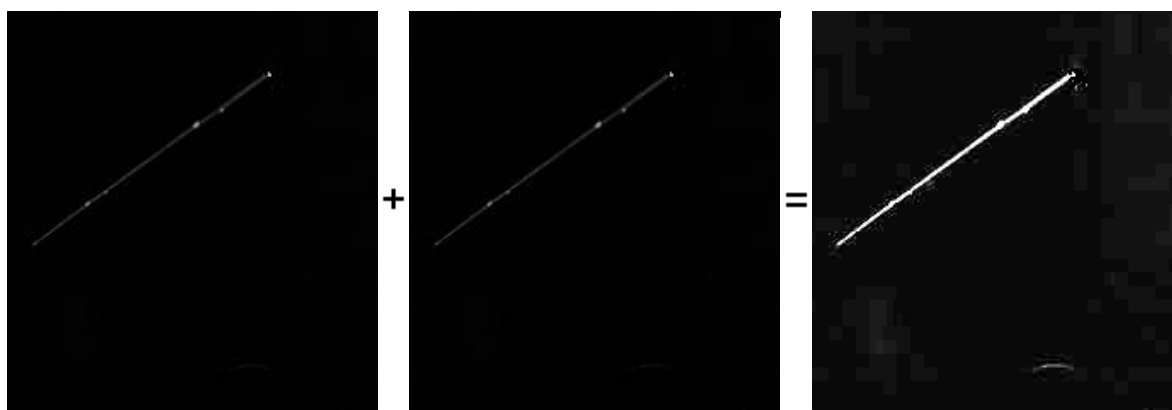


Figura D.1. Exemplo de Adição de Duas Imagens.

Já no caso da subtração, os valores de cada um dos pixels das imagens é subtraído conforme a Equação (D.2) e, caso o valor dessa subtração seja negativo ele é truncado em zero.

$$I(i, j) = I_1(i, j) - I_2(i, j) \quad (\text{D.2})$$

Na Figura D.2 é possível observar duas imagens e o resultado de sua subtração.

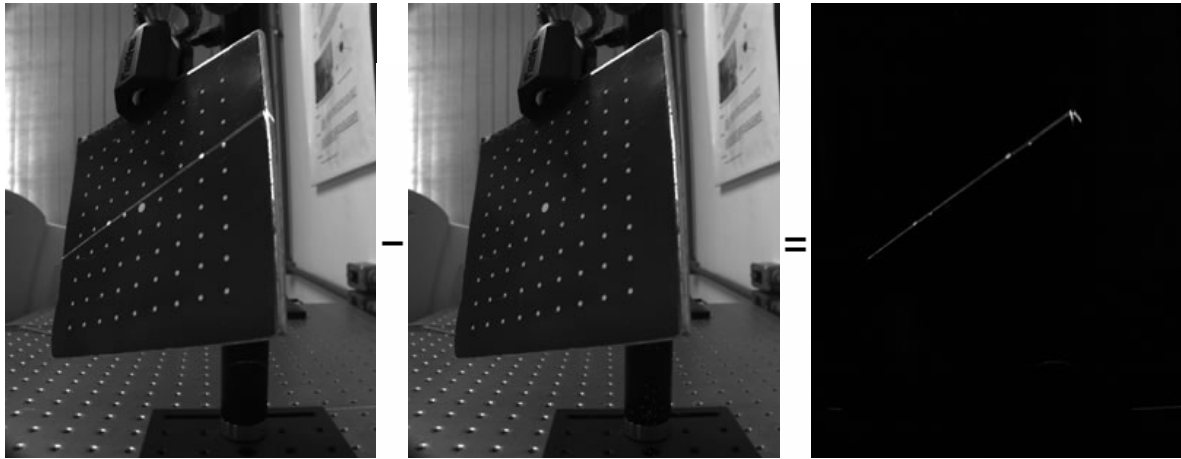


Figura D.2. Exemplo de Subtração de Duas Imagens.

D.2 Mediana

A função de mediana é um filtro passa-baixas de atenuação e é amplamente utilizada no processamento de imagens, principalmente para a atenuação de ruídos (Gonzales e Woods, 2002, Trucco e Verri, 1998, Parker, 1997).

A partir da definição de uma largura de filtro, pode-se compreender a aplicação do filtro de mediana através de uma janela centrada em cada pixel, sendo o novo valor para o pixel a mediana dos valores de pixel contidos dentro dessa janela, conforme ilustrado na Figura D.3, onde é apresentado o cálculo do novo valor para o pixel P_{22} utilizando-se um filtro de mediana com um pixel de largura.

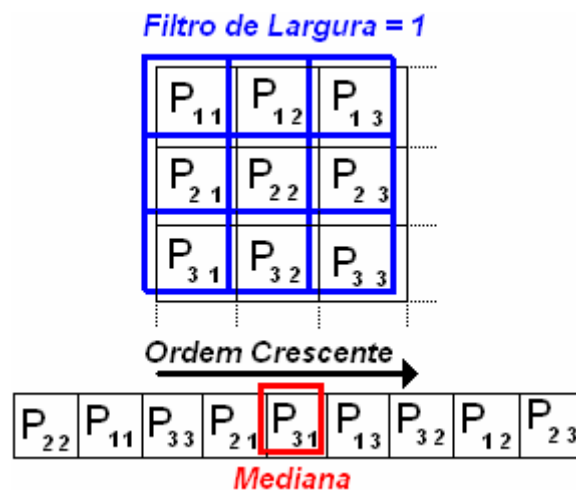


Figura D.3. Exemplo de Cálculo da Mediana de um Pixel de uma Imagem.

Na Figura D.4 é possível observar o efeito de um filtro de mediana com 20 pixels de largura sobre uma imagem de teste.



Figura D.4. Função de Mediana sobre uma Imagem de Teste.

Na aplicação de filtros sobre imagens é observada uma dificuldade quando centra-se as máscaras nas bordas de uma imagem. Existem diferentes possibilidades para a solução desse problema (Gonzalez e Woods, 2002) e aqui se optou pela expansão dos valores das bordas, conforme ilustrado na Figura D.5.

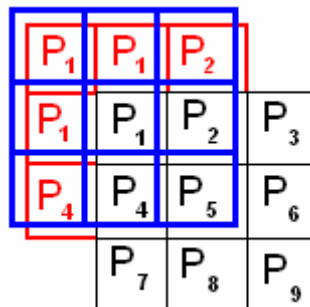


Figura D.5. Expansão das Bordas da Imagem para Aplicação de Filtros.

Nessa figura, os pontos P_1, P_2, \dots, P_9 em preto representam cada um dos pixels da imagem, enquanto a máscara está representada em azul e os pixels “expandidos” estão destacados em vermelho.

D.3 Dilatação e Erosão

A erosão e a dilatação são operadores fundamentais para o processamento morfológico de imagens (Gonzales e Woods, 2002, Parker, 1997).

Esses operadores são normalmente utilizados em imagens binarizadas, onde os pixels só podem assumir dois valores (0 e 1) e sua definição matemática é apresentada detalhadamente por Gonzales e Woods (2002) e Parker (1997), sendo aqui apresentada somente uma interpretação mais intuitiva desses operadores.

A dilatação de uma imagem consiste na definição de uma máscara ou elemento estrutural e na varredura da imagem utilizando esse elemento estrutural. Sempre que um pixel com valor “1” é encontrado na imagem, o elemento estrutural, centrado nesse pixel, é copiado na imagem, conforme pode ser observado na Figura D.6.

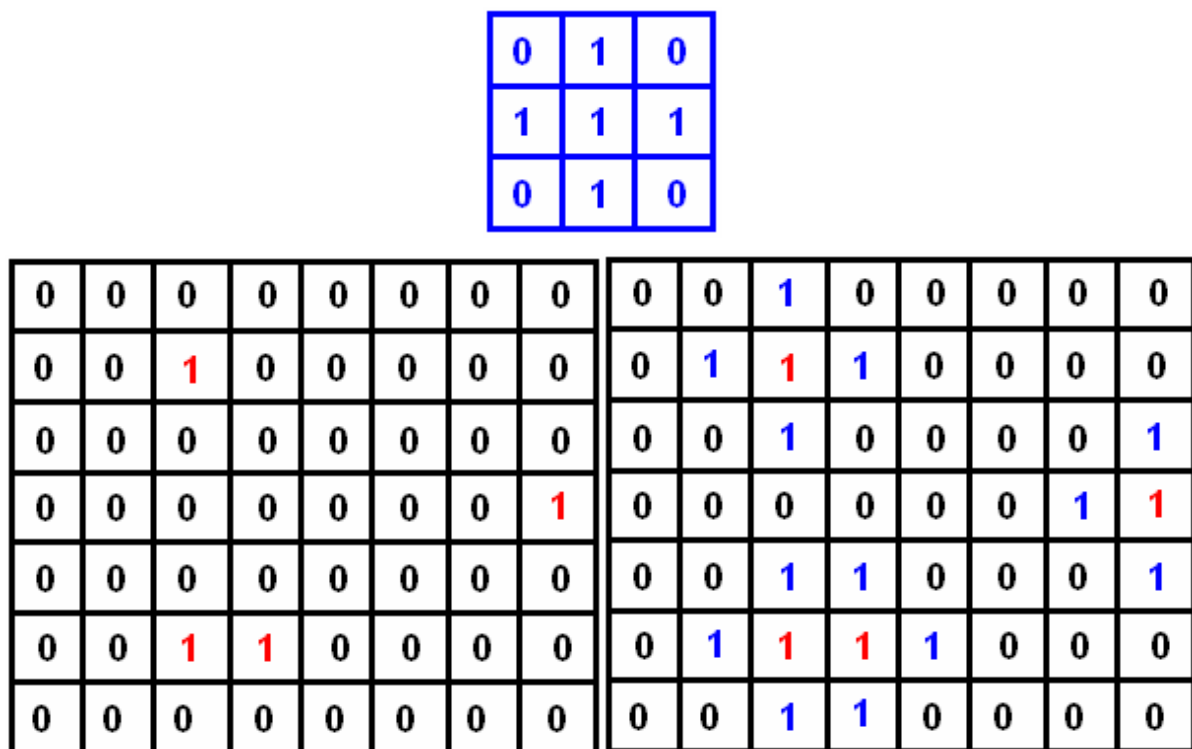


Figura D.6. Elemento Estrutural e Dilatação de uma Imagem.

Em sua implementação no sistema de processamento de imagens do *software* VISSCAN-3D, a operação de dilatação tem seu elemento estrutural definido conforme a Equação (D.3), não sendo possível sua alteração por um usuário do sistema. Na Figura D.7 é possível observar o efeito do operador de dilatação do sistema VISSCAN-3D sobre uma imagem real.

$$E_{Estrutural} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{D.3}$$



Figura D.7. Função de Dilatação sobre uma Imagem de Teste.

Já a erosão, assim como a dilatação, consiste também na definição de um elemento estrutural e na varredura da imagem utilizando esse elemento estrutural. Entretanto é durante essa varredura que a dualidade dessas funções pode ser observada. Durante a varredura é realizada na imagem uma busca pelo padrão definido pelo elemento estrutural, e, sempre que esse padrão é encontrado ele é substituído na imagem final por um único pixel “1”, conforme pode ser observado na Figura D.8. Ou seja, enquanto a dilatação substitui pixels “1” por padrões, a erosão substitui padrões por pixels “1”.

1	1	1
1	1	1
1	1	1

0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0
0	1	1	1	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura D.8. Elemento Estrutural e Erosão de uma Imagem.

Em sua implementação no sistema de processamento de imagens do *software* VISSCAN-3D, a operação de erosão também tem seu elemento estrutural definido conforme a Equação (D.3), não sendo possível sua alteração por um usuário do sistema. Na Figura D.9 é possível observar o efeito desse operador sobre uma imagem real.



Figura D.9. Função de Erosão sobre uma Imagem de Teste.

O único parâmetro que ambas as funções aceitam é o número de vezes que cada uma dessas operações deve ser aplicada sobre a imagem.

D.4 Blobs

O operador de *blobs*, assim como os operadores morfológicos, é utilizado normalmente para o processamento de imagens binarizadas. Esse operador trata a imagem como uma imagem de “0”s com manchas, ou *blobs*, de “1”s. Ele varre a imagem até a localização de um pixel “1”, então, utilizando esse pixel como semente, seleciona toda a região a que esse pixel pertence e calcula sua área em número de pixels. Caso essa área esteja fora dos limites definidos pelo usuário toda a região é removida da imagem, conforme pode ser observado na Figura D.10.

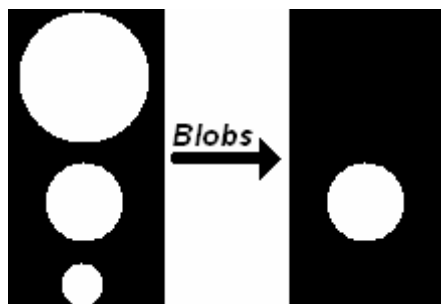


Figura D.10. Seleção e Avaliação de Área em Blobs.

Essa função tem grande aplicação para a filtragem de ruídos e remoção de áreas indesejadas de uma imagem a ser processada. Na Figura D.11 é possível observar o efeito desse operador sobre uma imagem real.

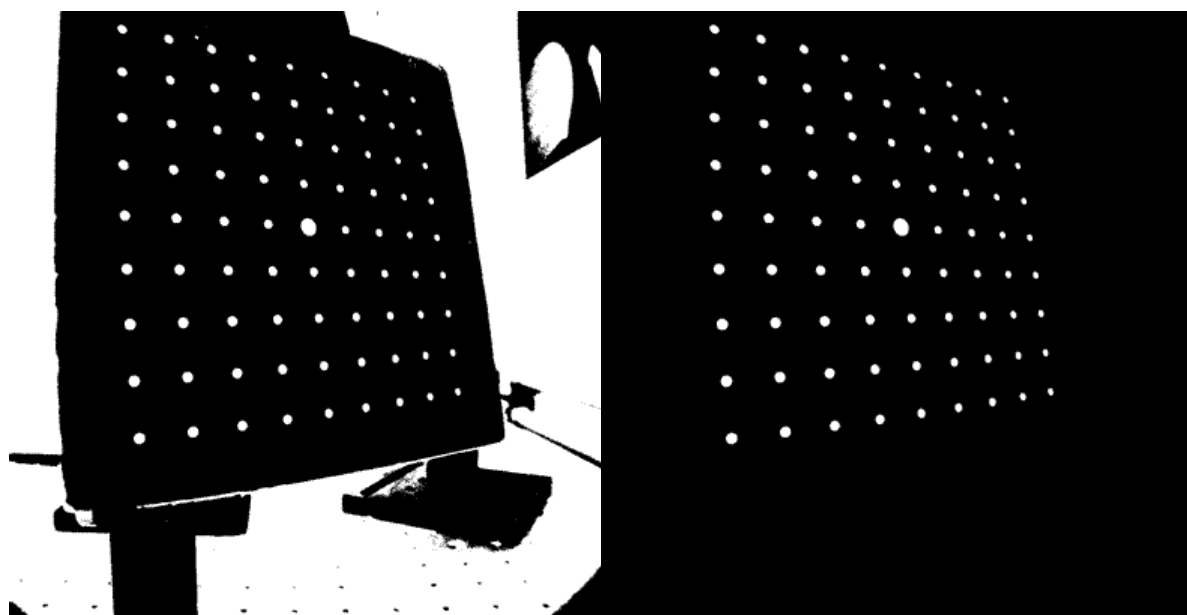


Figura D.11. Função de Blobs sobre uma Imagem de Teste.

D.5 Raio Máximo e Raio Mínimo

Os operador de avaliação de raio, assim como o operador de *blobs*, foi implementado para o processamento de imagens binarizadas. O operador avalia o raio de uma região branca nas direções X e Y da imagem, conforme pode ser observado na Figura D.12.

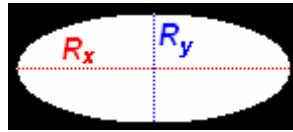


Figura D.12. Raios de um Pixel nas Direções X e Y da Imagem.

Dependendo de qual função foi escolhida pelo usuário, o maior ou o menor desses raios é selecionado e comparado com os valores de limiar. Caso o comprimento do raio escolhido esteja fora dos limites estipulados pelo usuário toda a região é então eliminada da imagem, conforme pode ser observado na Figura D.13 onde essas funções são aplicadas sobre uma imagem de teste.

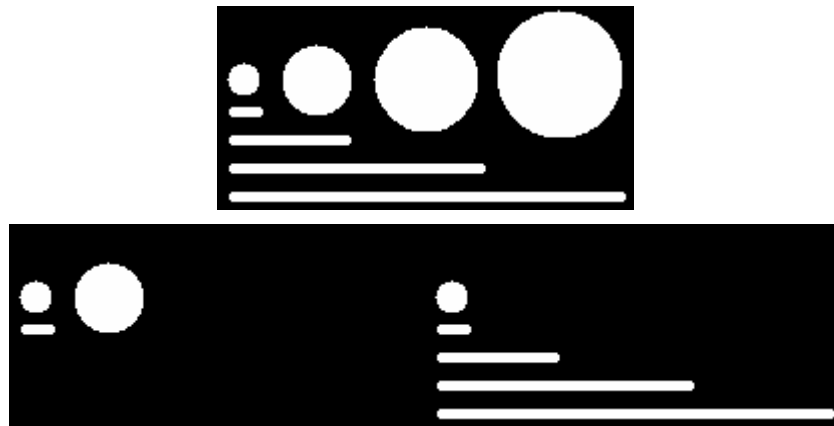


Figura D.13. Função de Raio Máximo e Raio Mínimo sobre uma Imagem de Teste.

D.6 Excentricidade

O operador de excentricidade também utiliza imagens binarizadas e, apesar do nome, não se refere ao parâmetro de excentricidade de cônicas. O operador calcula os raios de uma região branca nas direções X e Y da imagem, conforme pode ser observado na Figura D.12 e calcula a razão entre esses raios conforme a Equação (D.4).

$$\text{Excentricidade} = \frac{\text{Raio}_x}{\text{Raio}_y} \quad (\text{D.4})$$

Em seguida o valor de excentricidade calculado é comparado com a região definida pelos limiares fornecidos pelo usuário e caso esteja fora dessa região, toda a região é removida da imagem conforme pode ser observado na Figura D.14 onde essa função é aplicada sobre uma imagem de teste.

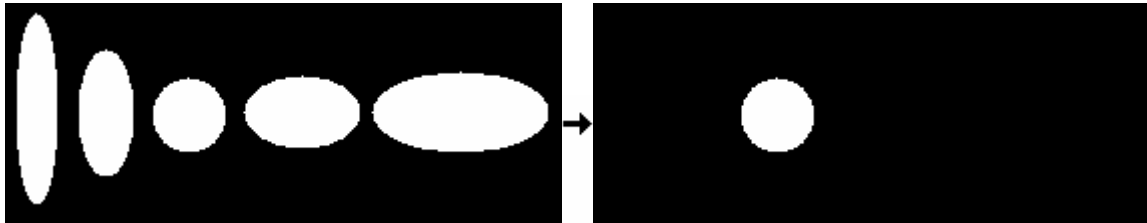


Figura D.14. Função de Excentricidade sobre uma Imagem de Teste.

D.7 Seleção de Região Válida

O operador de seleção de região válida é um operador simples, mas de grande utilidade. Ele permite a definição de uma região de interesse em uma imagem, removendo todo o restante da imagem. Os parâmetros básicos dessa função são o centro da região de interesse e a largura e altura dessa região. Na Figura D.15 é possível observar o efeito desse operador sobre uma imagem real.



Figura D.15. Operador de Seleção de Região Válida Sobre uma Imagem de Teste.

Em sua implementação no sistema de processamento de imagens do *software* VISSCAN-3D, entretanto, existem duas funcionalidades adicionais oferecidas dentro dessa função. A leitura do centro da região a partir de um arquivo de texto e a determinação automática da largura e da altura da região de interesse. Na Tabela D.1 todas as sintaxes possíveis para o uso dessa função são apresentadas.

Tabela D.1. Sintaxes Possíveis para uso da Função de Seleção de Região.

Diferentes Sintaxes da Função de Seleção de Região				
Função	Parâmetros			
REGIAO	<i>Centro X</i>	<i>Centro Y</i>	<i>Largura X</i>	<i>Largura Y</i>
REGIAO	<i>Nome Arquivo</i>		<i>Largura X</i>	<i>Largura Y</i>
REGIAO	<i>Centro X</i>	<i>Centro Y</i>	<i>AUTO</i>	
REGIAO	<i>Nome Arquivo</i>		<i>AUTO</i>	

Essas funcionalidades concedem uma grande adaptabilidade ao operador de seleção de região, entretanto é necessário observar alguns detalhes durante a utilização dessas funcionalidades.

Na leitura do centro da região de interesse a partir de um arquivo é necessário que o arquivo esteja organizado conforme ilustrado na Figura D.16, caso contrário a seleção da região não será realizada.

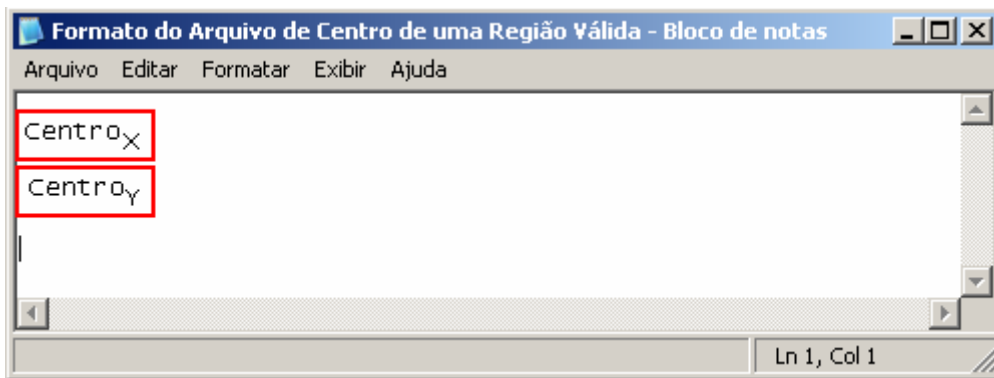


Figura D.16. Formato do Arquivo para Leitura do Centro da Região de Interesse.

Já a determinação automática da região de interesse é uma funcionalidade implementada para operação sobre imagens binarizadas, considerando que os pixels de interesse estejam definidos como “1”s. Uma vez escolhido o centro da região de interesse, a imagem é varrida na direção X e Y em ambos os sentidos, a partir desse centro, até que nenhum pixel “1” seja encontrado. Em seguida, os maiores valores de raios obtidos são selecionados e utilizados como a largura e a altura da região de interesse, conforme ilustrado na Figura D.17 onde o centro da região de interesse é o pixel destacado em azul.



Figura D.17. Varredura da Imagem para Determinação da Região de Interesse.

É importante observar que a detecção automática da região de interesse é somente uma ferramenta para uso da função. Nem sempre a região de interesse é corretamente determinada pelo algoritmo de varredura, como por exemplo na Figura D.17, onde parte da região branca da imagem é perdida com a seleção automática da região. Outra particularidade da varredura ocorre quando o tamanho da região de interesse obtida é inferior a um limite definido internamente pelo sistema, o tamanho da região é fixado como sendo esse tamanho mínimo.

D.8 Centro Geométrico

O operador de cálculo do centro geométrico de uma região, assim como alguns dos operadores apresentados anteriormente, é uma função normalmente utilizada para o processamento de imagens binárias. Ele trata a imagem como uma imagem de “0”s com manchas, ou blobs, de “1”s.

A partir de um ponto inicial definido pelo usuário o sistema calcula o centro geométrico da região especificada pela janela de busca. Esse cálculo é feito iterativamente, até que seja localizado um centro estável, ou seja, definidos o centro inicial e o tamanho de janela busca, essa janela desloca-se iterativamente pela imagem através de cada um dos novos centros calculados até que o centro geométrico calculado e o centro da janela sejam o mesmo conforme ilustrado na Figura D.18.

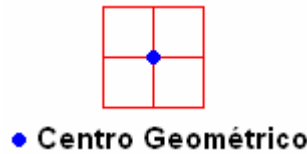


Figura D.18. Janela de Cálculo do Centro Geométrico.

O cálculo do centro geométrico da região definida pela janela de busca Q é realizado através da soma de todos os “1”s da imagem utilizando-se a Equação (D.5)

$$\begin{cases} x_{Centro} = \frac{\sum_Q x_i}{n} \\ y_{Centro} = \frac{\sum_Q y_i}{n} \end{cases}, \quad (D.5)$$

onde n é o número de “1”s dentro da região Q e (x_i, y_i) as coordenadas em pixels de cada um desses “1”s.

Na Figura D.19 é possível observar o cálculo do centro geométrico de uma região onde foram necessárias somente duas iterações.



Figura D.19. Cálculo do Centro Geométrico de uma Imagem de Teste.

A função de cálculo geométrico apresenta ainda algumas funcionalidades para a definição do ponto inicial e do tamanho da janela de busca. Na Tabela D.2 é possível observar as diferentes sintaxes dessa função.

Tabela D.2. Sintaxes Possíveis para uso da Função de Cálculo do Centro Geométrico.

Diferentes Sintaxes da Função de Cálculo do Centro Geométrico			
Função	Parâmetros		
CENTRO	<i>Centro X</i>	<i>Centro Y</i>	<i>Tamanho da Janela</i>
CENTRO	<i>Nome Arquivo</i>		<i>Tamanho da Janela</i>
CENTRO	<i>Centro X</i>	<i>Centro Y</i>	<i>AUTO</i>
CENTRO	<i>Nome Arquivo</i>		<i>AUTO</i>

Cada uma dessas sintaxes funcionam da mesma forma como para a função de seleção de uma região de interesse. O formato do arquivo que contem o ponto inicial a ser utilizado é o mesmo ilustrado na Figura D.16 e para o cálculo automático do tamanho da janela de busca é utilizado o mesmo algoritmo com uma única diferença. Como a janela para o cálculo do centro geométrico utiliza somente um parâmetro para sua descrição, uma vez encontrados os dois raios da região é utilizado somente o maior desses raios, conforme ilustrado na Figura D.20.

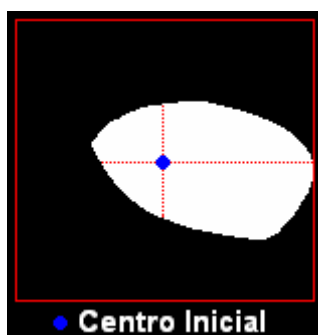


Figura D.20. Determinação da Janela de Cálculo do Centro Geométrico.

D.9 Transformada de Hough

Linhas e curvas são padrões muito importantes em visão computacional e uma das técnicas mais populares para a detecção desses padrões é a transformada de Hough (Trucco e Verri, 1998, Ballard e Brown, 1982).

A transformada de Hough foi apresentada pela primeira vez em 1962 (Hough, 1962) e introduziu uma técnica para a detecção de linhas e curvas a partir de padrões de pontos em imagens binárias. A principal idéia é a transformação de um problema de detecção de padrões em um problema de detecção de máximos (Trucco e Verri, 1998).

A detecção de retas é uma das formas mais simples da transformada de Hough (Ballard e Brown, 1982) e pode vista como uma transformação de coordenadas realizada em duas etapas (Trucco e Verri, 1998). Inicialmente transforma-se o problema de detecção de retas em um problema de detecção de cruzamentos e, em seguida, transforma-se o problema da detecção de cruzamentos em um problema de detecção de máximos.

Na primeira etapa da transformação de Hough, dada uma linha definida como:

$$y = mx + n, \tag{D.6}$$

sabe-se que essa reta é identificada unicamente pelos parâmetros (m, n) . Ou seja, essa reta é representada por um único ponto em um espaço de dados formado por esses parâmetros, conforme pode ser observado na Figura D.21.

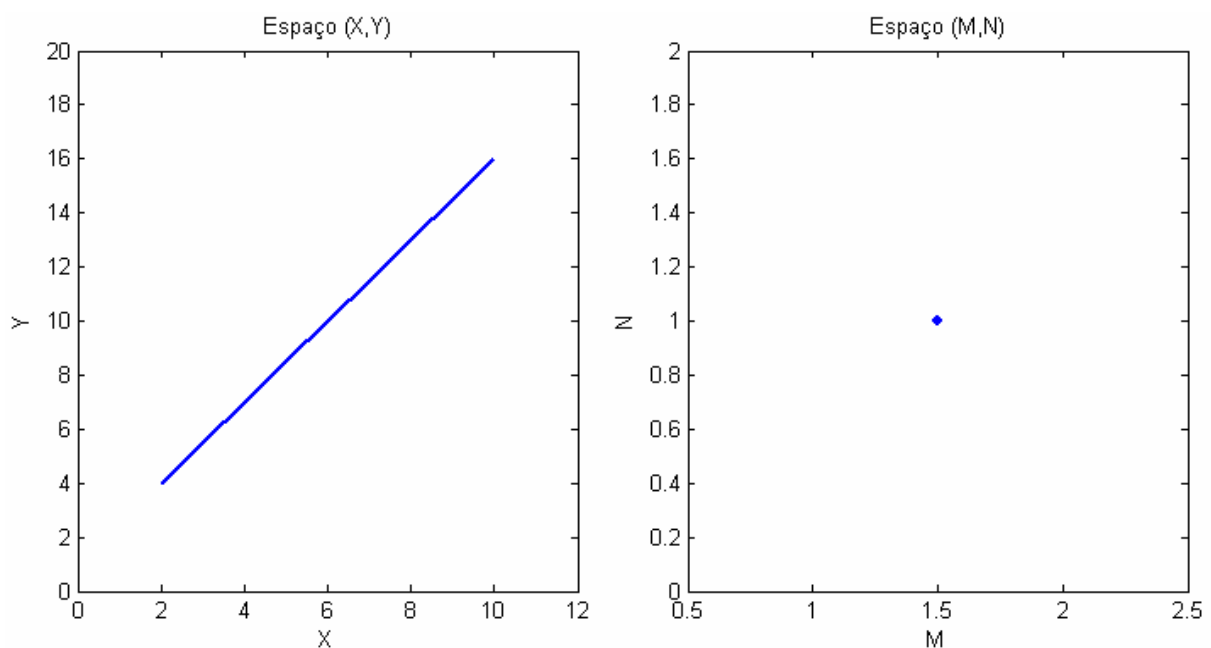


Figura D.21. Transformação de Coordenadas sobre uma Reta.

Já um ponto $P = (x, y)$ em uma imagem, conforme pode ser observado na Equação (4.39) e na Figura D.22, corresponderá a uma reta no espaço de parâmetros (M, N) que representa todas as retas no espaço (X, Y) que passam por esse ponto.

$$n = -xm + y \tag{D.7}$$

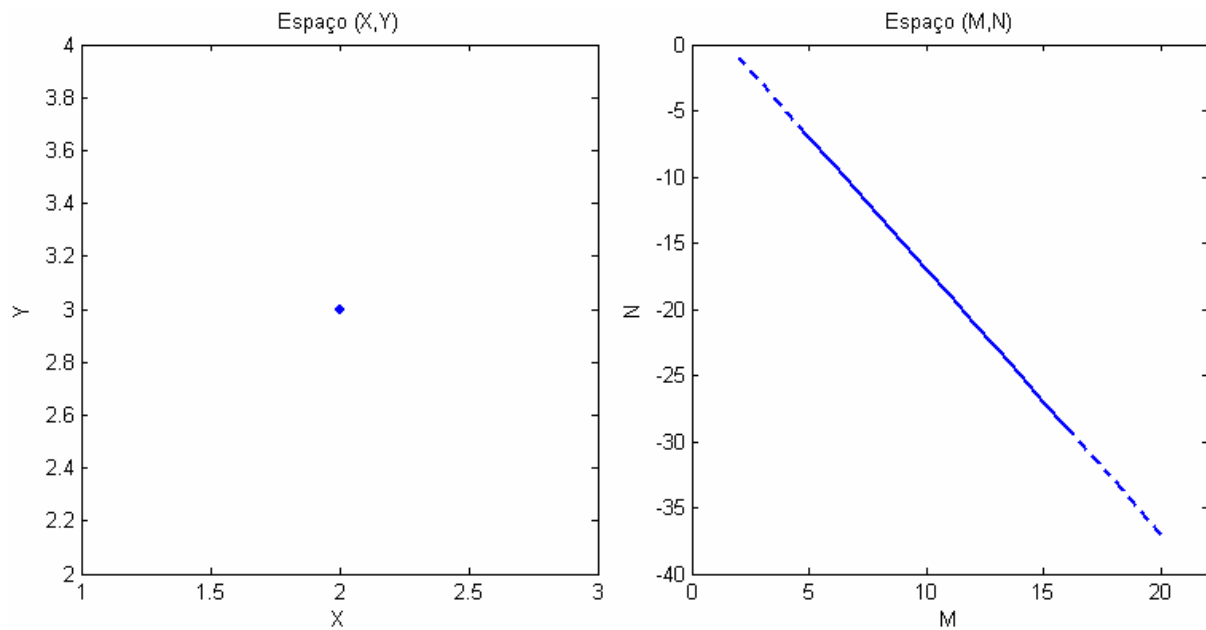


Figura D.22. Transformação de Coordenadas sobre um Ponto.

Logo, sendo uma reta no espaço (X,Y) definida por um série de pontos P_1, P_2, \dots, P_i colineares, essa reta será identificada no espaço (M,N) pela interseção das retas associadas a cada um dos pontos P_1, P_2, \dots, P_i , conforme ilustrado na Figura D.23.

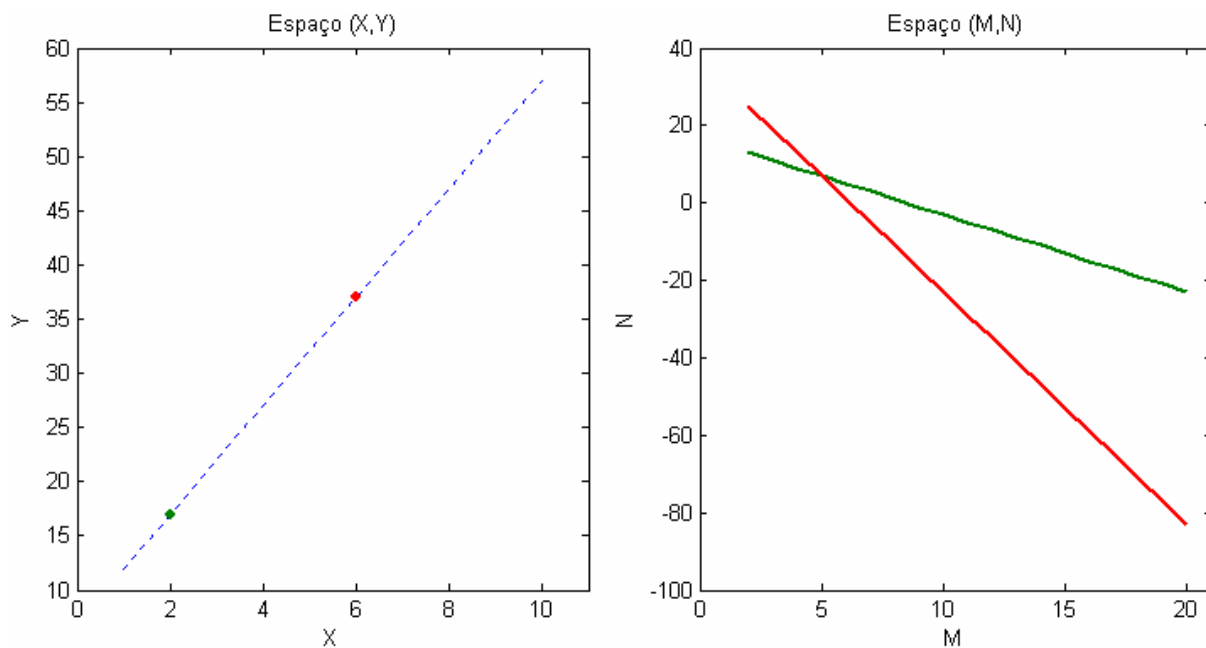


Figura D.23. Detecção de Retas Transformada em Detecção de Cruzamento.

Dessa forma o problema de detecção de retas em (X,Y) transforma-se em um problema de detecção de cruzamentos em (M,N) .

Agora, discretizando o espaço (M, N) com a resolução desejada, é possível associar um acumulador $C(M, N)$ a cada um dos valores possíveis de M e N . Logo, para cada ponto P no espaço (X, Y) , todos os acumuladores correspondentes a reta no espaço (M, N) associada a esse ponto são incrementados, de forma que após todos os pontos do espaço (X, Y) serem processados, os parâmetros m e n da reta no espaço (X, Y) da imagem serão dados pelo acumulador que apresentar o máximo valor, conforme pode ser observado na Figura D.24.

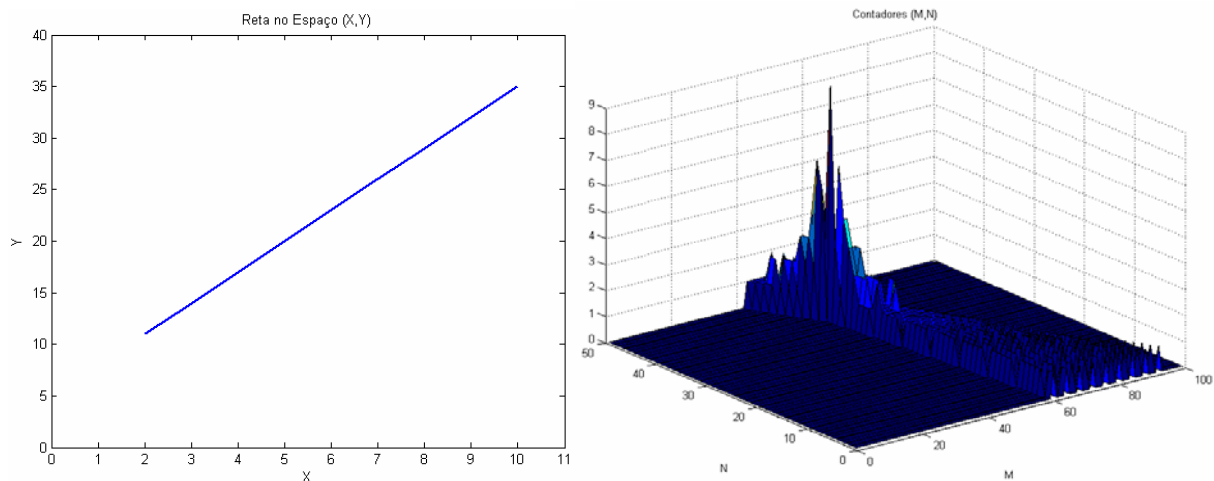


Figura D.24. Reta no Espaço (X, Y) e Acumuladores Resultantes no Espaço (M, N) .

Entretanto, uma das dificuldades desse método é o fato de que m e n podem assumir quaisquer valores dentro do intervalo $]-\infty, \infty[$, o que implica que não é possível considerar todo o espaço (M, N) e nem representar um reta da forma $x = k$, onde k é uma constante (Trucco e Verri, 1998, Ballard e Brown, 1982). A forma de se superar essa dificuldade é o uso da representação polar para a reta.

$$\rho = x \cos(\theta) + y \sin(\theta) \quad (\text{D.8})$$

, onde ρ é a distância entre a reta e a origem da imagem e θ a orientação dessa reta.

Conhecendo-se a transformada de Hough para a detecção de retas é possível generalizar essa transformação para a detecção de curvas (Trucco e Verri, 1998, Ballard e Brown, 1982). Dada uma curva com a forma:

$$y = f(x, \mathbf{A}) \quad (\text{D.9})$$

, onde $\mathbf{A} = [a_1, a_2, \dots, a_j]$ é um vetor contendo todos os parâmetros dessa curva, basta realizar a discretização do espaço paramétrico e para cada ponto $P = (x, y)$ da imagem incrementar todos os contadores definidos por essa curva e, finalmente, detectar o máximo no espaço paramétrico.

Dentre as funcionalidades de processamento de imagens oferecidas pelo sistema VISSCAN-3D, está incluída a transformada de Hough para a detecção de retas. Cada um dos parâmetros de controle do algoritmo estão ilustrados na Tabela D.3.

Tabela D.3. Parâmetros da Função de Detecção de Retas com Transformada de Hough.

Parâmetros para a Transformada de Hough	
Parâmetro	Descrição
N_{Retas}	<i>Número de Retas</i>
R_ρ	<i>Resolução do Módulo das Retas</i>
R_θ	<i>Resolução da Inclinação das Retas</i>
V_θ	<i>Região de Supressão de Máximos</i>

Os parâmetros de resolução R_ρ e R_θ são utilizados para a discretização do espaço paramétrico e N_{Retas} fornece o número de retas a serem detectadas. Já o parâmetro V_θ realiza uma otimização de implementação, eliminando a dupla identificação de uma mesma reta, definindo uma região nula em torno das linhas detectadas conforme ilustrado na Figura D.26.

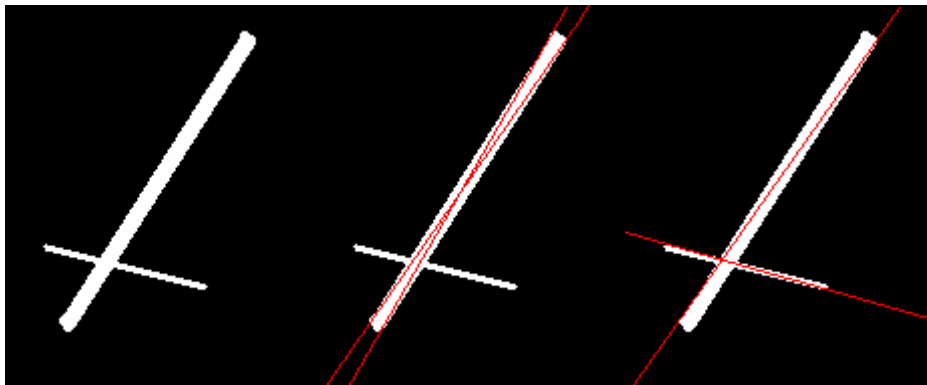


Figura D.25. Vizinhaça Nula e Efeito sobre a Detecção de Retas.

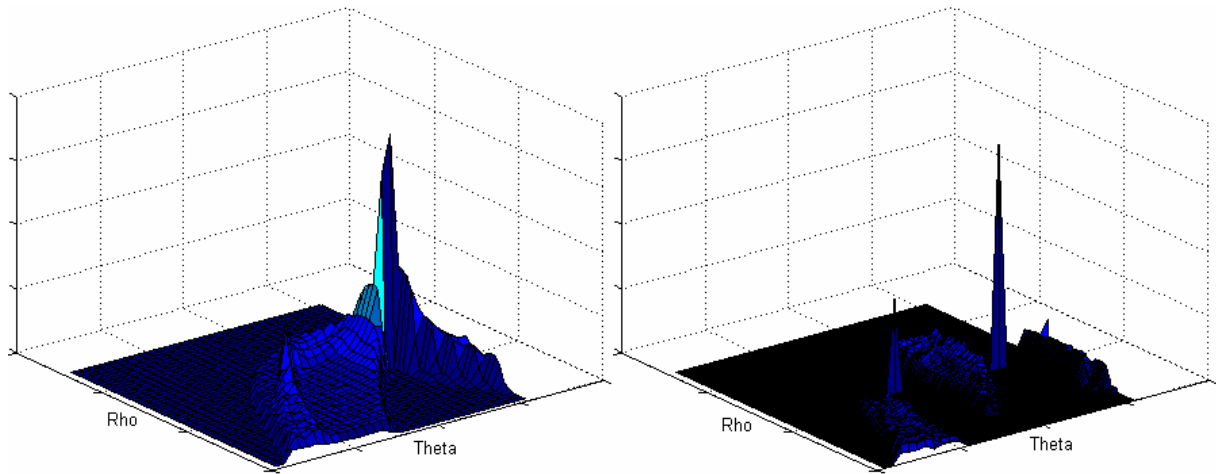


Figura D.26. Vizinhança Nula sobre os Acumuladores da Transformada de Hough.

A função gera como saída um arquivo de dados onde os parâmetros das retas detectadas são salvos seguindo o padrão ilustrado na Figura D.27.

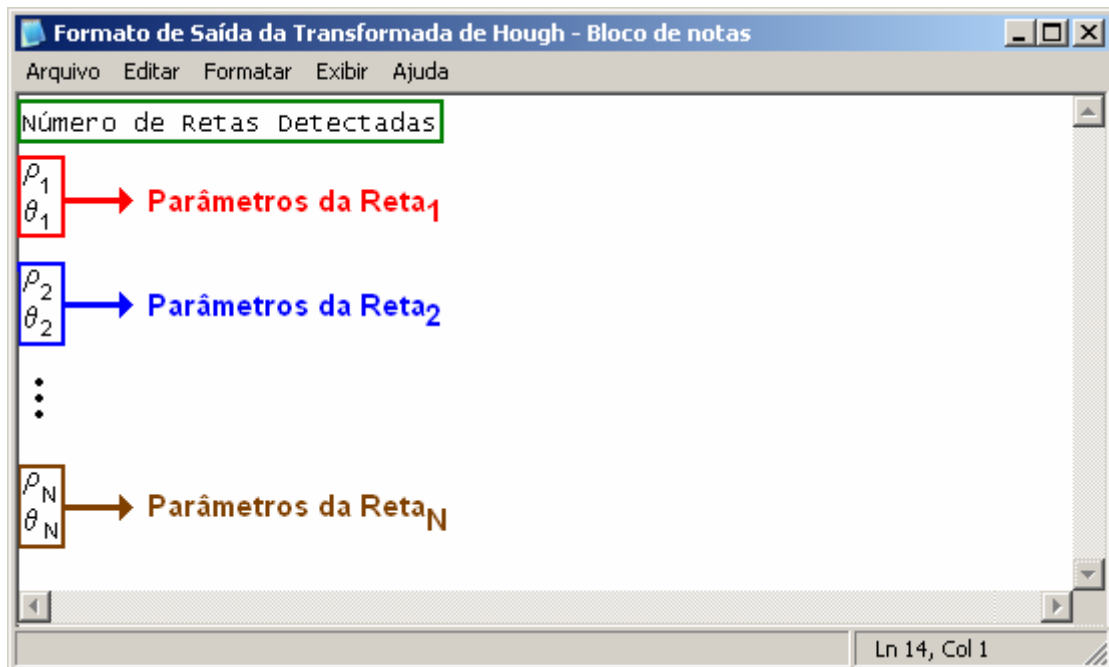


Figura D.27. Arquivo de Saída da Detecção de Retas com a Transformada de Hough.

D.10 Interseção

A função de interseção disponibilizada pelo sistema de processamento de imagens consiste, basicamente, numa aplicação da transformada de Hough. Dadas duas retas, cujas equações são dadas na forma polar conforme ilustrado na Equação (D.10), essa função implementa a detecção do ponto de cruzamento dessas retas.

$$\begin{cases} \rho_1 = x \cos(\theta_1) + y \operatorname{sen}(\theta_1) \\ \rho_2 = x \cos(\theta_2) + y \operatorname{sen}(\theta_2) \end{cases} \quad (\text{D.10})$$

Existem diferentes formas de se realizar o cálculo desse ponto de cruzamento (Edwards e Penney, 1998, Swokowski, 1995), a implementação aqui apresentada baseia-se na solução do sistema ilustrado na Equação (D.11) através da regra de Crammer.

$$\begin{cases} \rho_1 = x_{\text{Interseção}} \cos(\theta_1) + y_{\text{Interseção}} \operatorname{sen}(\theta_1) \\ \rho_2 = x_{\text{Interseção}} \cos(\theta_2) + y_{\text{Interseção}} \operatorname{sen}(\theta_2) \end{cases} \Rightarrow \begin{bmatrix} \cos(\theta_1) & \operatorname{sen}(\theta_1) \\ \cos(\theta_2) & \operatorname{sen}(\theta_2) \end{bmatrix} \begin{bmatrix} x_{\text{Interseção}} \\ y_{\text{Interseção}} \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} \quad (\text{D.11})$$

Logo, a solução do sistema será:

$$\begin{cases} x_{\text{Interseção}} = \frac{\rho_2 \operatorname{sen}(\theta_1) - \rho_1 \operatorname{sen}(\theta_2)}{\operatorname{sen}(\theta_1) \cos(\theta_2) - \operatorname{sen}(\theta_2) \cos(\theta_1)} \\ y_{\text{Interseção}} = \frac{\rho_1 \cos(\theta_2) - \rho_2 \cos(\theta_1)}{\operatorname{sen}(\theta_1) \cos(\theta_2) - \operatorname{sen}(\theta_2) \cos(\theta_1)} \end{cases} \quad (\text{D.12})$$

Optou-se pelo uso da regra de Crammer pois através do determinante principal do sistema é possível determinar se existe realmente um ponto de interseção das retas e se essas são paralelas, concorrentes ou coincidentes.

Entretanto, antes da detecção do ponto de interseção, é necessária a detecção das retas que geram essa interseção. Para a detecção dessas retas é utilizada a transformada de Hough apresentada no Apêndice D. São detectadas as duas retas mais prováveis da imagem, utilizando-se para a transformada de Hough os parâmetros definidos pelo usuário quando da chamada da função de interseção e em seguida o ponto de interseção é calculado conforme a Equação (D.12) e salvo em um arquivo de texto escolhido pelo usuário através do parâmetro *Nome Arquivo*. O formato desse arquivo de saída pode ser observado na Figura D.28 e na Figura D.29 tem-se um exemplo da detecção de interseção de retas sobre uma imagem de teste.

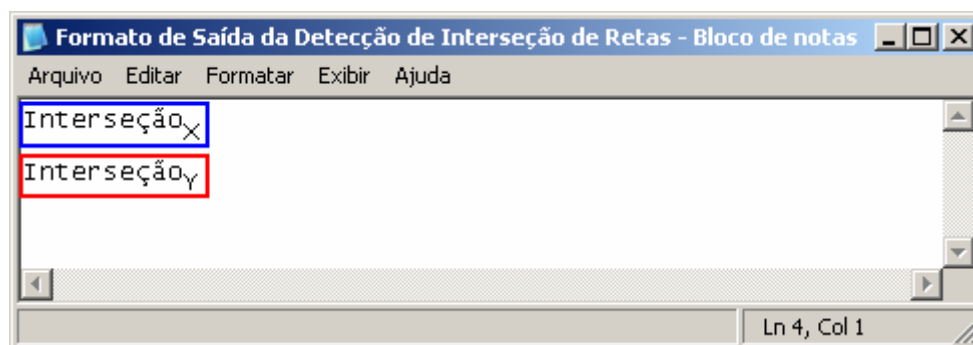


Figura D.28. Arquivo de Saída da Detecção de Interseção de Retas.

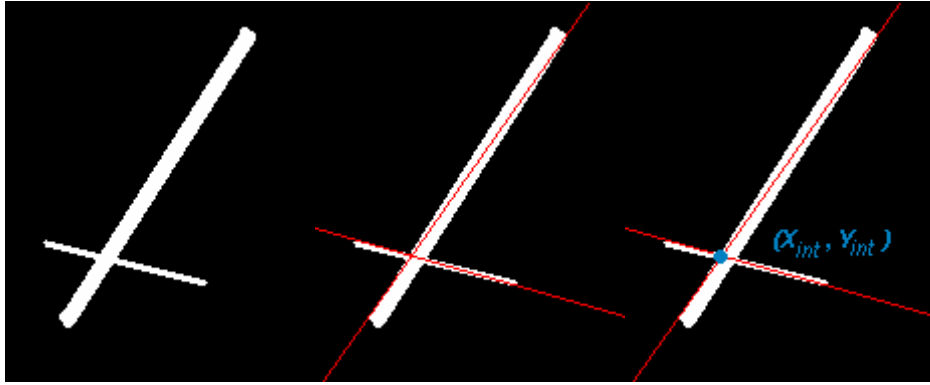


Figura D.29. Exemplo de Detecção de Interseção de Retas.

D.11 Vértices

A descrição geométrica de um vértice em uma imagem não é algo tão simples (Trucco e Verri, 1998). Um vértice pode ser definido como a interseção de duas bordas, ou ainda como um ponto que apresenta duas direções de borda dominantes. Ou seja, um vértice em uma imagem pode não apenas representar um vértice geométrico, mas também o final de uma linha, ou um pixel isolado que apresente um máximo ou mínimo local de intensidade, ou muitas outras características. São esses pontos que são detectados por um detector de vértices.

Existe uma grande variedade de detectores de vértices (Gonzales e Woods, 2002, Trucco e Verri, 1998). O sistema VISSCAN-3D disponibiliza um detector de vértices apresentado detalhadamente por Trucco e Verri (1998) baseado no detector de Tomasi e Kanade (1991).

Considerando-se um ponto qualquer P pertencente a uma imagem I e uma vizinhança Q desse ponto conforme ilustrado na Figura D.30, é possível, a partir do gradiente da imagem, definir-se uma matriz C que descreve as variações de intensidade da imagem dentro dessa vizinhança.

Vizinhança de Largura = 1

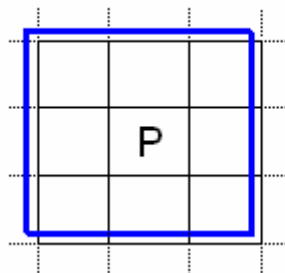


Figura D.30. Vizinhança de um Ponto em uma Imagem.

$$C = \begin{bmatrix} \sum_{\varrho} \left(\frac{\partial I}{\partial x} \right)^2 & \sum_{\varrho} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \sum_{\varrho} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \sum_{\varrho} \left(\frac{\partial I}{\partial y} \right)^2 \end{bmatrix}, \quad (\text{D.13})$$

onde as derivadas da imagem são obtidas numericamente conforme apresentado no Apêndice A.

Uma vez que a matriz C é uma matriz simétrica, é possível através de uma rotação de eixos reescrever essa matriz como uma matriz diagonal.

$$C = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, \quad (\text{D.14})$$

onde λ_1 e λ_2 são os autovalores da matriz C e definem a força das bordas nas direções definidas pelos autovetores da matriz.

Logo, assumindo-se que $\lambda_1 \geq \lambda_2$, um vértice será um ponto onde a força da borda mais fraca (λ_2) é superior a um limite.

Ou seja, para a detecção de vértices, para cada pixel da imagem calcula-se o valor de λ_2 e compara-se esse valor com um limite definido pelo usuário através da especificação de um fator de qualidade τ conforme a Equação (D.15).

$$I(x, y) = \text{Vértice}, \lambda_2 \geq \tau \max(\lambda_2) \quad (\text{D.15})$$

Selecionando-se então somente os pontos que apresentam um valor elevado de λ_2 constrói-se uma lista de vértices da imagem. Em seguida, o sistema de processamento realiza ainda uma segunda filtragem dessa lista utilizando um parâmetro para a definição de uma distância mínima entre os vértices. Iniciando-se com os vértices que possuem valores mais elevados de λ_2 , a distância entre esse vértice e todos os outros vértices é calculada e, caso essa distância seja inferior a distância mínima definida pelo usuário, os vértices demasiadamente próximos são removidos da lista de vértices, conforme pode ser observado na Figura D.31.

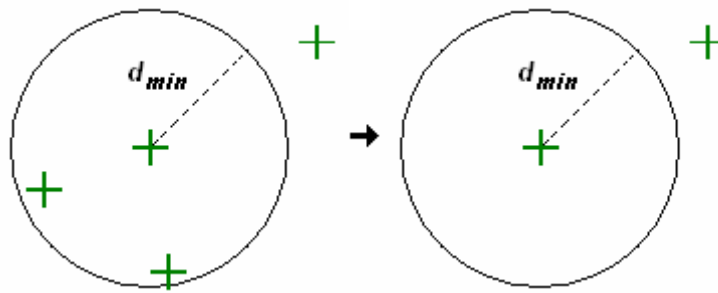


Figura D.31. Remoção de Vértices Utilizando o Critério de Distância Mínima.

Ao final do processamento da imagem, é fornecida uma lista de vértices que segue o formato descrito na Figura D.32.

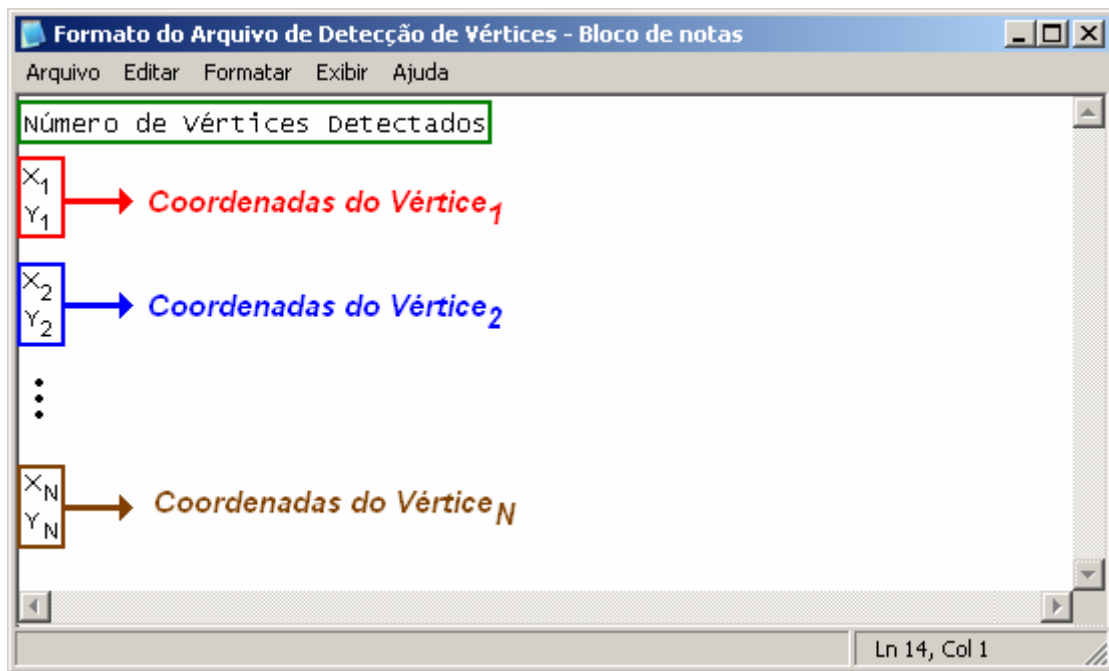


Figura D.32. Formato do Arquivo de Saída da Detecção de Vértices.

Na Figura D.33 é possível observar um exemplo da detecção de vértices sobre uma imagem de teste.

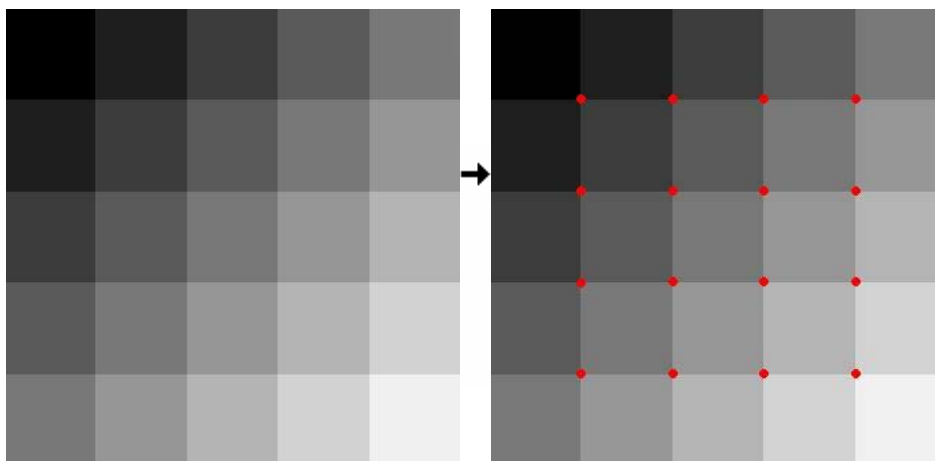


Figura D.33. Detecção de Vértices de uma Imagem de Teste.

E.1 Funções de Processamento

E.1.1 Saturação

Essa função consiste na aplicação de limites sobre a profundidade da nuvem de pontos, ou seja, ela define uma profundidade máxima e uma profundidade mínima para os pontos da superfície, e caso um ponto possua uma profundidade fora desses limites trunca os valores de profundidade desse ponto para dentro desses limites. Na Figura E.1 é possível observar o perfil de uma superfície, onde o efeito da aplicação dessa função fica evidente.

Já na Figura E.2 observa-se a aplicação da função de saturação sobre uma superfícies mais complexa.

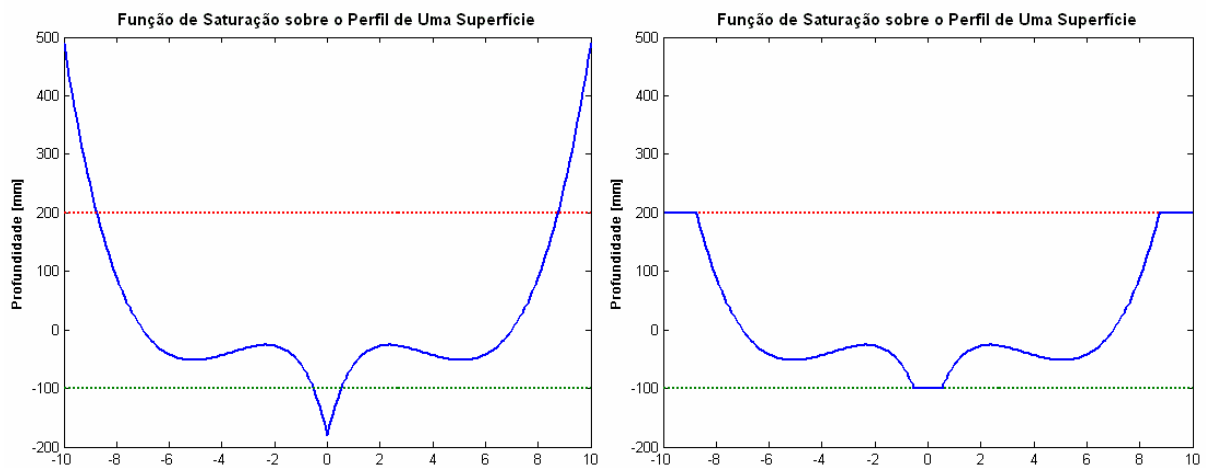


Figura E.1. Função de Saturação sobre o Perfil de um Superfície.

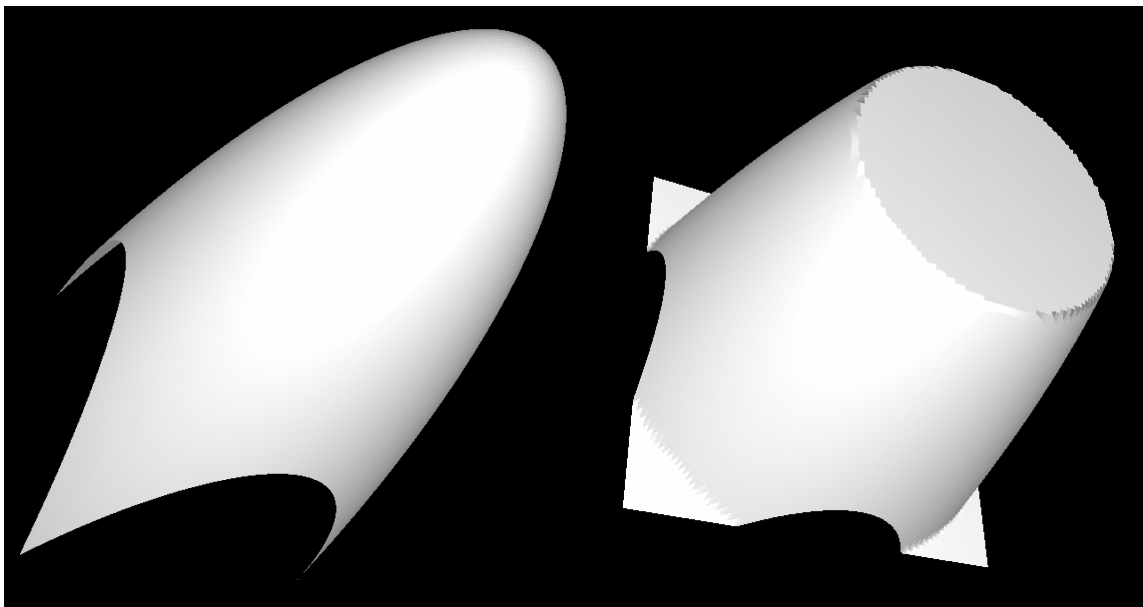


Figura E.2. Função de Saturação sobre uma Superfície de Teste.

E.1.2 Média

Essa função consiste na aplicação de um filtro de média sobre a nuvem. Uma vez definida a largura do filtro, uma máscara de média é aplicada sobre cada um dos pontos e seu novo valor de profundidade é calculado conforme a Equação (E.1).

$$Z = \frac{\sum_R Z_i}{n} \quad (\text{E.1})$$

, onde R é o conjunto de pontos que se encontram dentro da região da máscara do filtro e n o número de pontos desse conjunto.

Assim como na aplicação de filtros sobre imagens, a aplicação de filtros sobre uma nuvem de pontos também apresenta problemas quando aplica-se esse filtro sobre as bordas da nuvem de pontos. A solução escolhida para superar essa dificuldade é a mesma utilizada com as imagens, a expansão dos valores das bordas, conforme ilustrado na Figura D.5, considerando-se que os pontos P_1, P_2, \dots, P_9 em preto representam a nuvem de pontos, enquanto a máscara está representada em azul e os pontos “expandidos” estão destacados em vermelho.

Na Figura E.3 é possível observar o efeito desse filtro sobre uma superfície.

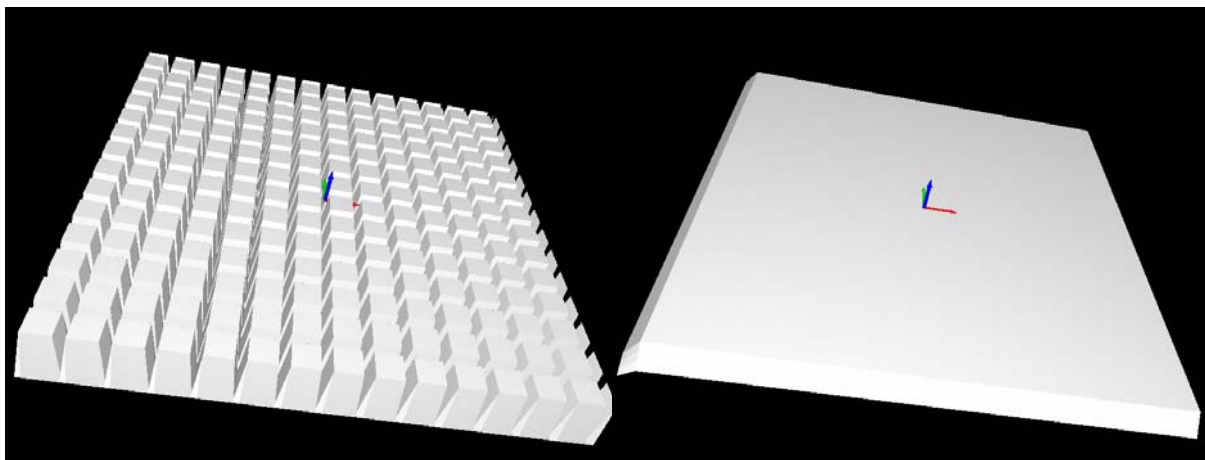


Figura E.3. Função de Média sobre uma Superfície de Teste.

E.1.3 Mediana

Essa função, assim como acontece com a função média, consiste na aplicação de um filtro sobre a nuvem. Uma vez definida a largura do filtro, uma máscara de mediana é aplicada sobre cada um dos pontos e seu novo valor de profundidade é obtido.

Para a obtenção da mediana de um ponto da nuvem, todos os pontos dentro do filtro são ordenados em ordem crescente de profundidade e o ponto médio desse vetor é utilizado como a nova profundidade. Na Figura E.4 pode-se observar o efeito do filtro de mediana sobre uma nuvem de pontos.

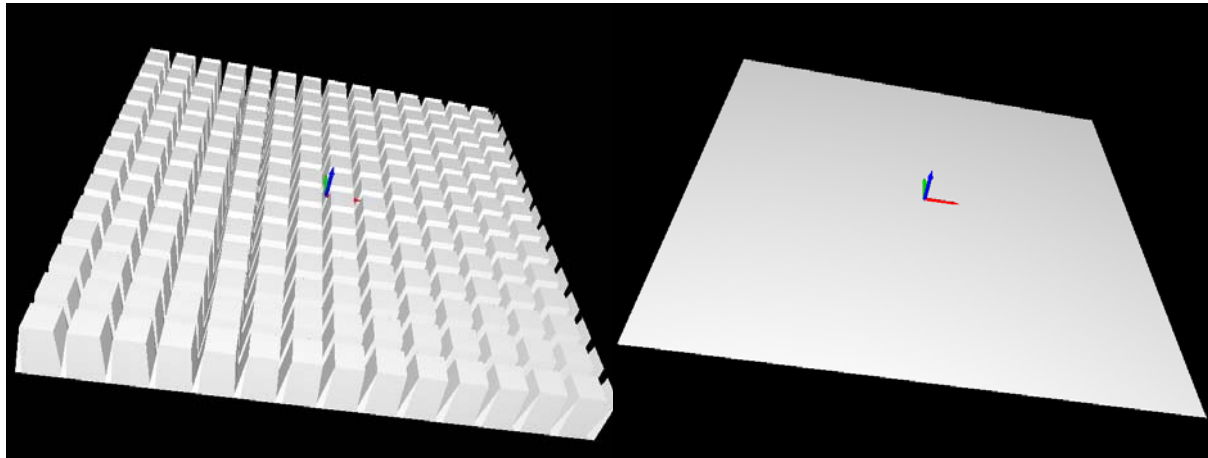


Figura E.4. Função de Mediana sobre uma Superfície de Teste.

No filtro de mediana observa-se o mesmo problema nas bordas que o observado no filtro de média e a solução implementada utiliza a mesma idéia que a implementada no filtro de média.

E.2 Funções de Detecção

E.2.1 Limiar

Essa função consiste na avaliação dos valores de profundidade de cada um dos pontos que fazem parte da nuvem. Caso a profundidade de um ponto esteja fora dos limites superior ou inferior definidos pelo usuário, esse ponto será detectado e marcado. Na Figura E.5 observa-se a aplicação desse detector sobre uma superfície digitalizada.

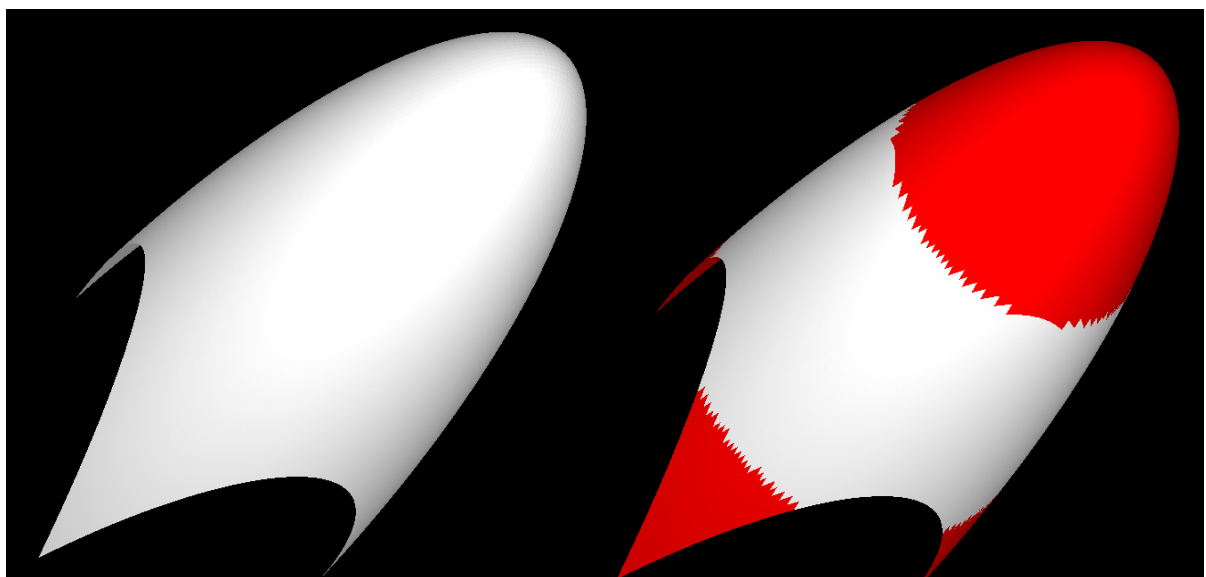


Figura E.5. Função de Limiar sobre uma Superfície de Teste.

E.2.2 Gradiente

Essa função consiste na avaliação dos valores de gradiente de cada um dos pontos que fazem parte da nuvem. Inicialmente o gradiente de cada um dos pontos é calculado conforme a Equação (E.2).

$$GRAD = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2} \quad (\text{E.2})$$

Em seguida, caso o gradiente de um ponto esteja acima do limiar definido pelo usuário, esse ponto será detectado e marcado. Na Figura E.6 observa-se a aplicação desse detector sobre uma superfície digitalizada.

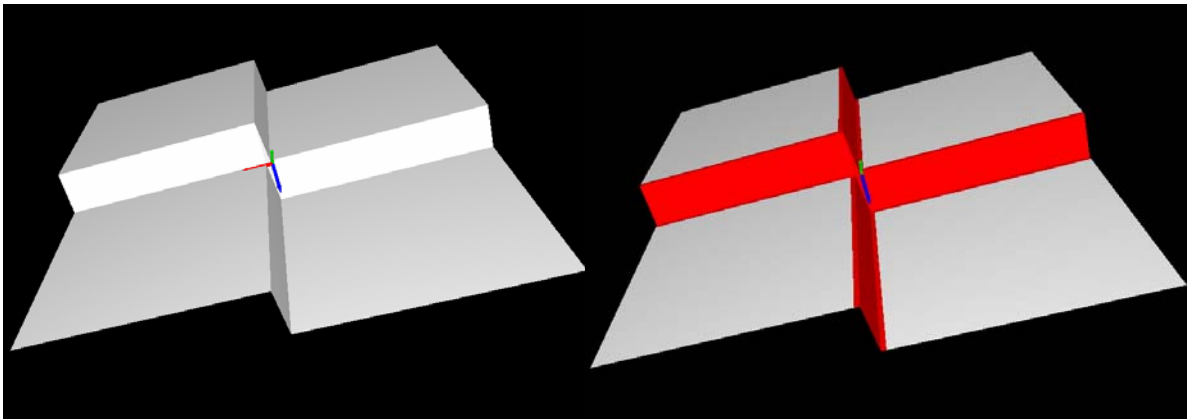


Figura E.6. Função de Gradiente sobre uma Superfície de Teste.

E.2.3 Laplaciano

Essa função consiste na avaliação dos valores do Laplaciano de cada um dos pontos que fazem parte da nuvem. Inicialmente o Laplaciano de cada um dos pontos é calculado conforme a Equação (E.3).

$$LAPLACIANO = \left(\frac{\partial^2 z}{\partial x^2}\right) + \left(\frac{\partial^2 z}{\partial y^2}\right) \quad (\text{E.3})$$

Em seguida, caso o Laplaciano de um ponto esteja acima do limiar definido pelo usuário, esse ponto será detectado e marcado. Na Figura E.7 observa-se a aplicação desse detector sobre uma superfície digitalizada.

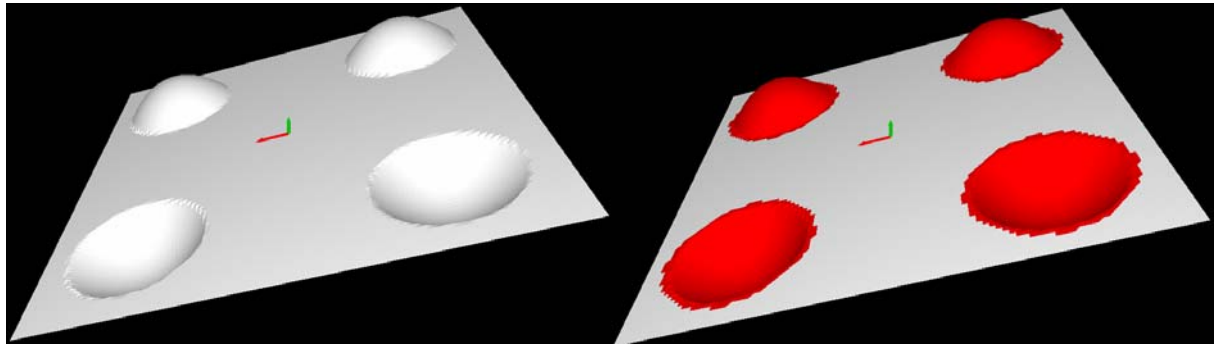


Figura E.7. Função de Laplaciano sobre uma Superfície de Teste.

E.2.4 Curvatura

A curvatura é uma propriedade amplamente utilizada para a identificação e caracterização de objetos em visão computacional e existe uma grande variedade de técnicas para a estimação da curvatura de uma superfície qualquer dado um mapa de profundidade (Davis *et.al.*, 1995).

Uma dessas técnicas que apresenta bons resultados e é utilizada freqüentemente na descrição de superfícies é a técnica de segmentação HK. Essa técnica particiona uma superfície em regiões que apresentam uma forma homogênea e baseia-se na avaliação da curvatura gaussiana (K) e da curvatura média (H) de uma superfície conforme a Figura E.8 (Cantzler e Fisher , 2001, Trucco e Verri, 1998, Davis *et.al.*, 1995).

	K -	0	+
H -			
0			Resultado Impossível
+			
	Hiperbólico	Cilindrico Côncavo	Elíptico Côncavo

Figura E.8. Classificação de Superfícies HK (Trucco e Verri, 1998).

A técnica de segmentação HK utiliza somente os sinais das curvaturas, o que concede uma robustez significativa à técnica, uma vez que, mesmo que a magnitude da curvatura possa ser influenciada pela presença de ruídos na digitalização, normalmente o sinal desta pode ser estimado corretamente (Trucco e Verri, 1998).

O cálculo dos valores da curvatura gaussiana e média de uma superfície digitalizada pode ser feito a partir da estimação das derivadas parciais dessa superfície. Sabendo que, dada as curvaturas principais P_1 e P_2 de uma superfície qualquer, as curvaturas gaussiana e média são dadas respectivamente por (Cantzler e Fisher, 2001, Trucco e Verri, 1998, Davis *et.al.*, 1995, Swokowski, 1995):

$$K = p_1 p_2 \quad (\text{E.4})$$

$$H = \frac{1}{2}(p_1 + p_2) \quad (\text{E.5})$$

É possível reescrever essas equações em função somente das derivadas parciais da superfície:

$$K = \frac{S_{xx}S_{yy} - S_{xy}^2}{(1 + S_x^2 + S_y^2)^2} \quad (\text{E.6})$$

$$H = \frac{1}{2} \frac{(1 + S_x^2)S_{yy} - 2S_x S_y S_{xy} + (1 + S_y^2)S_{xx}}{\sqrt{(1 + S_x^2 + S_y^2)^3}} \quad (\text{E.7})$$

A função de curvatura disponibilizada pelo sistema de processamento de malhas do *software* VISSCAN-3D oferece três opções de uso, cujas sintaxes estão definidas de acordo com a Tabela E.1.

Tabela E.1. Diferentes Sintaxes da Função de Curvatura.

Operação	Sintaxe
Segmentação HK	CURVATURA HK
Limiar H	CURVATURA H <i>Limiar</i>
Limiar K	CURVATURA K <i>Limiar</i>

Ela é capaz de realizar a segmentação HK, retornando uma superfície onde cada uma das formas ilustradas na Figura E.8 é destacada com uma cor diferente conforme ilustrado na Figura E.9 e na Tabela E.2. Também é possível observar na Figura E.9 a falha da segmentação HK em regiões de bordas. Isso se deve ao fato das estimativas de derivativas e curvaturas nas regiões de descontinuidade da superfície não terem significado para classificação (Trucco e Verri, 1998).

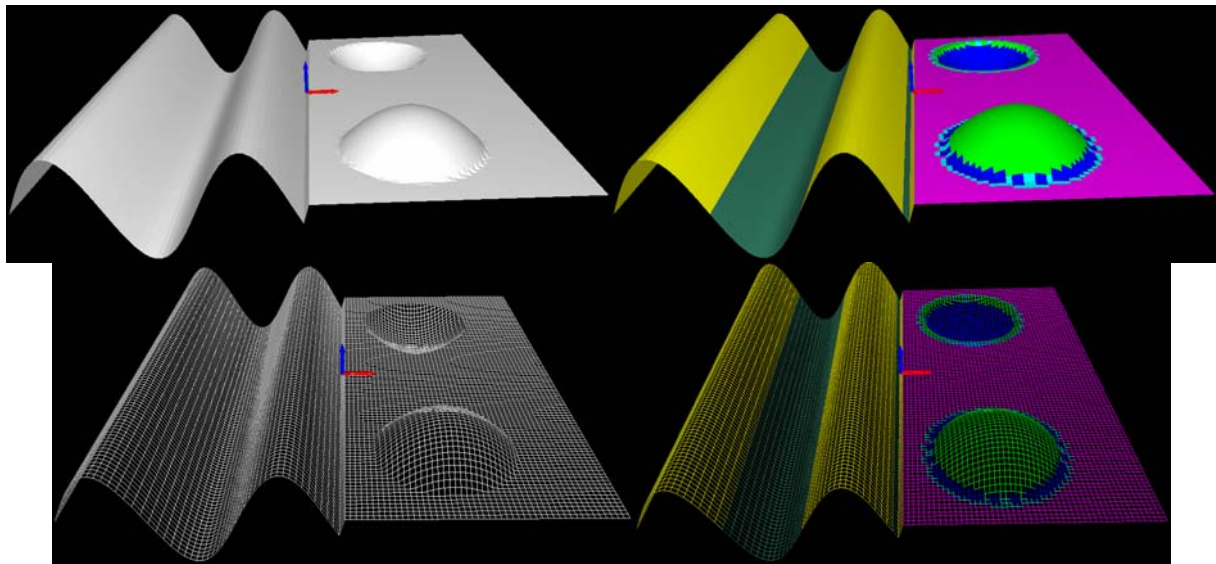


Figura E.9. Função de Segmentação HK sobre uma Superfície de Teste.

Tabela E.2. Código de Cores para a Classificação HK.

Cor	Superfície
Cyan	Hiperbólico
Yellow	Cilíndrico Convexo
Magenta	Plano
Green	Cilíndrico Côncavo
Bright Green	Elíptico Convexo
Blue	Elíptico Côncavo

É possível realizar uma avaliação de cada uma das curvaturas individualmente. Para isso basta selecionar qual das curvaturas deseja-se avaliar e, caso o módulo dessa curvatura seja superior ao limiar escolhido o ponto é destacado, conforme ilustrado na Figura E.10 e na Figura E.11.

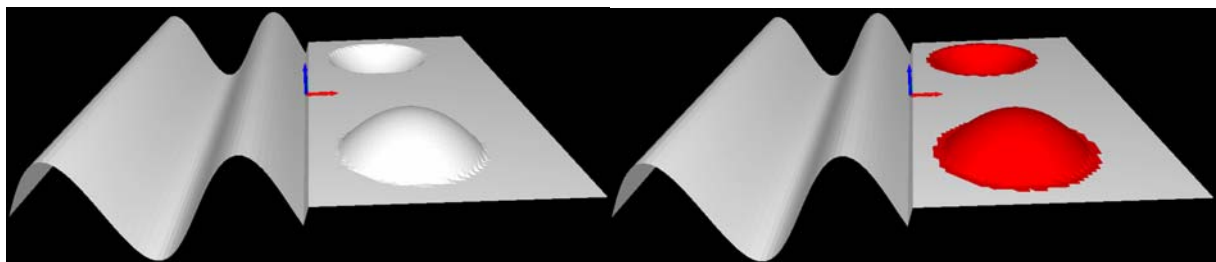


Figura E.10. Função de Curvatura Gaussiana sobre uma Superfície de Teste.



Figura E.11. Função de Curvatura Média sobre uma Superfície de Teste.

E.2.5 Plano Médio

A função de detecção através do plano médio é dividida em duas etapas principais. Inicialmente calcula-se o plano que melhor se adapta aos pontos na nuvem, minimizando o quadrado da distância entre os pontos e o plano, ou seja, calcula-se o plano que melhor representa os pontos da nuvem de acordo com o critério dos mínimos quadrados. Em seguida a distância entre cada um dos pontos que fazem parte da nuvem e esse plano é calculada de acordo com a Equação (E.8) (Kreyzig, 2006, Edwards e Penney, 1998, Swokowski, 1995) e, caso essa distância seja maior que um limiar especificado, este ponto é detectado e destacado.

$$d_i = \frac{|Ax_i + By_i - z_i + C|}{\sqrt{A^2 + B^2 + 1}}, \quad (\text{E.8})$$

onde o plano médio é dado por:

$$P_{\text{médio}} : Ax + By - Z + C = 0 \quad (\text{E.9})$$

e d_i é a distância entre esse plano e o i -ésimo ponto da nuvem.

Para o cálculo desse plano médio utiliza-se o conceito de pseudo-inversa, onde, dados os N pontos $[x_i \ y_i \ z_i]^T$ que constituem a nuvem, obtém-se o seguinte sistema de equações:

$$\begin{cases} Ax_1 + By_1 + C = z_1 \\ Ax_2 + By_2 + C = z_2 \\ \vdots \\ Ax_N + By_N + C = z_N \end{cases} \Rightarrow \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix}_{N \times 3} \begin{bmatrix} A \\ B \\ C \end{bmatrix}_{3 \times 1} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}_{N \times 1} \quad (\text{E.10})$$

Que pode ser reescrito como:

$$\mathbf{A}x = \mathbf{b}, \quad (\text{E.11})$$

onde

$$\mathbf{A} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix}_{N \times 3} \quad \mathbf{b} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}_{N \times 1} \quad x = \begin{bmatrix} A \\ B \\ C \end{bmatrix}_{3 \times 1} \quad (\text{E.12})$$

Utilizando então a pseudo-inversa da matriz \mathbf{A} é possível calcular diretamente cada um dos parâmetros do plano (Kreyzig, 2006, Trucco e Verri, 1998, Edward e Penney, 1998).

$$x = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (\text{E.13})$$

Desenvolvendo essa relação tem-se:

$$A = \frac{1}{\det(\mathbf{A}^T \mathbf{A})} \left[\begin{array}{l} N \sum_{i=1}^N y_i^2 \sum_{i=1}^N x_i z_i - \left(\sum_{i=1}^N y_i \right)^2 \sum_{i=1}^N x_i z_i - N \sum_{i=1}^N x_i y_i \sum_{i=1}^N y_i z_i \\ + \sum_{i=1}^N x_i \sum_{i=1}^N y_i \sum_{i=1}^N y_i z_i + \sum_{i=1}^N x_i y_i \sum_{i=1}^N y_i \sum_{i=1}^N z_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i^2 \sum_{i=1}^N z_i \end{array} \right] \quad (\text{E.14})$$

$$B = \frac{1}{\det(\mathbf{A}^T \mathbf{A})} \left[\begin{array}{l} -N \sum_{i=1}^N x_i y_i \sum_{i=1}^N x_i z_i + \sum_{i=1}^N x_i \sum_{i=1}^N y_i \sum_{i=1}^N x_i z_i + N \sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i z_i \\ - \left(\sum_{i=1}^N x_i \right)^2 \sum_{i=1}^N y_i z_i - \sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i \sum_{i=1}^N z_i + \sum_{i=1}^N x_i \sum_{i=1}^N x_i y_i \sum_{i=1}^N z_i \end{array} \right] \quad (\text{E.15})$$

$$C = \frac{1}{\det(\mathbf{A}^T \mathbf{A})} \left[\begin{array}{l} \sum_{i=1}^N x_i y_i \sum_{i=1}^N y_i \sum_{i=1}^N x_i z_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i^2 \sum_{i=1}^N x_i z_i - \sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i \sum_{i=1}^N y_i z_i \\ + \sum_{i=1}^N x_i \sum_{i=1}^N x_i y_i \sum_{i=1}^N y_i z_i + \sum_{i=1}^N x_i^2 \sum_{i=1}^N y_i^2 \sum_{i=1}^N z_i - \left(\sum_{i=1}^N x_i y_i \right)^2 \sum_{i=1}^N z_i \end{array} \right], \quad (\text{E.16})$$

com

$$\det(\mathbf{A}^T \mathbf{A}) = \sum_{i=1}^N x_i^2 \left[N \sum_{i=1}^N y_i^2 - \left(\sum_{i=1}^N y_i \right)^2 \right] + \sum_{i=1}^N x_i y_i \left[-N \sum_{i=1}^N x_i y_i + \sum_{i=1}^N x_i \sum_{i=1}^N y_i \right] + \sum_{i=1}^N x_i \left[\sum_{i=1}^N x_i y_i \sum_{i=1}^N y_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i^2 \right] \quad (\text{E.17})$$

Na Figura E.12 é possível observar uma nuvem de pontos e o plano médio calculado enquanto na Figura E.13 observa-se o efeito de detecção dessa função sobre uma superfície de teste.

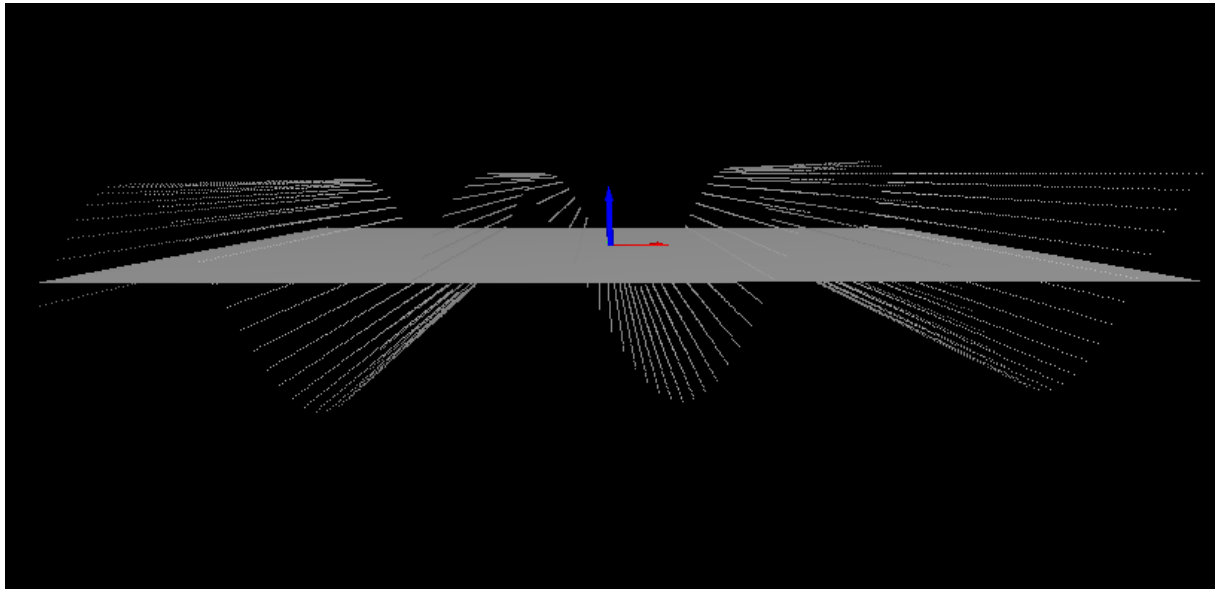


Figura E.12. Plano Médio Calculado para uma Nuvem de Pontos.

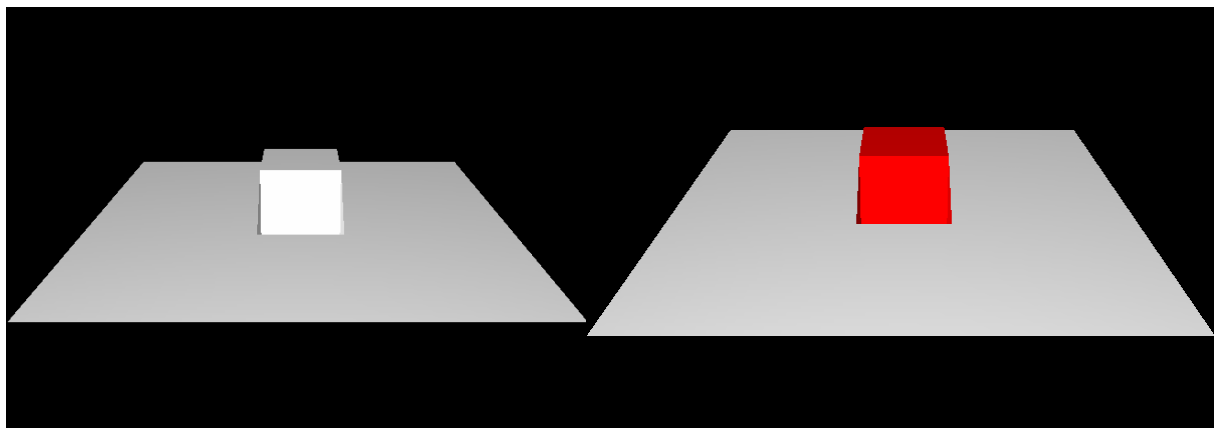


Figura E.13. Função de Plano Médio sobre uma Superfície de Teste.

E.2.6 Planicidade

A função de planicidade é na verdade uma expansão da função de detecção através do plano médio. Ela baseia-se no cálculo do plano que melhor representa os pontos da nuvem de acordo com o critério dos mínimos quadrados apresentado no Apêndice E e no cálculo da distância entre cada um dos pontos que fazem parte da nuvem e esse plano. Entretanto, ao invés de detectar os pontos cuja distância até o plano seja superior a um limiar, a função de planicidade faz um levantamento estatístico dessas distâncias, tratando-as como erros.

Ou seja, calculadas as distâncias d_i de cada ponto da nuvem até o plano médio conforme a Equação (E.8), são calculados os valores máximo, médio e de desvio padrão dessas distâncias conforme ilustrado na Equação (E.18) (Bussab e Morettin, 1987).

$$\left\{ \begin{array}{l} \bar{E} = \frac{\sum_{i=1}^m d_i}{m} \\ E_{\max} = \max(d_i) \\ \sigma = \sqrt{\frac{\sum_{i=1}^m (d_i - \bar{d})^2}{m}} \end{array} \right. , \quad (\text{E.18})$$

onde m é o número total de pontos da nuvem.

Uma vez calculados todos esses valores eles são salvos em um arquivo de texto.

Ao observar esses valores de distância como erros, avalia-se o quanto a nuvem de pontos se aproxima de um plano ideal. Essa é uma das abordagens utilizadas para a avaliação da precisão do sistema VISSCAN-3D realizada no Capítulo 7.

APÊNDICE F: O Processo de Calibração do Sistema Passo a Passo

Nesse apêndice é apresentado detalhadamente o *software* de calibração de câmeras e do sistema óptico, sendo explicada cada uma de suas funcionalidades.

O *software* tem como principal objetivo a calibração completa do sistema de digitalização VISSCAN-3D, tornando desnecessário o uso de outros sistemas de medição para a realização desse procedimento, entretanto ele também pode ser utilizado para a calibração de qualquer câmera utilizando o modelo de restrição de alinhamento radial (RAC).

As funcionalidades desse *software* podem ser divididas em três grupos. A aquisição e processamento de imagens para extração de dados, a calibração de câmeras e a calibração da arquitetura óptica do sistema VISSCAN-3D. Cada um desses grupos de funcionalidades apresenta uma interface amigável e bem destacada, conforme ilustrado na Figura F.1.

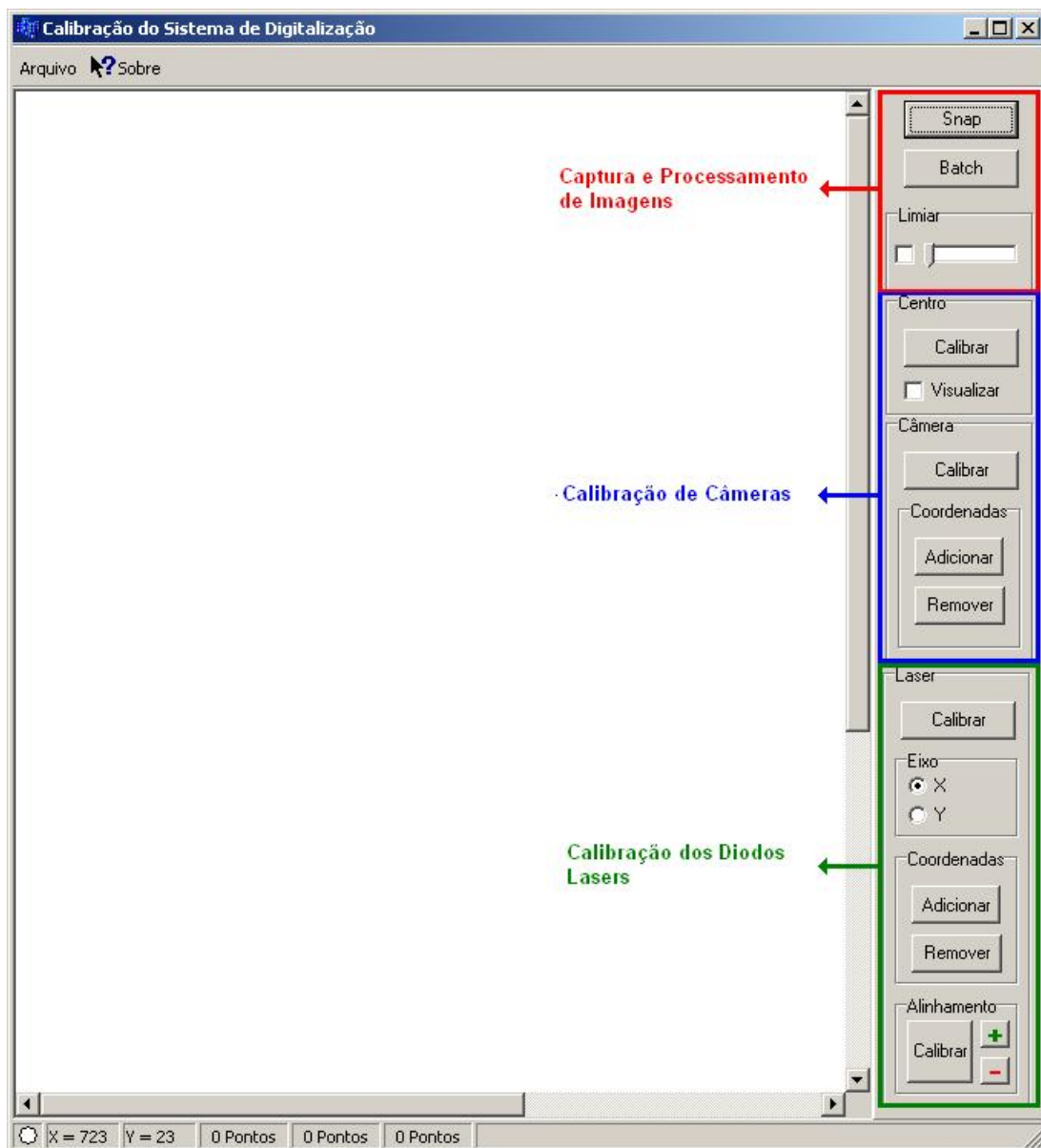


Figura F.1. Tela Principal do Software de Calibração e Grupos de Funcionalidades.

O sistema possibilita a aquisição de imagens a partir de câmeras conectadas ao computador através da interface *CameraLink*, ou através de uma conexão USB e oferece a opção de aplicação de uma operação de binarização dessa imagem através da aplicação de um limiar simples que é controlado pela barra deslizante ilustrada na Figura F.2.

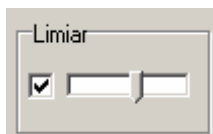


Figura F.2. Barra Deslizante para Aplicação de Binarização por Limiar Fixo em Imagens.

Apesar de ser capaz de capturar e binarizar uma imagem qualquer, as funcionalidades do sistema são voltadas para o processamento da imagem de uma placa de calibração. Na Figura F.3 pode ser observada uma imagem capturada e binarizada.

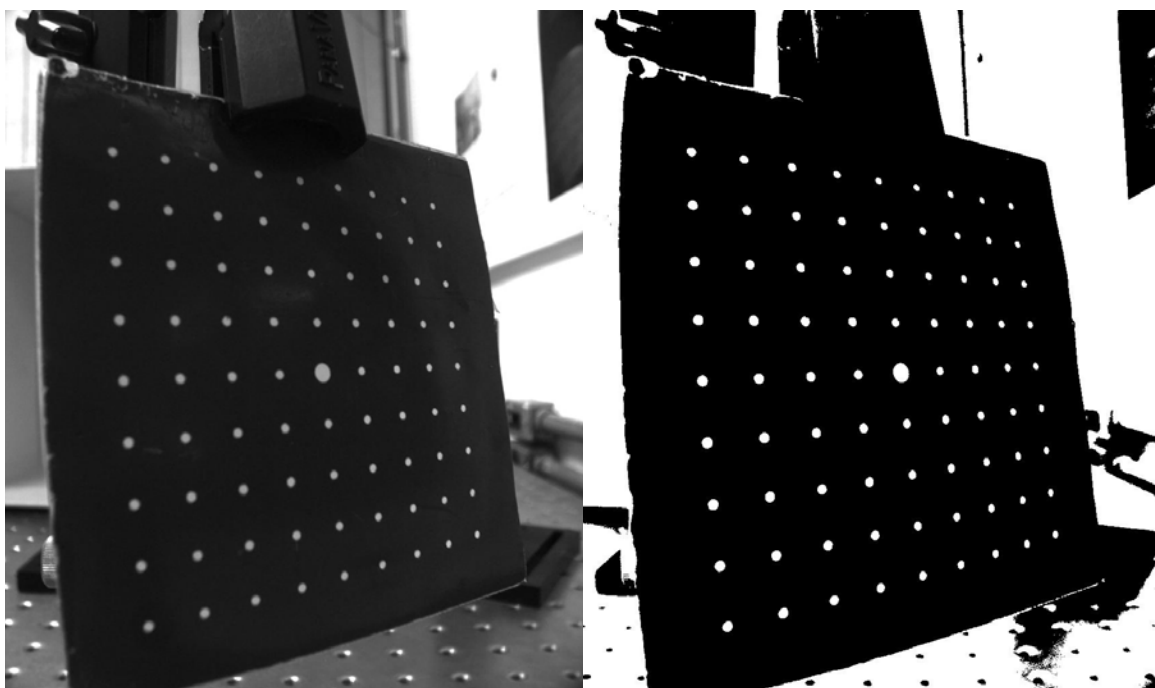


Figura F.3. Imagem Capturada e Binarizada da Placa de Calibração.

Após a realização da binarização da imagem da placa é necessário obter-se as coordenadas em pixels de cada um dos pontos da placa de calibração e as coordenadas desses pontos em unidades do sistema internacional (mm) em relação a um referencial global qualquer.

A determinação das coordenadas globais de cada um dos pontos é realizada automaticamente pelo sistema baseado nos parâmetros configurados pelo usuário. Cada um desses parâmetros e sua descrição podem ser observados na Tabela F.1 e na Figura F.4.

Tabela F.1. Parâmetros para Cálculo das Coordenadas da Placa de Calibração.

Parâmetros	Descrição
N_{Linhas}	Número de Linhas de Pontos na Placa de Calibração
N_{Pontos}	Número de Pontos em Cada Linha de Pontos
E_{Pontos}	Distância entre Cada um dos Pontos da Placa
(O_x, O_y)	Coordenadas do Ponto Inicial da Placa em Unidades S.I.

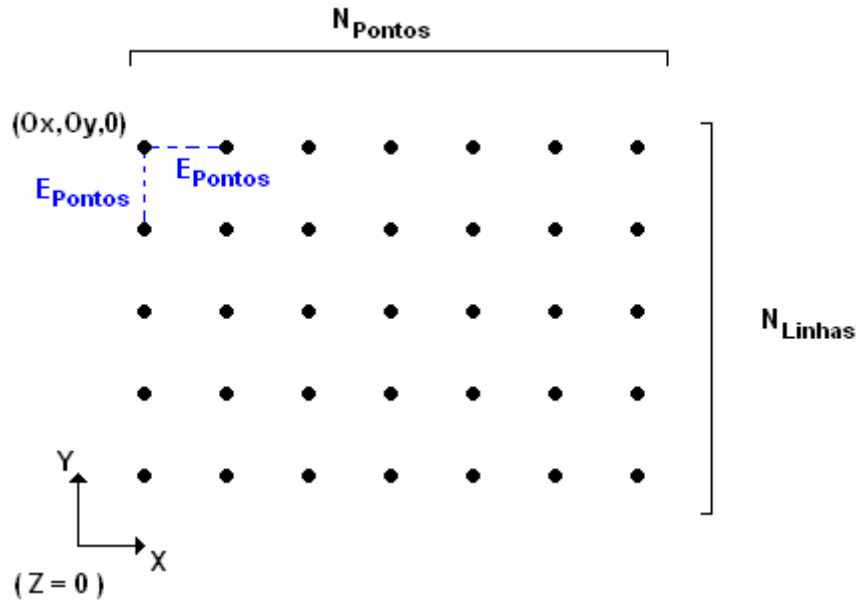


Figura F.4. Parâmetros para Cálculo das Coordenadas da Placa de Calibração.

A entrada automática de dados é realizada seguindo-se a seqüência específica ilustrada na Figura F.5, de forma que a entrada das coordenadas em pixels realizada posteriormente pelo usuário deve seguir o mesmo padrão.

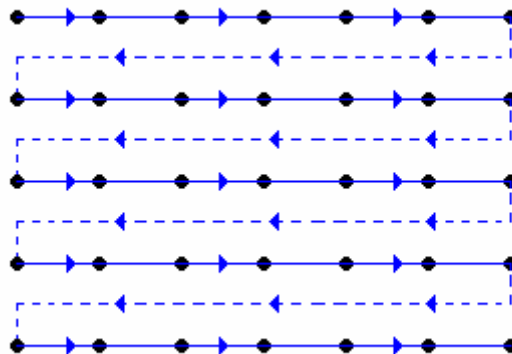


Figura F.5. Seqüência de Entrada dos Pontos da Placa de Calibração.

A entrada das coordenadas em pixels de cada ponto pode ser realizada de várias formas. O usuário pode selecionar manualmente cada um dos pontos, ou selecionar os pontos iniciais e deixar que os

pontos restantes sejam determinados automaticamente, e ainda tem a opção de inserir o centro geométrico de uma região.

Para iniciar a entrada de pontos da placa de calibração o usuário deve selecionar o botão “Adicionar” dentro do grupo de funções para calibração de câmeras conforme ilustrado na Figura F.6.



Figura F.6. Interface para Adicionar Coordenadas para a Calibração de Câmeras.

Em seguida basta clicar sobre a imagem no ponto que se deseja adicionar. Ao clicar com o botão esquerdo do *mouse* o usuário estará adicionando exatamente o ponto onde ele clicou, entretanto, utilizando o botão direito do *mouse* o sistema irá calcular o centro geométrico da região selecionada e adicionará esse ponto, removendo então a região da imagem conforme ilustrado na Figura F.7. Em ambos os casos o sistema adiciona à imagem um marcador para destacar o ponto adicionado.

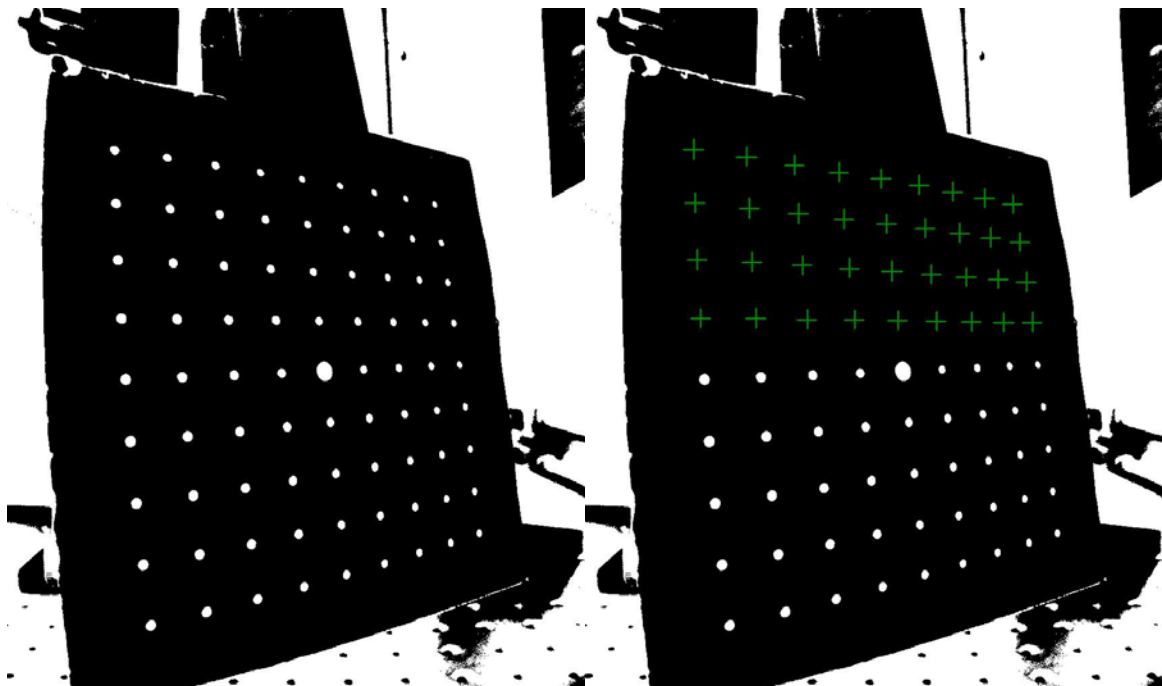


Figura F.7. Adição de Coordenadas para a Calibração de Câmeras.

Após a inserção de dois pontos o sistema oferece a opção de determinação automática dos pontos restantes. Caso o usuário deseje realizar a inserção manual dos pontos é necessário lembrar da seqüência de inserção ilustrada na Figura F.5 que deve ser seguida.

O botão “Remover” localizado logo abaixo do botão “Adicionar” pode ser utilizado caso uma coordenada seja adicionada erroneamente, ou caso deseje-se iniciar uma nova calibração. Em ambos os casos ele remove completamente todas as coordenadas adicionadas, sendo necessário reiniciar todo o processo de inserção de coordenadas.

Após a inserção no sistema de todos pontos desejados, clica-se novamente no botão “Adicionar”, desmarcando o botão.

Uma vez finalizada a entrada de dados pode-se realizar a calibração da câmera e do centro da imagem. Para isso basta clicar nos botões “Calibrar” localizados dentro do grupo de interface da câmera e do centro.

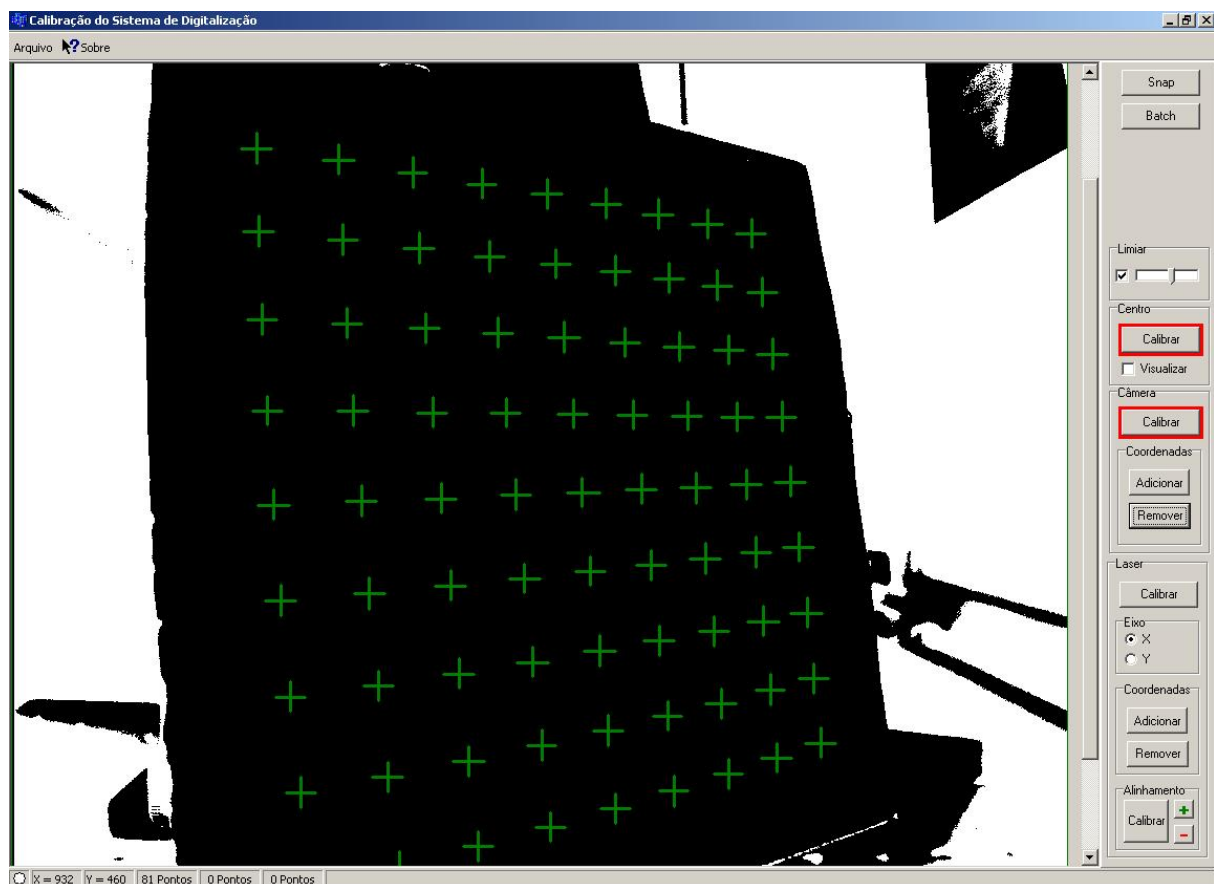


Figura F.8. Calibração de Câmeras.

Após a calibração da câmera é possível realizar então a calibração do sistema óptico formado pelos diodos lasers. Para isso, é realizada, assim como para a calibração da câmera, a entrada de pontos que fazem parte da linha laser. Entretanto, para as linhas lasers, não é necessário o uso de uma seqüência específica de entrada. Basta selecionar o botão “Adicionar” e clicar sobre os pontos que formam a linha laser. Pode-se clicar com o botão esquerdo, adicionando-se exatamente o pixel onde foi clicado, ou pode-se utilizar o botão direito do mouse. Ao utilizar o botão direito do *mouse* obtem-se diferentes resultados dependendo do eixo do laser selecionado conforme ilustrado na Figura F.11 e na Figura F.12.

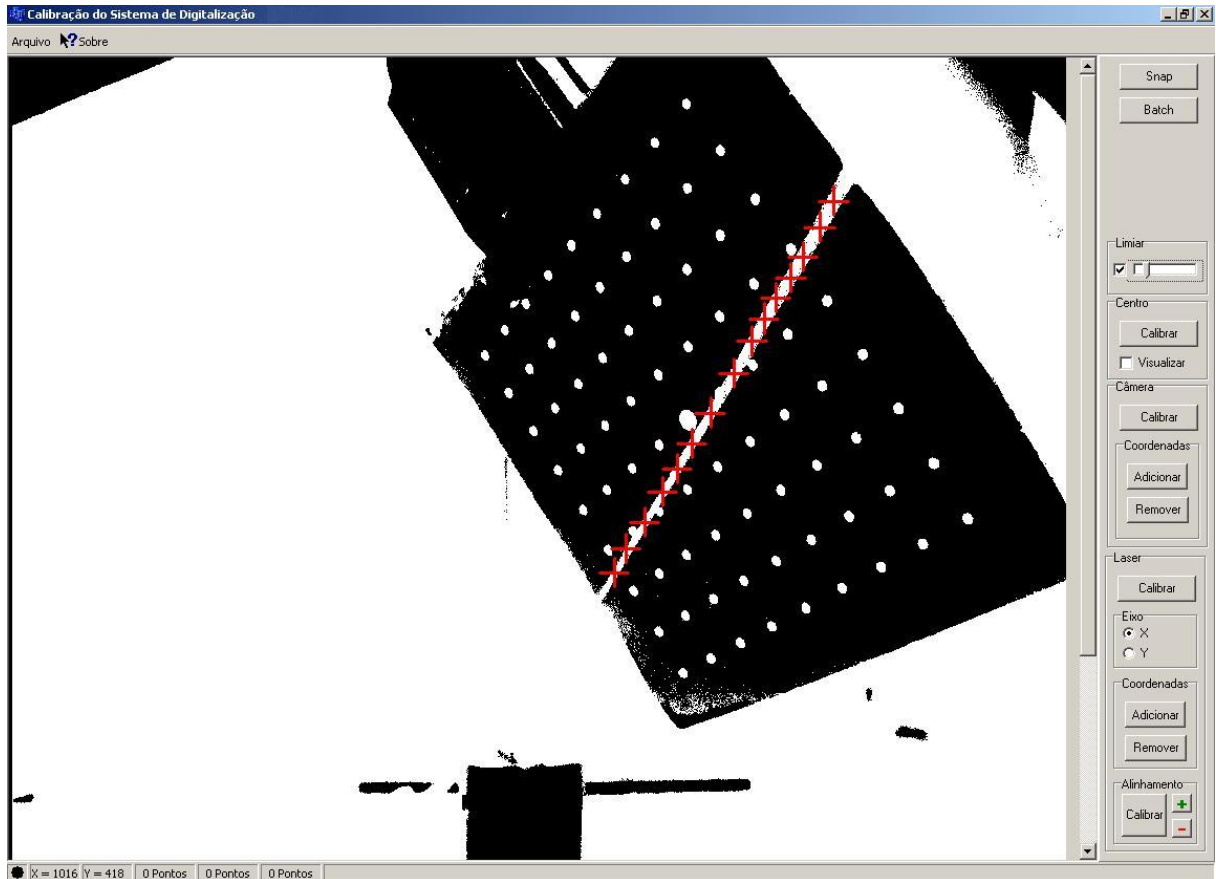


Figura F.9. Calibração dos Lasers.



Figura F.10. Grupo de Seleção de Eixos.

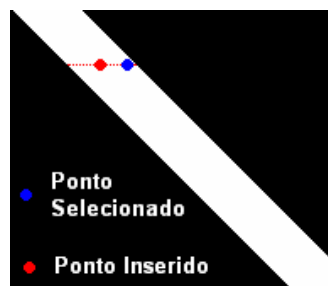


Figura F.11. Adição de Coordenadas da Linha Laser no Eixo X.

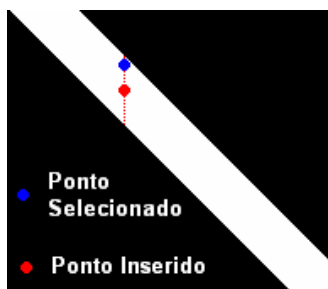


Figura F.12. Adição de Coordenadas da Linha Laser no Eixo Y.

Ao adicionar um ponto que pertence ao diodo laser cujos parâmetros a serem calibrados estão no eixo X, o sistema calcula o ponto médio do comprimento no eixo X da linha laser no ponto selecionado, e insere o ponto calculado conforme a Figura F.11. Já ao adicionar um ponto que pertence ao diodo laser cujos parâmetros a serem calibrados estão no eixo Y, o sistema calcula o ponto médio do comprimento no eixo Y conforme a Figura F.12.

Uma vez adicionados os pontos que formam a linha laser, basta clicar no botão “Calibrar” para calibrar os parâmetros do diodo laser.

A etapa seguinte da calibração de um diodo laser é o cálculo do seu desalinhamento em relação ao centro óptico. Para isso é necessário realizar a calibração do laser no mínimo duas vezes. Ao finalizar cada um dos procedimentos de calibração do laser clica-se no botão “+” dentro do grupo de interface de “Alinhamento”. E após a inserção de no mínimo dois resultados de calibração, clica-se no botão “Calibrar” para concluir a calibração do desalinhamento do diodo laser.